Testing, Learning, Sampling, Sketching

by

Nathaniel Harms

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Computer Science

Waterloo, Ontario, Canada, 2022

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner:	Hamed Hatami Associate Professor, School of Computer Science, McGill University
Supervisor:	Eric Blais Associate Professor, David R. Cheriton School of Computer Science, University of Waterloo
Internal Member:	Lap Chi Lau Professor, David R. Cheriton School of Computer Science, University of Waterloo
Internal-External Member:	Ashwin Nayak Professor, Institute for Quantum Computing and Department of Combinatorics and Optimization, University of Waterloo
Other Member(s):	Gautam Kamath Assistant Professor, David R. Cheriton School of Computer Science, University of Waterloo

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

This thesis contains results from the following papers:

- [Har20]: *Universal Communication, Universal Graphs, and Graph Labeling.* Presented at ITCS 2020.
- [BFH21]: *VC Dimension and Distribution-Free Sample-Based Testing.* Joint work with Eric Blais and Renato Ferreira Pinto Jr. Presented at STOC 2021.
- [HY21]: *Downsampling for Testing and Learning in Product Distributions.* Joint work with Yuichi Yoshida. Presented at ICALP 2022.
- [HWZ22]: *Randomized Communication and Implicit Graph Representations.* Joint work with Sebastian Wild and Viktor Zamaraev. Presented at STOC 2022.
- [EHK22]: *Sketching Distances in Monotone Graph Classes.* Joint work with Louis Esperet and Andrey Kupavskii. Presented at RANDOM 2022.
- [EHZ22]: *Optimal Adjacency Labels for Subgraphs of Cartesian Products.* Joint work with Louis Esperet and Viktor Zamaraev.

I will clearly identify (usually at the beginning of each chapter) which of these papers the results in this thesis are taken from. All results should be considered equal collaborations between coauthors.

This thesis also briefly cites the following paper, which was completed during my PhD program and is closely related to part I of my thesis, but which is not included here because it was partially included in my Master's thesis:

- [Har19]: *Testing Halfspaces over Rotation-Invariant Distributions.* Presented at SODA 2019.

The proof of Theorem 1.3.18 is due to Bonamy & Girão and was communicated to me by Louis Esperet; it does not appear in any published work.

Abstract

We study several problems about sublinear algorithms, presented in two parts.

Part I: Property testing and learning. There are two main goals of research in property testing and learning theory. The first is to understand the relationship between testing and learning, and the second is to develop efficient testing and learning algorithms. We present results towards both goals.

- An oft-repeated motivation for property testing algorithms is to help with *model selection* in learning: to efficiently check whether the chosen hypothesis class (i.e. learning model) will successfully learn the target function. We present in this thesis a proof that, for many of the most useful and natural hypothesis classes (including halfspaces, polynomial threshold functions, intersections of halfspaces, etc.), the sample complexity of testing in the distribution-free model is nearly equal to that of learning. This shows that testing does not give a significant advantage in model selection in this setting.
- We present a simple and general technique for transforming testing and learning algorithms designed for the *uniform* distribution over $\{0,1\}^d$ or $[n]^d$ into algorithms that work for arbitrary *product distributions* over $\mathbb{R}^d$. This leads to an improvement and simplification of state-of-the-art results for testing monotonicity, learning intersections of halfspaces, learning polynomial threshold functions, and others.

Part II. Adjacency and distance sketching for graphs. We initiate the thorough study of adjacency and distance sketching for classes of graphs. Two open problems in sublinear algorithms are: 1) to understand the power of randomization in communication; and 2) to characterize the sketchable distance metrics. We observe that *constant-cost* randomized communication is equivalent to *adjacency sketching* in a hereditary graph class, which in turn implies the existence of an efficient *adjacency labeling scheme*, the subject of a major open problem in structural graph theory. Therefore characterizing the adjacency sketchable graph classes (i.e. the constant-cost communication problems) is the probabilistic equivalent of this open problem, and an essential step towards understanding the power of randomization in communication.

This thesis gives the first results towards a combined theory of these problems and uses this connection to obtain optimal adjacency labels for subgraphs of Cartesian products, resolving some questions from the literature. More generally, we begin to develop a theory of graph sketching for problems that generalize adjacency, including different notions of distance sketching. This connects the well-studied areas of distance sketching in sublinear algorithms, and distance labeling in structural graph theory.

Acknowledgments

I am enormously grateful to my adviser, Eric Blais, whose guidance has been invaluable. He has encouraged me and pushed me to be independent and creative, giving me freedom to choose my own research interests, while also being available to help at any time.

Thanks to Yuichi Yoshida for hosting me at the National Institute of Informatics. Although my visit was cut short in March 2020 due to covid, what I learned in that visit had a strong influence on my later research, and I have missed the city of Tokyo ever since.

I owe a great debt to Sebastian Wild, Viktor Zamaraev, and Louis Esperet. When I got stuck trying to develop the relation between labeling schemes and communication complexity, Sebastian, Viktor, and Louis helped turned this idea into a full research program and a 3-month visit to Europe. The second half of this thesis could only happen due to Sebastian, who graciously remembered my ideas after just one conversation, and took them to across the ocean. Viktor, Louis, and Sebastian are excellent hosts and collaborators, who worked hard to bring me for a visit and patiently instruct me on graph theory.

The second chapter of this thesis would not have been conceived or completed without Renato Ferreira Pinto Jr, who has quickly become a close collaborator and friend. In his first year of PhD he is already slinging equations with enviable velocity. He deserves a PhD by the end of the year.

I am grateful to Amit Levi, Abhinav Bommireddi, Sharat Ibrahimpur, and Cameron Seth for feedback on several of the papers included in this thesis.

Many thanks to the numerous good and interesting humans I've met during my graduate studies, who have helped make these years the most enjoyable of my life (so far). Some particular good humans are, in no particular order: Anil Paçacı, Ali Wytsma, Abhinav Bommireddi, Amit Levi, Nate Braniff, Tom Bury, Sasha Vtyurina, Christian Gorenflo, Priyank Jaini, Sajin Sasy, Camila Perez Gavilan, Michael Abebe, Brad Glasburgen, Julie Messier, Sharat Ibrahimpur, Hemant Saxena, Priya Soundararajan, Harshita Mistry, and Hari Govind. A special mention goes to Masoumeh Shafieinejad, whose voice in the back of my mind is always challenging me to improve and achieve.

Finally, thanks to my examination committee: Lap Chi Lau, Gautam Kamath, Ashwin Nayak, and Hamed Hatami, for taking the time to review this thesis, and for their comments and excellent questions.

Dedication

This thesis is dedicated to:

My brother, Quinten Harms, whose honesty and openness inspire me to be a better scientist and person;

My mom, Shelly Harms, who always encouraged me to be weird;

And my dad, Robert Harms, who taught me to work hard and dream big.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

> *One of the symptoms of approaching nervous breakdown is the belief that one's work is terribly important... If I were a medical man, I should prescribe a holiday to any patient who considered his work important.* – Bertrand Russell[1]

Let me now explain why the work in this thesis is terribly important. It is a misfortune of humanity that we must often make decisions with incomplete and unreliable information. This is a misfortune that we increasingly inflict upon our algorithms, too. Although unlucky for the computers that serve us, this is lucky for computer scientists because, by studying these algorithms, we gain (incomplete, unreliable) information about the human condition. Considering the importance of making decisions, it is therefore inarguable that the study of such algorithms is the most urgent scientific – indeed, humanitarian – endeavor. This thesis was once the most recent progress in this study.

Classical computer science and algorithm design, including almost all standard undergraduate material, focuses on computational problems where the algorithm is given an input x and must compute a function $f(x)$, either exactly or approximately. An important assumption is that the *entire* input is known to the algorithm. In other words, the algorithm has access to *all* relevant information about the problem, and the question is how many computational resources are required to process this information.

These classical algorithms operate in a universe that is both *small* and *certain*. It is *small* because the input x must be small: an efficient algorithm in the classical sense, which runs in time polynomial in the size of x, or even *linear* in the size of x, will be prohibitively

[1] *The Conquest of Happiness*, Chapter 5.

expensive when x is very large. And it is *certain* because the algorithms have complete information about x and the problem they are required to solve.

A cursory examination of our universe reveals that it is neither small nor certain. We ourselves must constantly make decisions based upon data which are both incomplete and uncertain, and this is increasingly true of our computer programs as well. A human decision is an attempt to compute a function $f(x)$ where x is the state of the universe, observable only through a weak and often inaccurate lens. As the role of computers expands, from performing well-defined tasks like arithmetic, to making automated decisions in the real world, our algorithms must operate in the *large* and *uncertain* universe that humans occupy.

It is therefore necessary to study algorithms that compute functions $f(x)$ when x is large and intractable, accessible only through a lens that is weak and untrustworthy. This thesis is a small piece of the mechanism to cope with this misfortune that we now share with our computers.

1.1 Thesis Overview

This thesis studies several problems about *sublinear algorithms*. Sublinear algorithms eliminate the *small universe* assumption (which assumes that the algorithm can observe all relevant information in a reasonable amount of time), and often challenge the *certain universe* assumption too (which assumes that the algorithms has perfectly accurate observations). Sublinear algorithms are algorithms that compute something about the input x without observing the entire input x, and often under severe restrictions on the *type* of observations that they can make.

It is helpful to imagine that the input x and the algorithm itself are split between several entities and that their communication is limited in quantity or in type. For example, communication may be restricted to certain types of messages, or to a certain number of bits. Part I of this thesis studies the *oracle model*, where the input is held by an oracle (or set of oracles), and the algorithm learns about the input only through communication with this oracle. Part II of this thesis studies the *communication* and *sketching* models. In the communication model, the input is split between two parties who must communicate as little as possible to solve a problem; and in the *sketching* model, the input is held by an algorithm that produces a small *sketch* to send to a second algorithm, which must compute the output.

Part I: The Oracle Model. The oracle model captures both *machine learning algorithms* and *property testing algorithms*. In this model, x is held by an *oracle* who answers certain types of questions about x. The goal is to solve the problem while asking as few questions as possible, and also to weaken the oracle as much as possible so that it answers the least-informative types of questions. While it is of practical and theoretical interest to design more efficient testing and learning algorithms (and this thesis presents a number of these), the main focus of Part I is the relationship between these two types of algorithms, formalized in the *testing vs. learning* question. This question, asked by Goldreich, Goldwasser, & Ron [GGR98], is one of the most important questions in sublinear algorithms because it asks when testing algorithms can be used to improve the process of model selection. This is one of the most commonly suggested applications for property testing, but this thesis will show that in many natural and practical cases it is not possible.

We formally introduce this model and the related questions in Section 1.2, along with a summary of the results contained in this thesis. A detailed treatment of the results is in Chapters 2 and 3. This part of the thesis includes results from [BFH21] (coauthored with Eric Blais and Renato Ferreira Pinto Jr.) and [HY21] (coauthored with Yuichi Yoshida).

Part II: The Communication and Sketching Model. In the second part of this thesis, we study the phenomenon of *constant-cost randomized communication*. Some communication problems (like EQUALITY) can be computed with a randomized communication protocol whose cost does not increase with the size of the inputs. Understanding this phenomenon is essential for understanding randomized communication, but we will see that it is also closely related to major open problems in structural graph theory and distributed data structures. This thesis introduces a model where the input (a graph G from a certain class $\mathcal{F}$) is held by a *sketching algorithm* that must produce small randomized "sketches" of the vertices of G, so that a second party who does not know G can decide adjacency or distance between vertices using only the sketches. These types of sketches are natural to study on their own, but they also form a connection between constant-cost communication, distance sketching, and *informative labeling schemes*; in particular, we use this correspondence to resolve a few open questions from the recent literature on *adjacency labeling schemes*.

We introduce these problems and their connections in Section 1.3, and we give a detailed treatment in Chapters 4 to 7. This part of the thesis includes results from [Har20], [HWZ22] (coauthored with Sebastian Wild and Viktor Zamaraev), [EHK22] (coauthored with Louis Esperet and Andrey Kupavskii), and [EHZ22] (coauthored with Louis Esperet and Viktor Zamaraev).

1.2 The Oracle Model

In the oracle model, the input x is held by an *oracle*, or a set of oracles, which provide the algorithm with information about x by responding to requests of a certain form. Two main areas of research on the oracle model are 1) the *testing vs. learning* question, and 2) designing efficient testing and learning algorithms. In Section 1.2.1, we give an example that motivates the oracle model. Section 1.2.2 gives formal definitions of the oracle model, machine learning, and property testing algorithms. In Section 1.2.3, we discuss our results towards the *testing vs. learning* question, and in Section 1.2.4, we discuss our results on designing efficient testing and learning algorithms.

1.2.1 Motivating Example

Modern astronomy requires not only advanced telescopes, which produce enormous amounts of data, but also algorithms for processing those data. One important type of processing is to identify or distinguish particular astronomical phenomena, such as identifying gravitational lens effects [MMA+19] or distinguishing between stars and galaxies [SNHM+18] (see also [Bar19] and the references therein).

To motivate the oracle model of computation, the problems of learning and testing, and some of the specific problems we study in this thesis, consider the problem of designing an algorithm to detect gravitational lens effects (as in [MMA+19, N+19]). Say that our telescope produces observations which are $n \times n$ matrices of photon counts: so the telescope has an $n \times n$ array of sensors, and produces observations $X \in \mathbb{N}^{n \times n}$ which record the number of photons hitting each sensor. For now, we retain the (egregiously unrealistic) *certainty* assumption of the universe, discussed above, which in particular means that there is a well-defined function $\mathsf{lens} : \mathbb{N}^{n \times n} \to \{0, 1\}$ which satisfies $\mathsf{lens}(X) = 1$ if and only if X is an observation of a gravitational lens effect[2]. Our goal is to produce an algorithm that computes the function lens.

However, even under the assumption that lens is a well-defined function, we may not actually know what that definition is. And, if we did, an algorithm that computes it is likely too complicated to implement in a reasonable amount of time. Instead of creating the algorithm ourselves, we want an algorithm that creates our algorithm for us: a *learning algorithm*. Here are some requirements and restrictions on this learning algorithm:

[2]This assumption is unrealistic for a number of reasons, including that a *gravitational lens effect* is a human categorization of empirical observations, with no formal mathematical definition.

1. **Its access to the input is restricted.** The input to this learning algorithm is the function $\mathsf{lens} : \mathbb{N}^{n \times n} \to \{0, 1\}$, and the desired output is an algorithm that computes lens. But the algorithm cannot see the whole input: we scientists don't even know what the input is. The learning algorithm only gets to see certain *examples* X and their labels $\mathsf{lens}(X)$. That is, the learning algorithm can interact with its input only in a restricted way. We model this by saying that the input is held by an *oracle*, who responds to certain types of requests. The learning algorithm probably cannot construct its own examples and ask for labels, which would require algorithmically producing reasonable telescope images[3]. So we give the input to an oracle who can only provide *random* examples drawn from some probability distribution $\mathcal{D}$ (which we think of, informally, as the distribution over images received from the telescope). Of course, we also don't know $\mathcal{D}$, so the algorithm should work for *any* $\mathcal{D}$.

2. **It sees only a small number of examples.** We cannot allow the learning algorithm to see every possible example of a telescope image before producing its output. There are too many possible telescope images, and the scientists are not paid enough to assign labels to every single one. We must require that the learning algorithm produces its output after seeing an extremely small number of examples, compared to the total number of possible telescope images.

3. **It cannot be able to output any arbitrary function.** We cannot expect the learning algorithm to contain every possible function $f : \mathbb{N}^{n \times n} \to \{0, 1\}$ in its output space; it would require unbounded time to produce these functions. So we instead choose a certain *hypothesis class* $\mathcal{H}$ of functions that the algorithm could produce efficiently, and restrict the algorithm to produce a function from this class.

4. **It must be reasonably accurate.** We cannot expect the algorithm to be 100% accurate, because of all the restrictions we've placed on it. We could demand that the algorithm is accurate on 99% of possible telescope images, but this would not be helpful: many *possible* images might have extremely low likelihood of appearing in our telescope. It might be the case that the images we actually see comprise only 1% of possible telescope images. We want to be accurate with respect to $\mathcal{D}$, the probability distribution over observed images (which we don't know). It also might be the case that our chosen hypothesis class $\mathcal{H}$ is not powerful enough to be accurate. So we ask that

[3]The training data in [MMA+19] are synthetic, produced using physics simulations. So it is sometimes possible for an algorithm to produce query points, although here it is computationally expensive. But the particular setup of [MMA+19] is unusual; see references in [Bar19] for more typical situations.

our algorithm produces a function $f \in \mathcal{H}$ such that

$$\mathop{\mathbb{P}}_{X \sim \mathcal{D}} [f(X) \neq \mathsf{lens}(X)] \leq \mathsf{OPT} + \epsilon \,,$$

where $\mathsf{OPT} = \inf_{g \in \mathcal{H}} \mathop{\mathbb{P}}_{X \sim \mathcal{D}} [g(X) \neq \mathsf{lens}(X)]$ is the optimal performance of $\mathcal{H}$, and $\epsilon > 0$ is our desired accuracy.

This informally introduces the problem of learning, and the motivation for the oracle model. To motivate *property testing*, we consider a standard method for designing a learning algorithm. We first choose some set of efficiently computable *features* of a telescope image. Call these $f_1, \ldots, f_k : \mathbb{N}^{n \times n} \to \mathbb{R}$. For each image X, we can construct a feature vector $f(X) = (f_1(X), f_2(X), \ldots, f_d(X))$ in $\mathbb{R}^d$. In this way, we transform each example $(X, \mathsf{lens}(X))$ into an example $(f(X), \mathsf{lens}(X))$ in $\mathbb{R}^d \times \{0, 1\}$. (This is often called the *kernel trick*.)

Then, we apply the *support-vector machine* (SVM) algorithm, which will attempt to find a hyperplane in $\mathbb{R}^d$ that separates the 0-valued transformed examples from the 1-valued transformed examples. Our choice to use an SVM and our choice of features define a hypothesis class $\mathcal{H}$: the resulting learning algorithm will produce an output in $\mathcal{H}$. We want to know whether our choices will work. This is the subject of a typical applied machine learning paper: to make choices like this and see if they work. This is usually done by running the learning algorithm on some sample data and checking its accuracy on a test set.

What if there was an algorithm which could efficiently test whether a hypothesis class $\mathcal{H}$ will work (i.e. check whether $\mathsf{OPT} = 0$ or $\mathsf{OPT} < \epsilon$), without having to run the learning algorithm? This could help enormously with the task of choosing a learning algorithm. We could, say, select a set of features and run a *testing algorithm* to see if SVM will produce good results, before investing in a learning algorithm. This application was suggested by Goldreich, Goldwasser, & Ron [GGR98], who formalized the **testing vs. learning question**:

Question 1.2.1 (Testing vs. Learning)**.** *Which hypothesis classes $\mathcal{H}$ can be* tested *more efficiently than they can be* learned*?*

This is not the only proposed application of property testing algorithms, but it is one of the most common. But testing algorithms do not seem to be used this way in practice. In this thesis, we give theoretical results showing that this application is not feasible in some of the most common settings. In particular, we will prove that when the input distribution $\mathcal{D}$ is unknown and unrestricted, the algorithm sees only samples, and the hypothesis class

is a hyperplane or another practically useful class (as in the SVM example above), a testing algorithm cannot improve significantly upon the efficiency of a learning algorithm. In other words, the empirical method of running the learning algorithm and checking if it worked is, provably, almost optimal.

1.2.2 Definitions: Oracles, Testing, and Learning

We now give formal definitions of the model and problems introduced in the previous subsection.

Definition 1.2.2 (Distance)**.** Let $\mathcal{X}$ be some domain, let $\mathcal{D}$ be a probability distribution over $\mathcal{X}$, and let $\mathcal{H}$ be a set of functions $\mathcal{X} \to \{0,1\}$ (measurable with respect to $\mathcal{D}$). Then we define the following (we will drop the subscript $\mathcal{D}$ when it is clear from context):

$$\begin{aligned} \mathsf{dist}_{\mathcal{D}}(f,g) &:= \mathop{\mathbb{P}}_{x\sim\mathcal{D}}[f(x) \neq g(x)] \\ \mathsf{dist}_{\mathcal{D}}(f,\mathcal{H}) &:= \inf_{g\in\mathcal{H}} \mathsf{dist}_{\mathcal{D}}(f,g)\,. \end{aligned}$$

For a domain $\mathcal{X}$, a set $\mathcal{H}$ of functions $\mathcal{X} \to \{0,1\}$, and a class $\mathfrak{D}$ of probability distributions over $\mathcal{X}$, we define an oracle algorithm as follows. Let $\mathcal{O} = \{\mathcal{O}_1, \mathcal{O}_2, \dots\}$ be a collection of *oracles*. An algorithm in the $\mathcal{O}$-oracle model is given as input a parameter $\epsilon > 0$, a function $\mathcal{X} \to \{0,1\}$, and a probability distribution $\mathcal{D}$ over $\mathcal{X}$. On each input $(f,\mathcal{D}) \in \mathcal{H} \times \mathfrak{D}$, the oracles are instantiated as $\mathcal{O}_1(f,\mathcal{D}), \mathcal{O}_2(f,\mathcal{D}), \dots$, which can provide certain information about $(f,\mathcal{D})$ upon request, at unit cost. The algorithm is allowed to be randomized, and each request to an oracle has unit time cost. Some standard types of oracles are:

- *Query Oracle:* Given any query point $x \in \mathcal{X}$, the oracle responds with $f(x)$.
- *Sample Oracle:* Upon request, the oracle responds with an independently random sample point $x \sim \mathcal{D}$.
- *Labeled-Sample Oracle:* Upon request, the oracle responds with a pair $(x, f(x))$, where $x \sim \mathcal{D}$ is sampled independently from $\mathcal{D}$.

We will refer to algorithms with access to the query and sample oracle as operating in the *query model*, and refer algorithms with access to only the labeled-sample oracle as operating in the *labeled-sample model*. We will sometimes write query for the set $\mathcal{O}$ containing the query and sample oracle, and samp for the set $\mathcal{O}$ containing only the labeled-sample oracle.

For a set of oracles $\mathcal{O}$ and an algorithm A that operates in the $\mathcal{O}$-oracle model, the $\mathcal{O}$-oracle complexity of A is the supremum over all pairs $(f, \mathcal{D}) \in \mathcal{H} \times \mathfrak{D}$ of the number of requests to the oracles in $\mathcal{O}$ made by A on input $(f, \mathcal{D})$, as a function of ϵ. For oracle sets $\mathcal{O}$ that contain more than one type of oracle, we are also interested in the number of requests made to each oracle. For an algorithm in the query model, we say that the *query complexity* of A is the supremum over all $(f, \mathcal{D}) \in \mathcal{H} \times \mathfrak{D}$ of the number of requests made by A to the query oracle, and the *sample complexity* is similarly defined as the number of requests made by A to the sample oracle.

We may now define property testing and learning algorithms.

Definition 1.2.3 (Property Testing)**.** Let $\mathcal{X}$ be some domain, let $\mathcal{H}$ be a set of functions $\mathcal{X} \to \{0,1\}$, let $\mathfrak{D}$ be a set of probability distributions over $\mathcal{X}$, and let $\mathcal{O}$ be a set of oracles. An *$\mathcal{O}$-oracle tester for $\mathcal{H}$ under $\mathfrak{D}$* is an algorithm satisfying the following:

Input: $\epsilon > 0$ and $\mathcal{O}$-oracle access to $(f, \mathcal{D})$ where $f : \mathcal{X} \to \{0,1\}$ and $\mathcal{D} \in \mathfrak{D}$.

Output:

1. If $f \in \mathcal{H}$, the algorithm outputs Yes with probability at least $2/3$;
2. If $\mathsf{dist}_{\mathcal{D}}(f, \mathcal{H}) \geq \epsilon$, the algorithm outputs No with probability at least $2/3$.

For a set of oracles $\mathcal{O}$, a set of functions $\mathcal{H}$, and a class of distributions $\mathfrak{D}$, we will write $\mathsf{test}^{\mathcal{O}}_{\mathcal{H},\mathfrak{D}}(\epsilon)$ for the minimum $\mathcal{O}$-oracle complexity of an $\mathcal{O}$-oracle algorithm satisfying the above conditions. If we replace condition (1) with the following condition:

1′ If $f \in \mathcal{H}$, the algorithm outputs Yes with probability 1;

then we call the algorithm a *one-sided tester*, and we write $\mathsf{otest}^{\mathcal{O}}_{\mathcal{H},\mathfrak{D}}(\epsilon)$ for the minimum $\mathcal{O}$-oracle complexity of a one-sided $\mathcal{O}$-oracle tester satisfying the conditions. Finally, if our algorithm takes as input $0 \leq \epsilon_1 < \epsilon_2$ instead of ϵ and is required to satisfy the following conditions:

1* If $\mathsf{dist}_{\mathcal{D}}(f, \mathcal{H}) \leq \epsilon_1$, the algorithm outputs Yes with probability at least $2/3$;

2* If $\mathsf{dist}_{\mathcal{D}}(f, \mathcal{H}) \geq \epsilon_2$, the algorithm outputs No with probability at least $2/3$,

then we call the algorithm (ϵ_1, ϵ_2)-tolerant. We will write $\mathsf{test}^{\mathcal{O}}_{\mathcal{H},\mathfrak{D}}(\epsilon_1, \epsilon_2)$ for the minimum $\mathcal{O}$-oracle complexity of a (ϵ_1, ϵ_2)-tolerant $\mathcal{O}$-oracle tester satisfying these conditions. We remark that $\mathsf{test}^{\mathcal{O}}_{\mathcal{H},\mathfrak{D}}(0, \epsilon)$ is not necessarily equal to $\mathsf{test}^{\mathcal{O}}_{\mathcal{H},\mathfrak{D}}(\epsilon)$.

Definition 1.2.4 (Learning)**.** Let $\mathcal{X}$ be some domain, let $\mathcal{H}$ be a set of functions $\mathcal{X} \to \{0,1\}$, let $\mathfrak{D}$ be a set of probability distributions over $\mathcal{X}$, and let $\mathcal{O}$ be a set of oracles. An *$\mathcal{O}$-oracle improper agnostic learner for $\mathcal{H}$ under $\mathfrak{D}$* is an algorithm satisfying the following:

Input: Parameter $\epsilon > 0$, and oracle access to $(f, \mathcal{D})$, where $f : \mathcal{X} \to \{0,1\}$ and $\mathcal{D} \in \mathfrak{D}$.

Output: With probability at least $2/3$, a function $g : \mathcal{X} \to \{0,1\}$ that satisfies:

$$\mathsf{dist}_{\mathcal{D}}(f, g) \leq \mathsf{dist}_{\mathcal{D}}(f, \mathcal{H}) + \epsilon\,.$$

For a set of oracles $\mathcal{O}$, a set of functions $\mathcal{H}$, and a class of distributions $\mathfrak{D}$, we will write $\mathsf{alearn}^{\mathcal{O}}_{\mathcal{H},\mathfrak{D}}(\epsilon)$ for the minimum $\mathcal{O}$-oracle complexity of an $\mathcal{O}$-oracle algorithm satisfying the above conditions. If we replace condition (2) with the same condition, but add the requirement that $g \in \mathcal{H}$, then we call the algorithm a *proper agnostic learner*, and write $\mathsf{aplearn}^{\mathcal{O}}_{\mathcal{H},\mathfrak{D}}(\epsilon)$ for the optimal complexity of such an algorithm. Finally, if we change the input so that it has the promise $f \in \mathcal{H}$, we obtain a (non-agnostic) *improper learner* and a *proper learner*, respectively. We write $\mathsf{learn}^{\mathcal{O}}_{\mathcal{H},\mathfrak{D}}(\epsilon)$ and $\mathsf{plearn}^{\mathcal{O}}_{\mathcal{H},\mathfrak{D}}(\epsilon)$ for the optimal complexities of these types of algorithms.

For both learning and testing algorithms, the oracle set $\mathcal{O}$, the function class $\mathcal{H}$, and the class of input distributions $\mathfrak{D}$ will be usually be clear from context. In this case we drop the sub- and superscripts for these parameters. We will write $\mathsf{test}^{\mathsf{query}}$, $\mathsf{test}^{\mathsf{samp}}$, $\mathsf{plearn}^{\mathsf{query}}$, $\mathsf{plearn}^{\mathsf{samp}}$, etc., for the oracle complexities in the query and labeled-sample model, respectively.

1.2.3 Testing vs. Learning

Recall that there are two main questions we can ask about testing and learning algorithms. The first is the testing vs. learning question of [GGR98], which we now restate more formally:

Testing vs. Learning: *Which hypothesis classes $\mathcal{H}$, under which sets of distributions $\mathfrak{D}$ and which oracle models, can be tested more efficiently than they can be learned?*

The second is algorithm design: for a given hypothesis class $\mathcal{H}$ and set of distributions $\mathfrak{D}$, what is the most efficient tester or learner for $\mathcal{H}$ under $\mathfrak{D}$? Answering the second question, by giving tight bounds on both testing and learning a class $\mathcal{H}$, would answer the testing vs. learning question for that class. In this respect, if our goal is to understand both testing and learning algorithms, the testing vs. learning question is a prerequisite. Below, we introduce our results on testing vs. learning, from the paper [BFH21] coauthored with Eric Blais and Renato Ferreira Pinto Jr.; the details of these results are in Chapter 2. We return to designing efficient testers and learners in Section 1.2.4 and Chapter 3.

A simple but foundational result on the testing vs. learning question is that a (proper) learning algorithm can be used as a testing algorithm [GGR98]. This essentially formalizes the standard empirical practice of running the learning algorithm and checking its correctness on a test set. This formally justifies the testing vs. learning question, since we need not be concerned with the case where testing is *less* efficient than learning.

Theorem 1.2.5 ([GGR98])**.** *Let $\mathcal{H}$ be any hypothesis class of functions $\mathcal{X} \to \{0,1\}$, let $\mathfrak{D}$ be any set of probability distributions over $\mathcal{X}$, and let $\epsilon > 0$. Let $\mathcal{O}$ be a set of oracles that contains the labeled sample oracle. Then*

$$\mathsf{test}^{\mathcal{O}}(\epsilon) = O\left(\mathsf{plearn}^{\mathcal{O}}(\epsilon/2)\right) + O(1/\epsilon)\,.$$

We want to know when $\mathsf{test}^{\mathcal{O}}(\epsilon) \ll \mathsf{plearn}^{\mathcal{O}}(\epsilon)$. To approach a general theory of testing vs. learning, we should focus on:

1. The models of learning that are most well-understood; and,
2. The hypothesis classes and probability distributions that are either the most fundamental, or are the most practically useful.

The learning model that is most well-understood is the distribution-free labeled-sample model, known as the Probably Approximately Correct (PAC) learning model of Valiant [Val84]. In fact, for the purposes of this thesis, PAC learning is completely understood[4]. In this model, the algorithm has access only to the labeled-sample oracle, and the set $\mathfrak{D}$ of distributions is unrestricted[5]. In [GGR98], the authors "stress that [this model] is essential for some of the potential applications" listed in that paper, but despite much recent interest in both distribution-free and sample-based testing (e.g. [GS09, BBBY12, AHW16, GR16, CFSS17, BMR19, BY19, Har19, FY20, RR20]), even basic questions for this model remain unanswered.

Are there any testable classes? Having chosen the distribution-free labeled-sample model to study, we should look for the most natural hypothesis classes. But first, we should ask: in this model, are there *any* hypothesis classes for which $\mathsf{test}(\epsilon) \ll \mathsf{plearn}(\epsilon)$? Consider first the class $\mathcal{H}$ of *all* functions $\mathcal{X} \to \{0,1\}$. This is the hardest class to learn (indeed, when $\mathcal{X}$ is infinite, learning is impossible), but the easiest class to test: always output Yes.

[4]This is because we are not concerned with time complexity or constant factors, and we will treat ϵ as a small constant.

[5]Some technical restrictions on measurability are necessary, see [SB14].

A slightly less trivial example was given in [GGR98]: $\mathcal{H}$ is the class of functions $f : \{0,1\}^d \to \{0,1\}$ which takes value 1 on $x \in \{0,1\}^d$ if $x_1 = 1$ and is otherwise unrestricted. This class requires $\Omega(2^d)$ labeled samples to learn, but only $O(1/\epsilon)$ labeled samples to test. However, this class is not "natural", in the sense that it is not intrinsically interesting. Are there "natural" classes of functions that are easier to test than to learn? This thesis provides two: *monotone functions* (over an arbitrary partially ordered finite domain), the *juntas*, both of which are central to the literature on property testing.

Theorem 1.2.6. *Let $\mathcal{H}$ be the class of k-juntas on domain $\{0,1\}^d$, for any $k = k(d)$. Then $\mathsf{otest}^{\mathsf{samp}}_{\mathcal{H}}(\epsilon) = O\left(\frac{k2^{k/2}\log(d/k)}{\epsilon}\right)$, whereas for constant ϵ, $\mathsf{plearn}^{\mathsf{samp}}(\epsilon) = \Omega(2^k)$.*

Theorem 1.2.7. *Let $\mathcal{X}$ be any finite partial order with $n = |\mathcal{X}|$, and let $\mathcal{H}$ be the class of monotone functions $\mathcal{X} \to \{0,1\}$. Then $\mathsf{test}^{\mathsf{samp}}(\epsilon) = O\left(\sqrt{n}/\epsilon\right)$. On the other hand, $\mathsf{plearn}^{\mathsf{samp}}(\epsilon) = \Theta(\mathsf{width}(\mathcal{X})/\epsilon)$, where $\mathsf{width}(\mathcal{X})$ is the size of the largest antichain in $\mathcal{X}$.*

Testing Halfspaces. Having checked that some natural hypothesis classes do indeed satisfy $\mathsf{test}^{\mathsf{samp}}(\epsilon) \ll \mathsf{plearn}^{\mathsf{samp}}(\epsilon)$, so that our question is non-trivial, we should now identify the most important hypothesis classes to focus on. Arguably the most fundamental and practically useful hypothesis class is the class of *halfspaces*[6]. Recall from Section 1.2.1 that halfspaces (which are the class learned by SVMs) are used frequently in practice. They may also be the most fundamental: they are geometrically simple, they have been studied since the 1950s, and they are the class learned by the *perceptron*, a simplified model of a single neuron developed in the very early literature on neural networks. Therefore, a central goal for understanding testing vs. learning is to understand the complexity of *testing halfspaces* with the labeled-sample oracle, under arbitrary distributions.

Testing halfspaces has been studied under the Gaussian distribution and uniform distribution over $\{0,1\}^d$ (in the query model) [MORS10], the Gaussian distribution (in the labeled-sample model) [BBBY12], and under rotation-invariant distributions (in the labeled-sample model) [Har19]. Prior to this thesis, the best known lower bound was $\widetilde{\Omega}(\sqrt{d})$ for the standard Gaussian distribution over $\mathbb{R}^d$ [BBBY12], compared to $\Theta(d)$ for learning; the best known upper bound in the labeled-sample model is a matching $\widetilde{O}(\sqrt{d})$ under the class $\mathfrak{D}$ of all rotation-invariant distributions over $\mathbb{R}^d$ [Har19]. This left open the possibility that there is an efficient $\widetilde{O}(n^c)$-sample tester for some constant $c < 1$, even $c = 1/2$, in the general distribution-free setting. One of the main results of this thesis (from [BFH21]) is that

[6] One might argue that deep neural networks are the most practically useful hypothesis class. Testing this hypothesis class is beyond the scope of this thesis, and indeed a theoretical understanding of deep neural networks appears to be beyond the scope of modern science.

this is impossible. We essentially resolve the testing vs. learning question for (arguably) the most important special case: halfspaces in the labeled-sample model.

Theorem 1.2.8. *Let $\mathcal{H}$ be the set of halfspaces with domain $\mathbb{R}^d$ or $\{0,1\}^d$, let $\mathfrak{D}$ be the set of all probability distributions over the domain. Then for some constant $\epsilon > 0$,*

$$\mathsf{test}^{\mathsf{samp}}(\epsilon) = \Omega\left(\frac{d}{\log d}\right) = \widetilde{\Omega}\left(\mathsf{plearn}^{\mathsf{samp}}(\epsilon)\right) .$$

In follow-up work of Chen & Patel [CP22], our technique was extended to get a $\widetilde{\Omega}(d)$ lower bound for testing halfspaces even in the *query* model.

A General Theory. Although this resolves one of the central problems for understanding testing vs. learning, our goal is a more general theory. We present some progress towards this goal. Any theory of testing vs. learning must explain the relationship between testing and the *Vapnik-Chervonenkis (VC) dimension*. The labeled-sample complexity of learning a class $\mathcal{H}$ (i.e. the complexity of PAC learning) is determined entirely by the VC dimension of $\mathcal{H}$, which is defined as follows. A set $T \subseteq \mathcal{X}$ is *shattered* by $\mathcal{H}$ if for every $\ell : T \to \{0,1\}$ there is a function $f \in \mathcal{H}$ that agrees with ℓ on all points in T. The *VC dimension of $\mathcal{H}$ with respect to $S \subseteq \mathcal{X}$* is

$$\mathsf{VC}_S(\mathcal{H}) := \max\{k : \exists T \subseteq S \text{ of size } |T| = k \text{ that is shattered by } \mathcal{H}\}.$$

We will write $\mathsf{VC}(\mathcal{H}) = \mathsf{VC}_{\mathcal{X}}(\mathcal{H})$. The "fundamental theorem of PAC learning" is as follows (see e.g. [SB14] and [BHMZ20]).

Theorem 1.2.9 (Fundamental Theorem of PAC Learning)**.** *Let $\mathcal{H}$ be any set of functions $\mathcal{X} \to \{0,1\}$, let $\mathfrak{D}$ be the set of all distributions*[7] *over $\mathcal{X}$. Then*

$$\mathsf{plearn}^{\mathsf{samp}}(\epsilon) = \left\{O\left(\frac{\mathsf{VC}(\mathcal{H})}{\epsilon}\log\frac{1}{\epsilon}\right), \Omega\left(\frac{\mathsf{VC}(\mathcal{H})}{\epsilon}\right)\right\} \quad \mathsf{aplearn}^{\mathsf{samp}}(\epsilon) = \Theta\left(\frac{\mathsf{VC}(\mathcal{H})}{\epsilon^2}\right) .$$

This thesis makes progress towards understanding the relationship between testing and VC dimension, by defining a related quantity called the *lower VC* or *LVC dimension*, and relating the labeled-sample complexity of testing to the LVC and VC dimensions.

Informally, the LVC dimension of a class $\mathcal{H}$ quantifies the number of examples required to *prove* that a function $f : \mathcal{X} \to \{0,1\}$ satisfies $f \notin \mathcal{H}$, which is a fundamental quantity for testing with one-sided error. Intuitively, many natural hypothesis classes $\mathcal{H}$ (like

[7] These distributions must actually satisfy some technical measurability conditions, see [SB14].

halfspaces, intersections of halfspaces, polynomial threshold functions, decision trees, and others) satisfy the interesting property that to *prove* that $\mathcal{H}$ cannot approximate a target function f requires almost as many sample points as it takes to *learn* $\mathcal{H}$. We show that, whenever this is the case, it also requires nearly as many sample points to gather sufficient statistical *evidence* that $\mathcal{H}$ cannot approximate f.

Formally, for any subset $S \subseteq \mathcal{X}$, define the *LVC dimension* of $\mathcal{H}$ with respect to S as

$$\mathsf{LVC}_S(\mathcal{H}) := \max\{k : \forall T \subseteq S \text{ of size } |T| = k,\ T \text{ is shattered by } \mathcal{H}\}.$$

See Chapter 2 (Definition 2.1.2) for a discussion of this definition. Our main theorem on testing vs learning is the following:

Theorem 1.2.10. *There are constants $C, \epsilon_0 > 0$ such that the following holds. Let $\mathcal{H}$ be a set of functions $\mathcal{X} \to \{0,1\}$, let $\mathfrak{D}$ be the set of all distributions over $\mathcal{X}$, and let $S \subseteq \mathcal{X}$ satisfy $|S| \geq 5 \cdot \mathsf{VC}_S(\mathcal{H})$ and $\mathsf{LVC}_S(\mathcal{H}) \geq C \cdot \mathsf{VC}_S(\mathcal{H})^{3/4}\sqrt{\log \mathsf{VC}_S(\mathcal{H})}$. Then for all $\epsilon < \epsilon_0$,*

$$\mathsf{test}^{\mathsf{samp}}(\epsilon) = \Omega\left(\frac{\mathsf{LVC}_S^2(\mathcal{H})}{\mathsf{VC}_S(\mathcal{H}) \log \mathsf{VC}_S(\mathcal{H})}\right).$$

In particular, if $\mathsf{LVC}_S(\mathcal{H}) = \Omega(\mathsf{VC}(\mathcal{H}))$, then

$$\mathsf{test}^{\mathsf{samp}}(\epsilon) = \Omega\left(\frac{\mathsf{VC}(\mathcal{H})}{\log \mathsf{VC}(\mathcal{H})}\right).$$

This gives theoretical evidence that the often-cited motivation for property testing, to aid in model selection for learning algorithms, is essentially impossible in many practical cases, specifically when the input distribution is unstructured and when the hypothesis class has large LVC dimension. It is also interesting that, in general, the $\log \mathsf{VC}$ factor in the denominator is necessary: it follows from our lower bound and an upper bound of [GR16] that there exists a hypothesis class with $\mathsf{test}^{\mathsf{samp}}(\epsilon) = \Theta\left(\frac{\mathsf{VC}(\mathcal{H})}{\log \mathsf{VC}(\mathcal{H})}\right)$ (for constant ϵ). This logarithmic factor comes from the *support-size estimation* problem (see [VV11a, WY19]), which is the main ingredient in our proof, and where a logarithmic improvement is considered significant.

Using our general theorem, we obtain nearly optimal impossibility results for a wide range of hypothesis classes of fundamental importance to learning theory. A summary of these results is given in Table 1.1. These results, formal definitions, and proofs are given in detail in Chapter 2.

Domain	Class $\mathcal{H}$	$\mathsf{test}^{\mathsf{samp}}(\epsilon)$	$\mathsf{VC}(\mathcal{H})$
$[n]$ or $\mathbb{R}$	Unions of k intervals	$\Omega\left(\frac{k}{\log k}\right)$	$\Theta(k)$
$\mathbb{R}^n$	Halfspaces	$\Omega\left(\frac{n}{\log n}\right)$	$\Theta(n)$
	Intersections of k halfspaces	$\Omega\left(\frac{nk}{\log(nk)}\right)$	$\Theta(nk \log k)$
	Degree-k PTFs over $\mathbb{R}^n$	$\Omega\left(\frac{\binom{n+k}{k}}{\log\binom{n+k}{k}}\right)$	$\Theta\left(\binom{n+k}{k}\right)$
	Size-k decision trees	$\Omega\left(\frac{k}{\log k}\right)$	$\Omega(k)$
$\{0,1\}^n$	Halfspaces	$\Omega\left(\frac{n}{\log n}\right)$	$\Theta(n)$
	Degree-k PTFs	$\Omega\left(\frac{(n/4ek)^k}{k\log(n/k)}\right)$	$\Theta\left(\binom{n}{\leq k}\right)$
	Size-k decision trees	$\Omega\left(\frac{k}{\log k \cdot \log\log k}\right)$	$\Omega(k), O(k \log n)$

Table 1.1: Summary of lower bounds in Chapter 2. See the citations in that chapter for calculations of the VC dimension.

Unions of Intervals (Section 2.3.1). Balcan, Blais, Blum, & Yang [BBBY12] (see also [KR00, Nee14]) showed that there is an algorithm that can test unions of k intervals over *any* distribution on $[0,1]$ with only $O(\sqrt{k})$ samples — as long as the distribution is known to the algorithm. Our lower bound for this class shows that the sample must be quadratically larger if the distribution is not known to the algorithm.

Our bound also has implications for the *active testing* model [BBBY12], where a tester can draw some unlabelled samples from the unknown distribution $\mathcal{D}$ and then query the value of the target function on any of the sampled points. Blum and Hu [BH18] showed that it is possible to tolerantly test unions of k intervals in this model with $O(k)$ samples and $O(1)$ queries. Theorem 2.3.2 implies that $\widetilde{\Omega}(k)$ samples are necessary, even for intolerant active testers (regardless of how many samples are queried), so their result is essentially optimal.

Intersections of Halfspaces & PTFs (Sections 2.3.3, 2.4.2 and 2.5.2). Intersections of halfspaces [BEHW89, CMK19] and polynomial threshold functions [KS04, HS07, DHK+10, OS10] have received much attention in the learning theory literature, but very few bounds are known on the sample or query complexity for testing these classes. As far

as we know, the only bound known for testing intersections of k halfspaces is an upper bound of $\exp(k \log k)$ queries for testing the class over the Gaussian distribution [DMN19] and no bound is known for testing polynomial threshold functions of degree greater than 1. So our results appear to establish the first non-trivial lower bounds specific for either of these classes in any model of property testing.

Decision Trees (Sections 2.3.4 and 2.4.3). Kearns and Ron [KR00] first studied the problem of testing size-k decision trees, showing that $\Omega(\sqrt{k})$ samples are necessary to test the class over the uniform distribution and that this bound can be matched in the parameterized property testing model where the algorithm must only distinguish size-k decision trees from functions that are far from size-k' decision trees over the uniform distribution for some $k' > k$. The sample complexity of the (non-parameterized) size-k decision tree testing problem over the uniform distribution is not known. (The query complexity for testing size-k decision trees is also far from settled: despite recent notes to the contrary in [Sağ18, Bsh20], the best current lower bound for the query complexity of testing size-k decision trees is $\Omega(\log k)$ [CGM11, Tan20]; see also [BBM12] for a stronger lower bound for testers with one-sided error.)

Other Models of Testing (Section 2.6). Our techniques can also be used to establish lower bounds for other models of testing. First, we show an application to testing *radius clustering*, a problem introduced by Alon *et al.* [ADPR03]. Here we define the class $\mathcal{C}_k$ of all sets of points $X \subseteq \mathbb{R}^n$ that can be covered by the union of at most k unit-radius balls. A distribution $\mathcal{D}$ on $\mathbb{R}^n$ is *k-clusterable* if its support is in $\mathcal{C}_k$, and it is *ϵ-far from k-clusterable* if the total variation distance between $\mathcal{D}$ and any k-clusterable distribution is at least ϵ. Alon *et al.* [ADPR03] prove an upper bound of $O\left(\frac{nk\log(nk)}{\epsilon}\right)$ samples for one-sided testing of k-clusterability when the distribution is uniform over an unknown set of points. We present an improved bound of $O\left(\frac{nk\log k}{\epsilon}\log\frac{1}{\epsilon}\right)$ that follows from modern VC dimension results.

Prior to this work, the only lower bound for the sample complexity of this problem was Epstein and Silwal's recent lower bound of $\Omega(n/\epsilon)$ samples for ϵ-testing 1-clusterability with one-sided error [ES20]. We give a lower bound for two-sided error testers that is tight up to poly-log factors.

We also obtain lower bounds on testing feasibility of linear programs, in the model defined recently by Epstein & Silwal [ES20]. Here, the algorithm sees a random subset of linear constraints for an LP and must decide whether the program is feasible, or whether at least an ϵ fraction of the constraints must be removed to achieve feasibility. We obtain

	$\mathsf{unif}(\{\pm 1\}^d)$	$\mathsf{unif}([n]^d)$	Gaussian	$\forall$ Products
1-Sided Testing Monotonicity (Query model)	$\widetilde{O}\left(\frac{\sqrt{d}}{\epsilon^2}\right)$ [KMS18]	$\widetilde{O}\left(\frac{d^{5/6}}{\epsilon^{4/3}}\right)$ [BCS20]	$\widetilde{O}\left(\frac{d^{5/6}}{\epsilon^{4/3}}\right)$ [BCS20]	$\widetilde{O}\left(\frac{d^{5/6}}{\epsilon^{4/3}}\right)$ queries, $\widetilde{O}\left(\left(\frac{d}{\epsilon}\right)^3\right)$ samples (Thm. 1.2.11)
1-Sided Testing Convex Sets (Sample model)	–	–	$\left(\frac{d}{\epsilon}\right)^{(1+o(1))d}$ $2^{\Omega(d)}$ [CFSS17]	$\left(\frac{d}{\epsilon}\right)^{(1+o(1))d}$ (Thm. 3.4.1)
Tolerant Testing Functions of k Convex Sets (Sample model)	–	–	–	$\left(\frac{dk}{\epsilon}\right)^{O(d)}$ (Thm. 3.4.3)
Tolerant Testing k-Alternating Functions (Sample model)	–	$\left(\frac{dk}{\tau}\right)^{O\left(\frac{k\sqrt{d}}{\tau^2}\right)}$ $\tau = \epsilon_2 - 3\epsilon_1$ [CGG+19]	–	$\left(\frac{dk}{\tau}\right)^{O\left(\frac{k\sqrt{d}}{\tau^2}\right)}$ $\tau = \epsilon_2 - \epsilon_1$ (Thm. 3.7.2)

Table 1.2: Efficient testing algorithms. Empty cells mean there are no prior results.

lower bounds on two-sided error testers for these problems, whereas [ES20] gave lower bounds for one-sided error.

1.2.4 Designing Efficient Algorithms

We now turn our attention to the second major question about testing and learning algorithms: for a given hypothesis class $\mathcal{H}$ and class of distributions $\mathfrak{D}$, what is the most efficient tester or learner for $\mathcal{H}$ under $\mathfrak{D}$?

For any class $\mathcal{H}$, the goal is to obtain testers or learners which not only minimize the number of requests to the oracles, but also that work for the least restricted class of distributions $\mathfrak{D}$. A standard approach to designing such algorithms is to choose a simple class $\mathfrak{D}$, like the uniform distribution over $\{0,1\}^d$ or the standard Gaussian distribution over $\mathbb{R}^d$, design an efficient algorithm for testing or learning $\mathcal{H}$ under this restricted class $\mathfrak{D}$, and then attempt to generalize.

Most "simple" distributions in $\mathbb{R}^d$ are *product distributions*, where the coordinates of a random sample $x \sim \mathcal{D}$ are independent random variables; for example, the uniform distribution over $\{0,1\}^d$ or $[n]^d$ and the standard Gaussian distribution over $\mathbb{R}^d$ are all product distributions. So a natural step is to take an algorithm for one of these simple distributions and generalize it to work for the class $\mathfrak{D}$ of all product distributions over $\mathbb{R}^d$.

	$\mathsf{unif}(\{\pm 1\}^d)$	$\mathsf{unif}([n]^d)$	Gaussian	$\forall$ Products
Functions of k Convex Sets	$\Omega(2^d)$	–	$d^{O\left(\frac{\sqrt{d}}{\epsilon^4}\right)}, 2^{\Omega(\sqrt{d})}$ [KOS08]	$O\left(\frac{1}{\epsilon^2}\left(\frac{6dk}{\epsilon}\right)^d\right)$ (Thm. 3.4.4)
Functions of k Halfspaces	$d^{O\left(\frac{k^2}{\epsilon^4}\right)}$ [KKMS08]	$(dn)^{O\left(\frac{k^2}{\epsilon^4}\right)}$ [BOW10]	$d^{O\left(\frac{\log k}{\epsilon^4}\right)}$, $\mathrm{poly}\left(d, \left(\frac{k}{\epsilon}\right)^k\right)$ [KOS08, Vem10a] (Intersections only)	$\left(\frac{dk}{\epsilon}\right)^{O\left(\frac{k^2}{\epsilon^4}\right)}$ (Thm. 1.2.12)
Degree-k PTFs	$d^{\psi(k,\epsilon)}$ [DHK+10]	$(dn)^{\psi(k,\epsilon)}$ [DHK+10, BOW10]	$d^{\psi(k,\epsilon)}$ [DHK+10, BOW10]	$\left(\frac{dk}{\epsilon}\right)^{\psi(k,\epsilon)}$ (Thm. 1.2.13)
k-Alternating Functions	$2^{\Theta\left(\frac{k\sqrt{d}}{\epsilon}\right)}$ [BCO+15]	$\left(\frac{dk}{\tau}\right)^{O\left(\frac{k\sqrt{d}}{\tau^2}\right)}$ (Testing) [CGG+19]	–	$\left(\frac{dk}{\epsilon}\right)^{O\left(\frac{k\sqrt{d}}{\epsilon^2}\right)}$ (Thm. 3.7.2)

Table 1.3: Efficient learning algorithms. Empty cells mean there are no prior results. All algorithms are agnostic except that of [Vem10a]. The PTF result for the Gaussian follows from the two cited works but is not stated in either. All statements are informal, see references for restrictions and qualifications. For PTFs, $\psi(k,\epsilon) := \min\left\{O(\epsilon^{-2^{k+1}}), 2^{O(k^2)}\left(\log(1/\epsilon)/\epsilon^2\right)^{4k+2}\right\}$.

Chapter 3 presents a simple and general algorithm design method, called *downsampling*, for accomplishing this step of generalization (which appears in the paper [HY21], coauthored with Yuichi Yoshida). Using downsampling we obtain several new algorithms whose complexity is significantly better than the previous state-of-the-art, for a number of important testing and learning problems like learning intersections of halfspaces or polynomial threshold functions, and testing monotonicity. Furthermore, the downsampling technique is conceptually simple and allows short, clear proofs. For example, we improve upon the recent monotonicity tester of Black, Chakrabarty, & Seshadhri [BCS20] while shortening the proof from ~ 25 pages to 2. We briefly discuss these results here, and give a detailed treatment in Chapter 3. See Table 1.2 for a summary of property testing results, and Table 1.3 for a summary of learning results.

Testing Algorithms. One of the most well-studied problems in the property testing literature is *testing monotonicity*; we already discussed a result of this thesis on testing monotonicity in the previous section. Not only is monotonicity a simple and natural property, but testing monotonicity has also revealed many interesting connections to graph theory, such as the construction of Ruzsá-Szemerédi graphs [FLN+02] and directed isoperimetry [CS16, KMS18]. As discussed in the previous section, monotonicity is, in general, more efficiently testable than learnable; the question is, just how efficiently is it testable, given various restrictions on the domain and probability distribution?

Previous work on this problem has mostly focused on uniform probability distributions (exceptions include [AC06, HK07, CDJS17, BFH21]) and finite domains. Most progress on this question has been for the hypergrids $[n]^d$ endowed with the natural partial order, and with Boolean-valued functions. The case $[2]^d$ is well understood: there is a non-adaptive $\widetilde{O}(\sqrt{d}/\epsilon^2)$-query tester due to Khot, Minzer, & Safra [KMS18] (following earlier sublinear upper bounds [CS16, CST14]), a matching non-adaptive lower bound of $\widetilde{\Omega}(\sqrt{d})$ queries due to Chen, Waingarten, & Xie [CWX17] (following earlier lower bounds [FLN+02, CDST15]), and an *adaptive* lower bound of $\widetilde{\Omega}(d^{1/3})$, also due to Chen, Waingarten, & Xie [CWX17] (following an earlier bound of Belovs & Blais [BB16]). The general case of $[n]^d$ is less understood. Ailon & Chazelle [AC06] gave a monotonicity tester for *real-valued* functions under product distributions on $[n]^d$, with query complexity $O(\frac{1}{\epsilon}d2^d \log n)$. Chakrabarty, Dixit, Jha, & Seshadhri [CDJS17] improved this to $O(\frac{1}{\epsilon}d \log n)$ and gave a matching lower bound that applies to real-valued (but not Boolean-valued) functions.

For Boolean-valued functions, Black, Chakrabarty, & Seshadhri [BCS20] recently gave an upper bound of $\widetilde{O}(d^{5/6}/\epsilon^{4/3})$ and, following the general plan outlined above, showed how to generalize the algorithm to work for the set $\mathfrak{D}$ of product distributions over $\mathbb{R}^d$. Their algorithm for product distributions uses $\widetilde{O}(d^{5/6}/\epsilon^{4/3})$ queries and $O((d/\epsilon)^7)$ samples. In this thesis, we obtain a simpler tester for $[n]^d$ with the same query complexity, and improve the result for product distributions while greatly simplifying the proof.

Theorem 1.2.11. *There is a one-sided non-adaptive tester for monotonicity under product distributions over $\mathbb{R}^d$, with query complexity $\widetilde{O}(d^{5/6}/\epsilon^{4/3})$ and sample complexity $\widetilde{O}((d/\epsilon)^3)$.*

Our downsampling technique also easily yields new results for testing convex sets. As shown in Table 1.2, we get a one-sided error tester in the sample model, under product distributions, that matches the complexity of the tester of [CFSS17] for the Gaussian, and we get a tolerant tester for functions of k convex sets.

Learning Algorithms. For learning algorithms, it is important to consider the time complexity, not just the oracle complexity[8]. For example, even though the class of intersections of two halfspaces in $\mathbb{R}^d$ has an upper bound of $O(d)$ on the VC dimension [CMK19], and therefore an upper bound on the sample complexity, there is no efficient algorithm for producing an intersection of two halfspaces that agrees with the sample unless $\mathsf{P} = \mathsf{NP}$ [BR92]. Finding efficient *improper* learning algorithms for intersections of halfspaces is a major open problem [DKS18, KOS04].

There is a similar barrier for agnostically learning degree-k polynomial threshold functions (PTFs). Degree-k PTFs in $\mathbb{R}^d$ have VC dimension $\Theta\left(\binom{d+k}{k}\right) = O((d+k)^k)$, and therefore there is an upper bound of $O((d+k)^k/\epsilon^2)$ on the number of samples required for *agnostic* learning (see [SB14]). But it is hard (assuming $\mathsf{P} \neq \mathsf{NP}$ or the unique games conjecture) to agnostically and properly learn PTFs under general distributions [DOSW11]. It is therefore important to find efficient improper agnostic learning algorithms under restricted distributions.

Prior work on these two problems has shown that there are efficient algorithms for certain fixed distributions, like the uniform distribution over $\{\pm 1\}^d$ [KKMS08, DHK+10] or the Gaussian [KOS08, Vem10a]. A natural step is to generalize these algorithms to the hypergrid $[n]^d$ and then to arbitrary product distributions; this was done by Blais, O'Donnell, & Wimmer [BOW10] but the algorithm's complexity depended on n and therefore the generalization to product distributions could only be done under some restrictions. In this thesis, we use downsampling (which is a very different technique from that of [BOW10]) to eliminate the dependence on n and the restrictions on the product distribution. Below, we write $\mathcal{H}$ for the class of halfspaces and $\mathcal{B}_k$ for the class of functions $f(x) = g(h_1(x), \ldots, h_k(x))$ where $g : \{\pm 1\}^k \to \{\pm 1\}$ is arbitrary and each $h_i \in \mathcal{H}$ is a halfspaces (so that intersections of two halfspaces are in $\mathcal{B}_2 \circ \mathcal{H}$). See Table 1.3 for a comparison to prior results, and note that our results match the exponents of the earlier results and eliminate the dependence on n.

Theorem 1.2.12. *There is a distribution-free, improper agnostic learning algorithm for $\mathcal{B}_k \circ \mathcal{H}$ under (continuous or finite) product distributions over $\mathbb{R}^d$, with time complexity*

$$\min\left\{\left(\frac{dk}{\epsilon}\right)^{O\left(\frac{k^2}{\epsilon^4}\right)}, O\left(\frac{1}{\epsilon^2}\left(\frac{3dk}{\epsilon}\right)^d\right)\right\}.$$

[8] In the property testing literature, it is common to ignore the time complexity, since it is often the case that the time complexity is not much higher than the oracle complexity. But this assumption is not always justified.

Theorem 1.2.13. *There is an improper agnostic learning algorithm for degree-k PTFs under (finite or continuous) product distributions over $\mathbb{R}^d$, with time complexity*

$$\min\left\{\left(\frac{kd}{\epsilon}\right)^{\psi(k,\epsilon)}, O\left(\frac{1}{\epsilon^2}\left(\frac{9dk}{\epsilon}\right)^d\right)\right\}.$$

As seen in Table 1.3, we also get new learning results for arbitrary functions of k convex sets (where the prior work only studied a single convex set) and obtain a nearly optimal exponent; and k-alternating functions, where we again nearly match the optimal exponent for the uniform distribution over the hypercube $\{\pm 1\}^d$. All of our results are agnostic learners. Our proofs show that the powerful *polynomial regression* technique of [KKMS08] can be generalized to work for product distributions.

1.3 The Sketching and Communication Model

The first part of this thesis studied the oracle model, where the input is held by an oracle (or set of oracles), who responds to certain types of requests. The second part of this thesis studies a model where both the input and the algorithm is split among several parties. We think of the input as being too large to share completely, so decisions must again be made with incomplete information, although as algorithm designers we now have control over all parties. There are many models for studying problems like this (e.g. in distributed computing); we will study *communication complexity*, *sketching*, and *informative graph labeling*.

Understanding the power of randomness in communication is one of the main goals in communication complexity (see e.g. [CLV19, PSW20, HHH21b]). A simple but fundamental result in communication complexity is that two parties, Alice and Bob, can check whether their inputs x and y are identical using only a *constant number* of bits of communication, regardless of the size of x and y, when they share a source of randomness. This is known as the Equality function, and it is the standard example of the power of randomness in communication [NK96, RY20]. Although this "shared source of randomness" seems like an unrealistic assumption, this technique is extremely common in practice. It is standard to use a hash function like SHA256, which produces a constant-size hash value independent of file size, to check whether two files are the same. One motivation of the second part of this thesis is to understand which other decision problems can be similarly compressed to constant-size messages. This problem has also been studied concurrently and independently in another line of work by others [HHH21b, CHZZ22, HHP+22].

One of the conceptual contributions of this thesis is to relate this problem to a major open problem in structural graph theory and distributed computing, about *adjacency labeling schemes* for graphs. For a class $\mathcal{F}$ of graphs, an adjacency labeling scheme (introduced by Kannan, Naor, & Rudich [KNR92] and Muller [Mul89], consists of a decoder algorithm D, such that for any $G \in \mathcal{F}$ there is a function $\ell : V(G) \to \{0,1\}^*$ (which we call a *labeling*) that allows adjacency between any two vertices x and y to be decided by D using only $\ell(x)$ and $\ell(y)$. More generally, *informative labeling schemes*, introduced by Peleg [Pel05], allow the decoder to decide some other "local" information (like distance) about x and y. The main open problem in this area is what we will call the *Implicit Graph Question*, which asks which (hereditary) classes of graphs admit efficient adjacency labels. We introduce this question formally in Section 1.3.1.

This thesis initiates the thorough study of *randomized* labeling schemes, which we call *sketches*, where the labels are assigned by a randomized algorithm, and the decoder must be correct with probability at least 2/3; we will define these formally in Section 1.3.2. We will see that constant-cost communication problems are equivalent to constant-size randomized adjacency labeling schemes (i.e. *adjacency sketches*). Furthermore, constant-size randomized labeling schemes can be easily derandomized to obtain efficient *deterministic* labeling schemes. In this way, we consider the problem of determining the constant-cost randomized communication problems to be the probabilistic version of the Implicit Graph Question. We introduce these concepts formally in Section 1.3.2. Lest the reader suspect that we have diverged too far from Part I of this thesis, we note that constant-cost communication problems also correspond to problems which can be represented as halfspaces with constant *margin*, and can therefore be learned efficiently using the classic perceptron algorithm (see Remark 1.3.12).

We have chosen the term *sketch* to emphasize the similarity of these randomized labeling schemes to the previously-studied sketching problems in the field of sublinear algorithms. Labeling schemes and (the more common notion of) sketches are important primitives for distributed computing, streaming, communication, data structures for approximate nearest neighbors, and even classical algorithms (see e.g. [KNR92, GP03, Spi03, Pel05, EIX22], and [AMS99, Ind06, AK08, Raz17, AKR18] and references therein). As such, a great deal of research has been done on finding other spaces having nice sketching and labeling properties. Labeling schemes for adjacency are the most commonly-studied, and the next most well-studied labeling schemes are for distance (e.g. [GPPR04, GL07a, GP08, WP11, AGHP16b, AGHP16a, FGNW17] and approximate distance (e.g. [Pel00, GKK+01, Tho04, TZ05]).

The most well-studied problem in *sketching* is to identify metric spaces which admit approximate distance threshold (ADT) sketches, as defined in [SS02]. Here, n points

$X \subseteq \mathcal{X}$ in a metric space $(\mathcal{X}, \mathsf{dist})$, should be assigned random sketches $\mathsf{sk} : X \to \{0,1\}^*$ such that $\mathsf{dist}(x,y) \leq r$ or $\mathsf{dist}(x,y) \geq \alpha r$ can be determined (with high probability) from $\mathsf{sk}(x)$ and $\mathsf{sk}(y)$. The goal is to obtain sketches of constant size (independent of n). Determining which metric spaces have such sketches is a well-known open problem in sublinear algorithms (see e.g. [AK08, Jay09, Raz17]).

This thesis studies sketches for adjacency, distance, and approximate distance in graphs. We will formally relate informative labeling to communication complexity and sketching; surprisingly, the only prior work making this observation appears to be the unpublished manuscript of Andoni & Krauthgamer [AK08] which we learned of recently. The only prior work on randomized labeling schemes is a paper of Fraigniaud & Korman [FK09].

In Section 1.3.1, we will introduce the adjacency labeling problem. In Section 1.3.2, we will introduce the probabilistic version, graph sketching. We then proceed in Section 1.3.4 and Section 1.3.5 to discuss our results.

1.3.1 Graph Labeling and Implicit Representations

A *marked*[9] graph on n vertices is a graph G with vertex set $[n]$, where two marked graphs on $[n]$ are *equal* if they have the same edge set $E \subseteq [n] \times [n]$ (i.e. they may be isomorphic without being equal). A *class* $\mathcal{F}$ of graphs is a set of marked graphs that is closed under isomorphism. The function $n \mapsto |\mathcal{F}_n|$ which counts the number of n-vertex graphs is called the *speed* (and observe that two isomorphic graphs are counted separately unless they are also equal).

Consider the problem of efficiently representing a graph. The two most well-known graph representations are the adjacency matrix and the adjacency list. A graph may also be represented *implicitly*, where for each $n \in \mathbb{N}$ there is an abstract space $\mathcal{X}_n$ and a symmetric relation $\mathcal{R}_n \subseteq \mathcal{X}_n \times \mathcal{X}_n$, where an n-vertex graph is obtained by choosing n points $x \in \mathcal{X}_n$ and putting an edge xy when $(x,y) \in \mathcal{R}_n$. For example, an *interval graph* is one which can be represented by taking the space $\mathcal{X}_n$ to be the set of all intervals in $\mathbb{R}$, where $\mathcal{R}_n$ relates two intervals if and only if they intersect. A *unit disk graph* is one where each vertex may be associated with a point in $\mathbb{R}^2$ such that two vertices are adjacent if and only if their points have Euclidean distance at most 1.

Unfortunately, in these examples, vertices are associated with elements of a continuous space: it may not be possible to actually *write them down* (or store them in a computer).

[9] The standard terminology is to call these graphs *labeled*, but this is not to be confused with *graph labeling schemes*.

We would like to replace the space $(\mathcal{X}_n, \mathcal{R}_n)$ that defines our class with a *finite* set, as small as possible. In particular, we want to replace $\mathcal{X}_n$ with a set of (short) binary strings, so that we may represent vertices of our graphs with a small number of bits. We formalize this as follows.

A class of graphs $\mathcal{F}$ admits an *adjacency labeling* of size $s(n)$ if there is a relation $D : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$ over binary strings such that for every $G \in \mathcal{F}$ on n vertices, there is a labeling $\ell : V(G) \to \{0,1\}^{s(n)}$ where

$$\begin{aligned} \forall x, y \in V(G): \qquad xy \in E(G) &\implies D(\ell(x), \ell(y)) = 1 \\ xy \notin E(G) &\implies D(\ell(x), \ell(y)) = 0\,. \end{aligned}$$

Here, D is a relation over the space of binary strings, and each graph is implicitly represented as a set of n points in this space, each labeled by a binary string of length $s(n)$. Without a small bound on the size, every graph admits such a representation (see examples below). These types of size-bounded implicit representations were introduced by Kannan, Naor, & Rudich [KNR92] and Muller [Mul89]. They are useful because "local" information – adjacency between two vertices – can be decided from their labels, without requiring "global" information about the rest of the graph structure. These labels can be easily distributed among many parties so that they can determine adjacency without knowing the whole graph. Consider these examples, from [KNR92] and [Spi03]:

Example 1.3.1. Recall the unit disk graphs, where every vertex can be associated with a point in $\mathbb{R}^d$ so that vertices are adjacent if and only if they have Euclidean distance at most 1. The natural encoding is to simply convert the coordinates to binary representation. But McDiarmid & Müller showed that this requires $2^{\Omega(n)}$ bits [MM13].

Example 1.3.2. The class of all graphs admits an adjacency labeling of size $\lceil \log n \rceil + n$. Label each vertex $x \in [n]$ of a graph G with the name of x, along with its row in the adjacency matrix of G. Define D as the algorithm which checks if two vertices x, y are adjacent by checking the appropriate entry of the adjacency matrix included in the labels. This can be shortened to $\lceil \log n \rceil + n/2$ by including only the even elements or odd elements of the adjacency matrix, depending on the parity of the name of the vertex.

Example 1.3.3. The class of interval graphs admits an adjacency labeling of size $2\lceil \log(2n) \rceil$, because for an interval graph on n vertices, we may assign a number in $[2n]$ to the endpoints of the n intervals according to their natural ordering, and label each vertex using its two endpoints.

Example 1.3.4. The class of degree 3 graphs admits an adjacency labeling of size $4\lceil \log n \rceil$, since we can use the name of each vertex along with its adjacency list as the label. Improved labelings for bounded-degree graphs have been studied in [ELO08, But09, AR14, AAH+17].

Example 1.3.5. The class of forests has an adjacency labeling of size $2\lceil \log n \rceil$, since we can assume that each component tree is rooted, and label each vertex with its own name and the name of its parent. More generally, a graph has *degeneracy* k if there is a total order on its vertices so that each vertex has at most k neighbors larger than it in the ordering; a forest has degeneracy 1. The class of degeneracy k graphs has an adjacency labeling of size $(1+k)\lceil \log n \rceil$, where we simply use the adjacency list for each vertex, pruned to include only the vertices later in the ordering. Improved labelings for trees and forests have been studied in [Chu90, AR02, FK10, ADK17].

Example 1.3.6. Planar graphs have adjacency labels of size $5\lceil \log n \rceil$ because they have degeneracy 4 (and labels of size $4\lceil \log n \rceil$ using their *arboricity* instead of degeneracy). This was one of the main observations of [Mul89, KNR92], and it has been improved to $\log n + o(\log n)$ after significant effort [GL07b, BGP20, DEG+21, GJ22].

The latter four examples have an interesting property: the adjacency labels are of size $O(\log n)$, only a constant-factor larger than what is required to give each vertex a unique label. The main open problem in graph labeling, posed in [KNR92], is to identify the graph classes which admit such an efficient adjacency labeling. It is usually assumed that the graph classes are *hereditary*: a hereditary class is one that is closed under taking induced subgraphs. This is a standard assumption on a graph class (see e. g. [NO12]) and it is quite natural: it should be the case that the n-vertex graphs of a class $\mathcal{F}$ are somehow related to the N-vertex graphs, for $n < N$. Without the hereditary structure, one could imagine the class of graphs obtained by taking an arbitrary $\sqrt{\log n}$-vertex graph G and adding $n - \sqrt{\log n}$ independent vertices to obtain an n-vertex graph. This class has an $O(\log n)$ adjacency labeling but it is not due to any interesting properties. We then have the formal statement of the Implicit Graph Question:

Question 1.3.7 (Implicit Graph Question)**.** *Which hereditary classes of graphs admit an adjacency labeling of size* $O(\log n)$*?*

This question has received significant attention; see the references above and e. g. [Spi03, ACLZ15] and the references therein. [KNR92] observed that a necessary condition is that any such class must have at most $|\mathcal{F}_n| = 2^{O(n \log n)}$ graphs on n vertices (because the graphs can be encoded using the $O(n \log n)$ total bits in the labels assigned to each vertex). It was suggested in [KNR92] and conjectured in [Spi03] that this is also a sufficient condition. This was often called the Implicit Graph Conjecture in the literature. It was refuted by Hatami & Hatami [HH21] shortly after they learned of the problem and its connection to communication complexity from the paper [HWZ22] presented in this thesis. This leaves the question wide-open (and in need of a new name).

A further motivation for this question from the perspective of communication complexity (given in [Har20]), is that adjacency labeling for a class $\mathcal{F}$ is a generalization of *simultaneous message passing (SMP)* communication. In the standard SMP model, Alice and Bob receive inputs x and y to a (Boolean) function $f(x, y)$. They each send one message to a third-party referee who must output $f(x, y)$. It is important in this model that all three parties are given the function f in advance. We arrive at the adjacency labeling problem by removing this assumption: instead, Alice and Bob are given f which belongs to some class $\mathcal{F}$, and the third-party referee knows the general class $\mathcal{F}$ but not the specific function f. The general upper bound of $\lceil \log n \rceil$ on communication in the SMP model (where n is the size of the input domain) no longer holds when the referee is not given f in advance, and the question is how much extra information must be included in the messages to compensate. The Implicit Graph Question asks when this ignorance on the part of the referee increases the complexity by at most a constant factor.

This thesis is mainly concerned with the probabilistic version of this question, introduced formally in the next section. However, as stated earlier, these questions are formally related by the observation that randomized labels can be derandomized to obtain adjacency labels of size $O(\log n)$. We use this relationship to resolve some problems from the recent literature on adjacency labeling, and we extend these techniques to design optimal adjacency labeling schemes for subgraphs and induced subgraphs of Cartesian products, resolving the Implicit Graph Question in these cases and improving upon the best-known bounds of Chepoi, Labourel, & Ratel [CLR20].

1.3.2 Graph Sketching

Our study of graph sketching originated with the observation in the previous section that graph labeling is a generalization of SMP communication. It then becomes natural to consider the randomized version of this model, which corresponds to designing randomized labeling schemes, and we were particularly motivated by a few simple observations derived from the examples above.

Recall Example 1.3.4 (max-degree 3 graphs) and Example 1.3.5 (bounded degeneracy graphs). These graph classes have $O(\log n)$-bit adjacency labels, which are just the pruned adjacency lists. One can simply replace the pointers (vertex names) in these lists with *hashes*, as in the EQUALITY communication protocol, to get *constant-size* sketches, from which adjacency can be computed with high probability. On the other hand, Example 1.3.3 (interval graphs) used a different technique to get $O(\log n)$ adjacency labels: it was required to compare integers. This *cannot* be turned into a constant-size sketch: this would imply

a constant-cost randomized communication protocol for the Greater-Than problem, which cannot exist (see Appendix B).

These observations should be followed by recalling these results about hypercube:

1. Adjacency in the hypercube can be computed (with high probability) from sketches of constant size (which follows from the Hamming distance communication protocol [HSZZ06], see the simple exposition given in Section 4.1);
2. Distinguishing between $\mathsf{dist}(x,y) \leq r$ and $\mathsf{dist}(x,y) > r$ can be done with sketches of size depending only on r (which also follows from the Hamming distance protocol, see Chapter 4);
3. Distinguishing between $\mathsf{dist}(x,y) \leq r$ and $\mathsf{dist}(x,y) > \alpha r$ (for constant $\alpha > 1$) can be done with sketches of size independent of r and n [KOR00].

These recollections raise the question of which classes of graphs have similarly efficient sketches. We now formally define these three graph sketching problems. For a graph class $\mathcal{F}$, we say:

1. $\mathcal{F}$ admits an *adjacency sketch of size* $s(n)$ if there is a function $D : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$ such that $\forall G \in \mathcal{F}$ on n vertices, there is a random function $\mathsf{sk} : V(G) \to \{0,1\}^{s(n)}$ satisfying

$$\forall x, y \in V(G): \qquad \mathbb{P}\left[D(\mathsf{sk}(x), \mathsf{sk}(y)) = 1 \iff x, y \text{ are adjacent}\right] \geq 2/3\,.$$

   $\mathcal{F}$ is *adjacency sketchable* if it admits an adjacency sketch of constant size.

2. $\mathcal{F}$ admits a *small-distance sketch of size* $s(n,r)$ if for every $r \in \mathbb{N}$ there is a function $D_r : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$ such that $\forall G \in \mathcal{F}$ on n vertices, there is a random function $\mathsf{sk} : V(G) \to \{0,1\}^{s(n,r)}$ satisfying

$$\forall x, y \in V(G): \qquad \mathbb{P}\left[D_r(\mathsf{sk}(x), \mathsf{sk}(y)) = 1 \iff \mathsf{dist}_G(x,y) \leq r\right] \geq 2/3\,.$$

   $\mathcal{F}$ is *small-distance sketchable* if it admits a small-distance sketch of size independent of n. We are borrowing the terminology from *small-distance labeling* [ABR05, GL07a].

3. For constant $\alpha > 1$, $\mathcal{F}$ admits an *α-approximate distance threshold (ADT) sketch of size* $s(n)$ if for every $r \in \mathbb{N}$ there is a function $D_r : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$ such that $\forall G \in \mathcal{F}$ on n vertices, there is a random function $\mathsf{sk} : V(G) \to \{0,1\}^{s(n)}$ satisfying

$$\begin{aligned} \forall x, y \in V(G): \qquad \mathsf{dist}(x,y) \leq r &\implies \mathbb{P}\left[D_r(\mathsf{sk}(x), \mathsf{sk}(y)) = 1\right] \geq 2/3 \\ \mathsf{dist}(x,y) > \alpha r &\implies \mathbb{P}\left[D_r(\mathsf{sk}(x), \mathsf{sk}(y)) = 0\right] \geq 2/3\,. \end{aligned}$$

For a constant $\alpha > 1$, we say that $\mathcal{F}$ is *α-ADT sketchable* if $\mathcal{F}$ admits an α-ADT sketch with size independent of n. $\mathcal{F}$ is *ADT sketchable* if there is a constant $\alpha > 1$ such that $\mathcal{F}$ is α-ADT sketchable. We discuss some nuances of ADT sketch size in Section 6.4. It is important to note here that we want sketches whose size is independent of r, unlike small-distance sketching where the size is allowed to depend on r but the sketch is required to make exact distinctions.

These definitions lead immediately to the questions we study in this thesis.

Question 1.3.8. *Which hereditary graph classes are adjacency sketchable?*

We think of this question as the probabilistic version of the well-studied Implicit Graph Question (Question 1.3.7), and note that the only prior result on randomized adjacency labels appears in a paper of Fraigniaud & Korman [FK09] who independently observed the constant-size labels for trees and proved that no constant-size sketch with *one-sided error* exists for the complements of trees.

Recall that Question 1.3.8 is equivalent to the question of which communication problems have constant-cost protocols; we establish this equivalence, and introduce the basics of adjacency sketching, in Chapter 4. We will put this question in context of structural graph theory by discussing the lattice of hereditary graph classes and where the adjacency sketchable classes belong in this lattice. We briefly discuss our results towards this question below, in Sections 1.3.3 to 1.3.5. Next, we introduce the question

Question 1.3.9. *Which hereditary graph classes are small-distance sketchable?*

Prior to this thesis, the only result towards this question that we are aware of is the above-mentioned communication protocol for Hamming distance [HSZZ06]. One might wonder whether such a sketch exists for planar graphs. Planar graphs are extremely well-studied, and it is quite natural to imagine Alice and Bob each having vertices in a planar graph and desiring to compute whether they are within a certain radius of each other. This question was posed in [Har20], which presented a sketch for determining $\mathsf{dist}(x, y) \leq 2$ vs $\mathsf{dist}(x, y) > 2$ in planar graphs. We resolved this question in [HWZ22] and in more generality in [EHK22] using *sparsity theory* [NO12] and the theory of *first-order transductions* of graphs.

Finally, we introduce a graph version of the approximate distance sketching problem that has seen significant attention in sublinear algorithms:

Question 1.3.10. *Which hereditary graph classes are ADT sketchable?*

Most work on approximate distance sketching has focused on norms [AKR18], and the only prior work on graphs that we are aware of is the unpublished manuscript of Andoni & Krauthgamer [AK08].

Note that small-distance sketching implies adjacency sketching, by definition. The opposite implication is false, as witnessed by the class of graphs with arboricity 2 (see Example 6.3.2). For $\epsilon < 1$, it is also true that $(1+\epsilon)$-ADT sketches imply adjacency sketches. It seems reasonable to suspect that the small-distance and approximate distance problems are also related in some way. So we ask:

Question 1.3.11. *What is the relationship, if any, between adjacency, small-distance, and ADT sketching?*

Finally, we remark upon an interesting connection to learning theory which gives an additional motivation for Question 1.3.8, although this connection will not be exploited in our results.

Remark 1.3.12. Write S^{d-1} for the unit sphere in $\mathbb{R}^d$. For a bipartite graph $G = (X, Y, E)$, we say that $\phi_X : X \to S^{d-1}$, $\phi_Y : Y \to S^{d-1}$ is a d-dimensional dot-product representation of G if

$$\forall x \in X, y \in Y : \qquad xy \in E \iff \langle \phi_X(x), \phi_Y(y) \rangle \geq 0 \,.$$

We say that the *margin* of ϕ_X, ϕ_Y is $\mathsf{mar}(\phi_X, \phi_Y) := \min_{(x,y)\in X\times Y} |\langle \phi_X(x), \phi_Y(y)\rangle|$. We may then define the *margin complexity* of G as $\mathsf{mc}(G) := \min_{\phi_X,\phi_Y}(\mathsf{mar}(\phi_X, \phi_Y))^{-1}$, where the minimum is taken over all d-dimensional dot-product representations of G in all dimensions d. For a class $\mathcal{F}$ of graphs, we define $\mathsf{mc}(\mathcal{F}) := \sup_{G\in\mathcal{F}} \mathsf{mc}(G)$.

Linial & Shraibman [LS09] proved an equivalence (up to constant factors) between discrepancy and margin; in our terms, this means a bipartite graph G with a constant-cost protocol computing adjacency has $\mathsf{mc}(G) = O(1)$. Then any class $\mathcal{F}$ with a constant-size adjacency sketch has $\mathsf{mc}(\mathcal{F}) = O(1)$. It is an easy exercise to prove the converse: any class with $\mathsf{mc}(\mathcal{F}) = O(1)$ has a constant-size adjacency sketch, which can be obtained by sampling random halfspaces h through the origin and recording the sign of $\phi_X(x)$ with respect to each h in the sketch. So the existence of constant-size adjacency sketches is equivalent to the condition $\mathsf{mc}(\mathcal{F}) = O(1)$. We may think of the vertices $y \in Y$ of any graph $G = (X, Y, E) \in \mathcal{F}$ as a halfspace $H_y := \{x \in X : \langle \phi_X(x), \phi_Y(y)\rangle \geq 0\}$, so Y corresponds to a hypothesis class over X. A classic result of learning theory is that this class can be learned by the perceptron algorithm making at most $\mathsf{mc}(G)^2$ mistakes. We therefore have that, in adjacency sketchable graph classes, the hypothesis class defined by the columns (or rows) of the adjacency matrix can be learned efficiently by the perceptron.

1.3.3 Results I: Subgraphs of Cartesian Products

We introduce our model of sketching in Chapter 4 and present some basic results and definitions. These include the formal relation between constant-cost communication and adjacency sketching and labeling, the notion of *probabilistic universal graphs* (which are the probabilistic version of induced-universal graphs), some basic facts about the lattice of hereditary graph classes, and, as a warm-up, an adjacency sketch and corresponding adjacency labeling scheme for induced subgraphs of hypercubes.

This leads us to Chapter 5, which presents some results for Cartesian products, taken from the papers [HWZ22] and [EHZ22], that demonstrate the usefulness of randomized communication complexity techniques in the study of adjacency labeling schemes, as well as the limitations of randomized communication based on the standard example of the EQUALITY problem.

We introduce some notation. For two graphs G and H, we write $G \subset H$ if G is a subgraph of H, and $G \sqsubset H$ if it is an *induced* subgraph. For a set $\mathcal{F}$ of graphs, we write

$$\mathsf{mon}(\mathcal{F}) := \{G : \exists H \in \mathcal{F}, G \subset H\} \text{ and}$$
$$\mathsf{her}(\mathcal{F}) := \{G : \exists H \in \mathcal{F}, G \sqsubset H\}$$

for the *monotone closure* and *hereditary closure* of $\mathcal{F}$, respectively.

Cartesian products. Let $d \in \mathbb{N}$ and $G_1, \ldots, G_d$ be any graphs. The Cartesian product $G_1 \square G_2 \square \cdots \square G_d$ is the graph whose vertices are the tuples $(v_1, \ldots, v_d) \in V(G_1) \times \cdots \times V(G_d)$, and two vertices v, w are adjacent if and only if there is exactly one coordinate $i \in [d]$ such that $(v_i, w_i) \in E(G_i)$ and for all $j \neq i$, $v_j = w_j$.

For any set of graphs $\mathcal{F}$, we will define the set of graphs $\mathcal{F}^{\square}$ as all graphs obtained by taking a product of graphs in $\mathcal{F}$:

$$\mathcal{F}^{\square} := \{G_1 \square G_2 \square \cdots \square G_d : d \in \mathbb{N}, \forall i \in [d]\ G_i \in \mathcal{F}\}\ .$$

For a fixed graph G we will write G^d for the d-wise Cartesian product of G. For example, the hypercube on $n = 2^d$ vertices is the Cartesian product K_2^d. We refer to the classes $\mathsf{mon}(\{K_2\}^{\square})$ and $\mathsf{her}(\{K_2\}^{\square})$ as the subgraphs and induced subgraphs of hypercubes, respectively.

Cartesian products have appeared several times in the recent literature on labeling schemes [CLR20, AAL21, AAA+22], and are extremely natural for the problem of adjacency labeling. For example, taking $\mathcal{F}$ to be the class of complete graphs, a labeling

scheme for $\mathsf{her}(\mathcal{F}^{\square})$ is equivalent to an encoding $\ell : T \to \{0,1\}^*$ of a set of strings $T \subseteq \Sigma^*$, with Σ being an arbitrarily large finite alphabet, such that a decoder who doesn't know T can decide whether $x, y \in T$ have Hamming distance 1, using only the encodings $\ell(x)$ and $\ell(y)$. Replacing complete graphs with, say, paths, one obtains induced subgraphs of grids in arbitrary dimension. Switching to $\mathsf{mon}(\mathcal{F}^{\square})$ allows arbitrary edges of these products to be deleted. An adjacency labeling scheme of size $O(\log^2 n)$ for the subgraphs of the hypercube follows from a folklore bound of $\log n$ on their degeneracy (see e.g. [Gra70]) combined with a general $O(k \log n)$ bound on the size of labeling schemes for graphs of degeneracy k [KNR92]. Prior to this thesis, no improvement on this was known, even for *induced* subgraphs; in fact, it was unknown even whether these classes satisfy the required condition $|\mathcal{F}_n| \leq 2^{O(n \log n)}$.

It is also natural to consider the problem of constructing *induced-universal graphs* (or simply *universal graphs*) for Cartesian products. A sequence of graphs $(U_n)_{n \in \mathbb{N}}$ are universal graphs of size $n \mapsto |V(U_n)|$ for a hereditary class $\mathcal{F}$ if every $G \in \mathcal{F}_n$ is an induced subgraph of U_n. [KNR92] observed that an adjacency labeling scheme of size $s(n)$ is equivalent to a universal graph of size $2^{s(n)}$, so that universal graphs of size $\mathrm{poly}(n)$ are equivalent to adjacency labels of size $O(\log n)$. This is especially interesting for the classes $\mathsf{her}(\mathcal{F}^{\square})$, since Cartesian products have natural universal graphs by definition: if U_n is a universal graph for $\mathcal{F}$ then for large enough $d = d(n)$, U_n^d is a universal graph for $\mathsf{her}(\mathcal{F}^{\square})$. However, in general, it has at least exponential size: a star with $n-1$ leaves is a member of $\mathsf{her}(\{K_2\}^{\square})$ but the smallest product it can be embedded into is K_2^{n-1}.

Adjacency and Small-Distance Sketching (Section 5.1). Designing an adjacency labeling scheme for induced subgraphs of hypercubes (rather, the weaker question of proving bounds on $|\mathcal{F}_n|$) was an open problem of Alecu, Atminas, & Lozin [AAL21]. They also asked whether any graph class of unbounded *functionality* (which we will not define here) has an adjacency labeling schemes of size $O(\log n)$, and they proved that hypercubes have unbounded functionality. These latter two questions are answered by the following theorem, which easily follows by derandomizing the communication protocol for 1-Hamming Distance, for which we give a simple exposition in Section 4.1.

Theorem 1.3.13. *There is a constant-size adjacency sketch for* $\mathsf{her}(\{K_2\}^{\square})$*, and consequently an adjacency labeling scheme of size* $O(\log n)$*.*

We generalize this result to arbitrary Cartesian products, by showing that the Cartesian product essentially preserves constant-size sketches.

Theorem 1.3.14. *Let $\mathcal{F}$ be a hereditary class of graphs that admits a small-distance sketch of size $s(n,k)$. Then $\mathcal{F}^{\square}$ admits a small-distance sketch of size $O(s(n,k)\cdot k^2 \log k)$. Consequently, if $\mathcal{F}$ is adjacency sketchable, then $\mathsf{her}(\mathcal{F}^{\square})$ is adjacency sketchable and admits an adjacency labeling scheme of size $O(\log n)$.*

Adjacency Labeling (Section 5.1). Theorem 1.3.14 does not yet help to understand adjacency labeling schemes when $\mathcal{F}$ is not adjacency sketchable, and it also does not help to understand adjacency labeling schemes for the *subgraphs* of Cartesian products, i.e. the classes $\mathsf{mon}(\mathcal{F}^{\square})$. Indeed, these latter classes are not adjacency sketchable (see Theorem 1.3.19 below).

Chepoi, Labourel, & Ratel [CLR20] recently studied the structure of general Cartesian products with the motivation of designing adjacency labeling schemes for the monotone classes $\mathsf{mon}(\mathcal{F}^{\square})$. They observe that for $G \in \mathsf{mon}(\{K_2\}^{\square})$, a bound of $\mathsf{vc}(G)$ on the degeneracy of G holds by an inequality of Haussler [Hau95], where $\mathsf{vc}(G)$ is the VC dimension of G. Extending this relation to more general Cartesian products, they give upper bounds on the label size for a number of special cases but do not improve in general upon the $O(\log^2 n)$ bound for subgraphs of hypercubes.

We extend our techniques from Theorem 1.3.14 to achieve the optimal $O(\log n)$, and in general show how to construct optimal labels for all subgraphs and induced subgraphs of Cartesian products. In terms of universal graphs, our proof shows that one can obtain universal graphs for $\mathsf{her}(\mathcal{F}^{\square})$ and $\mathsf{mon}(\mathcal{F}^{\square})$ from the universal graphs U_n for $\mathcal{F}$, although this transformation may not be clearly interpretable from a graph theoretic perspective.

Theorem 1.3.15. *Let $\mathcal{F}$ be a hereditary class with an adjacency labeling scheme of size $s(n)$. Then:*

1. *$\mathsf{her}(\mathcal{F}^{\square})$ has a labeling scheme of size at most $4s(n) + O(\log n)$.*
2. *$\mathsf{mon}(\mathcal{F}^{\square})$ has a labeling scheme where each $G \in \mathsf{mon}(\mathcal{F}^{\square})$ on n vertices is given labels of size at most $4s(n) + O(\delta(G) + \log n)$, where $\delta(G)$ is the degeneracy of G.*

This theorem is optimal up to constant factors (see Section 5.1.4). All of the labeling schemes of Chepoi, Labourel, & Ratel [CLR20] are obtained by bounding $\delta(G)$ and applying the black-box $O(\delta(G)\cdot \log n)$ bound of [KNR92]. For example, they get labels of size $O(d \log^2 n)$ when the base class $\mathcal{F}$ has degeneracy d, by showing that $\mathsf{mon}(\mathcal{F}^{\square})$ has degeneracy $O(d \log n)$. Our result can substituted for that black-box, replacing the multiplicative $O(\log n)$ with an *additive* $O(\log n)$, thereby improving all of the results of [CLR20]; for example, achieving $O(d \log n)$ when $\mathcal{F}$ has degeneracy d.

Recall that the Implicit Graph Question asks for adjacency labels of size $O(\log n)$, which requires a bound of $\log |\mathcal{F}_n| = O(n \log n)$ on the number of graphs in the class. One may generalize the question to ask for adjacency labels that meet the information-theoretic minimum for encoding a graph in the class; in other words, we may ask when labels of size $O(\frac{1}{n} \log |\mathcal{F}_n|)$ are possible. Say that a hereditary class $\mathcal{F}$ admits an *efficient* labeling scheme if it either admits a *constant-size* labeling scheme (which is equivalent to the condition $\log |\mathcal{F}_n| = o(n \log n)$ [Sch99]; we include here the case where $\mathcal{F}$ is finite), or it admits a labeling scheme of size $O(\frac{1}{n} \log |\mathcal{F}_n|)$. Equivalently, the class $\mathcal{F}$ admits a universal graph of size $\text{poly}(|\mathcal{F}_n|^{1/n})$. Then Theorem 1.3.15 has the following consequence, which follows from the fact that it is optimal.

Corollary 1.3.16. *If a hereditary class $\mathcal{F}$ has an efficient labeling scheme, then so do* $\mathsf{her}(\mathcal{F}^{\square})$ *and* $\mathsf{mon}(\mathcal{F}^{\square})$.

The Limitations of Equality (Section 5.2) Recall that the Equality communication problem is the standard example of randomization in communication, and that computing equality is the purpose of standard practical hashing algorithms like SHA256. We call any communication protocol, sketch, or labeling scheme *equality-based* if it reduces to a constant number of instances of Equality (see Chapter 4 for a formal definition). One fundamental question is whether Equality is the *only* way to use randomness to achieve constant cost protocols.

To formalize this, we can define *deterministic* communication protocols with access to a unit-cost Equality oracle. In this type of protocol, the two players Alice and Bob can each construct a binary string and give it to the oracle, who deterministically reports whether they have given identical strings or not; the players are charged 1 bit of communication to use the oracle.

Our question now becomes: Can any constant-cost randomized protocol be simulated by a constant-cost deterministic protocol of this type? Recent work [CLV19] has studied the power of the Equality oracle and shown that it does not capture the power of randomized communication in the *non-constant* setting, but this leaves our question open. We show that the hypercubes give a negative answer. This result was also proved concurrently and independently, using a different technique, by Hambardzumyan, Hatami, & Hatami [HHH21b].

Theorem 1.3.17. *There is no constant-cost equality-based protocol for computing adjacency in K_2^d.*

Adjacency in the hypercube (and its generalizations, including Cartesian products and k-Hamming Distance for constant k) remain the only examples known to us of constant-cost communication problems that cannot be reduced to Equality.

It was observed by Bonamy & Girão (communicated to us by Louis Esperet) that our proof technique may be generalized to show the following. (Note that the induced subgraphs of hypercubes are bipartite and contain no $K_{2,3}$ subgraph.)

Theorem 1.3.18. *For any $t \in \mathbb{N}$, if $\mathcal{F}$ is any class of bipartite graphs with no $K_{t,t}$ subgraph, then $\mathcal{F}$ admits a constant-size equality-based adjacency sketch if and only if it has bounded degeneracy.*

One non-trivial application of this theorem is for point-box incidence graphs, which are bipartite graphs $G = (P, B, E)$ where P is a set of points in $\mathbb{R}^2$ and B is a set of axis-aligned boxes, with an edge $pb \in E$ for $p \in P, b \in B$ if and only if $p \in b$. Restricting ourselves to the $K_{2,2}$-free point-box incidence graphs, we know that if these graphs have $O(n)$ edges then they have bounded degeneracy and therefore a constant-size equality-based adjacency sketch. However, a recent construction [BCS+21] shows that these graphs may have $\Omega(n\frac{\log n}{\log\log n})$ edges and therefore no equality-based adjacency sketch. In fact, this holds for the restricted class of $K_{2,2}$-free *dyadic* point-box incidence graphs, where there is a matching upper bound on the number of edges.

1.3.4 Results II: Monotone Classes of Graphs

In Chapter 6 we give our results for *monotone* classes of graphs, from the paper [EHK22], coauthored with Louis Esperet and Andrey Kupavskii. A class of graphs is monotone if it is closed under taking subgraphs. Monotone classes are ubiquitous in graph theory; examples include planar graphs, bounded treewidth graphs, bounded degree graphs, minor-closed classes, and so on. For these classes, we give a complete theory of adjacency and small-distance sketching, and make progress towards a theory of ADT sketching.

To answer Question 1.3.11 about the relationship between adjacency, small-distance, and ADT sketching, we give a high-level hierarchy. Let ADJ be the adjacency sketchable monotone graph classes, SD the small-distance sketchable monotone graph classes, and ADT the ADT sketchable monotone graph classes. Then

$$\mathsf{ADT} \subsetneq \mathsf{SD} \subsetneq \mathsf{ADJ}\,.$$

That $\mathsf{SD} \subseteq \mathsf{ADJ}$ follows by definition, and $\mathsf{SD} \neq \mathsf{ADJ}$ is witnessed by the class of arboricity-2 graphs (Example 6.3.2), so the challenging part of this hierarchy is $\mathsf{ADT} \subsetneq \mathsf{SD}$. Our results clarify these relations as follows.

Adjacency Sketching (Section 6.2) Recall that a graph has degeneracy δ if all of its subgraphs have a vertex of degree at most δ, and a graph class $\mathcal{F}$ has *bounded degeneracy* if there is some constant δ such that all $G \in \mathcal{F}$ have degeneracy δ. Bounded degeneracy graphs have a simple equality-based adjacency sketch obtained by replacing the vertex names in the pruned adjacency list with hashes, as mentioned at the top of Section 1.3, which gives a bound of $O(\delta \log \delta)$ on the sketch size; we can also get an improved bound of $O(\delta)$ using Bloom filters (Lemma 4.2.16). For monotone classes, we show that these types of sketches are always sufficient:

Theorem 1.3.19. *Let $\mathcal{F}$ be a monotone class of graphs. Then $\mathcal{F}$ is adjacency sketchable if and only if $\mathcal{F}$ has bounded degeneracy.*

Small-Distance Sketching (Section 6.3). Next, we answer Question 1.3.9 by characterizing the monotone graph classes that are small-distance sketchable as exactly those with *bounded expansion* (as in [NO12]; see our Definition 6.1.1). Informally, bounded expansion means that the edge density of a graph increases only as a function of r when contracting subgraphs of radius r into a single vertex. Many graph classes of theoretical and practical importance have bounded expansion, including bounded-degree graphs, proper minor-closed graph classes, and graphs of bounded genus [NO12], along with many random graph models and real-world graphs [DRR+14].

To state our theorem, we briefly describe a more general type of sketch, called *first-order* sketching, which we introduce in this thesis (taken from [HWZ22]). A graph class $\mathcal{F}$ is *first-order sketchable* if any first-order (FO) formula $\phi(x, y)$ over the vertices and edge relation of the graph (with two free variables whose domain is the set of vertices) is sketchable (see Section 4.2.1). This type of sketch was introduced in [HWZ22] and generalizes small-distance sketching, along with (for example) testing whether vertices x, y belong to a subgraph isomorphic to some fixed graph H. We show that, for monotone graph classes, first-order sketchability is equivalent to small-distance sketchability.

Theorem 1.3.20. *Let $\mathcal{F}$ be a monotone class of graphs. Then the following are equivalent:*

1. *$\mathcal{F}$ is small-distance sketchable;*
2. *$\mathcal{F}$ is first-order sketchable;*
3. *$\mathcal{F}$ has bounded expansion.*

The implications (3) $\implies$ (2) $\implies$ (1) do not require monotonicity, and (2) $\implies$ (1) holds by definition. We actually prove a stronger version of (3) $\implies$ (2) than this theorem

requires: Our proof holds for all class of *structurally* bounded expansion, which are classes obtained by first-order (FO) transductions of classes with bounded expansion. This proof, using the recent structural result of [GKN+20], does not give explicit bounds on the sketch size. To get explicit bounds, we give a separate proof of (3) $\implies$ (1) that gives, as a corollary, a bound polynomial in r for planar graphs and K_t-minor free graphs.

Approximate Distance Threshold Sketching (Section 6.4). Keeping in mind Theorem 1.3.20, a reasonable question is whether ADT sketching for monotone classes is also determined by expansion. Our first result is that bounded expansion is necessary.

Theorem 1.3.21. *If a monotone class $\mathcal{F}$ is ADT sketchable, then it has bounded expansion.*

Combined with Theorem 1.3.20, this proves $\mathsf{ADT} \subseteq \mathsf{SD}$. We are then concerned with the converse of this theorem. We show that the class of max-degree 3 graphs, which has bounded expansion exponential in r [NO08], is not ADT sketchable.

Theorem 1.3.22. *For any $\alpha > 1$, any α-ADT sketch for the class of graphs with maximum degree 3 has size at least $\Omega(n^{\frac{1}{4\alpha}-\varepsilon})$, for any constant $\varepsilon > 0$.*

This establishes that $\mathsf{ADT} \neq \mathsf{SD}$. But max-degree 3 graphs have exponential expansion. Smaller bounds on the expansion are associated with structural properties: for example, in monotone classes, polynomial expansion is equivalent to the existence of strongly sublinear separators [DN16]. One may then wonder if smaller bounds on the expansion suffice to guarantee ADT sketchability. We prove that this is not the case for two natural examples: subgraphs of the 3-dimensional grid (with polynomial expansion [NO12]), and subgraphs of the 2-dimensional grid with crosses (with linear expansion [Dvo21]) are not ADT sketchable. We strengthen this result by showing that one can obtain monotone classes of graphs with expansion that grows arbitrarily slowly, which are not ADT sketchable.

Theorem 1.3.23. *For any function ρ tending to infinity, there exists a monotone class of expansion $r \mapsto \rho(r)$ that is not ADT sketchable. Moreover, for any $\varepsilon > 0$, there exists a monotone class $\mathcal{F}$ of expansion $r \mapsto O(r^\varepsilon)$, such that, if $\mathcal{F}$ admits an α-ADT sketch of size $s(n)$, then we must have $s(n) = n^{\Omega(1/\alpha)}$.*

We will conclude Chapter 6 with a brief discussion of upper bounds for ADT sketching. Using the sketches obtained from *sparse covers*, combined with results of [Fil20] on sparse covers (based on [KPR93, FT03]), we obtain the following, which complements our Theorem 1.3.23; note that the graph classes with constant expansion are exactly the proper minor-closed classes [NO12].

Corollary 1.3.24. *For any $t \geq 4$, the class of K_t-minor-free graphs has a $O(2^t)$-ADT sketch of size $O(t^2 \log t)$. The sketch is equality-based and has one-sided error. As a consequence, every monotone class of constant expansion is ADT sketchable.*

See Chapter 6 for a deeper discussion of upper bounds.

1.3.5 Results III: Beyond Monotone Classes

Chapter 7 will discuss our results on more general classes of graphs, from the paper [HWZ22] coauthored with Sebastian Wild and Viktor Zamaraev. To state these results, we introduce the notion of *stability* of a graph class. The name *stability* is taken from the literature on model checking, e.g. [CS18, NMP+21, GPT21].

The Greater-Than communication problem is the problem $f_n : [n] \times [n] \to \{0, 1\}$ defined as $f(i, j) = 1$ if and only if $i > j$. This problem has a lower bound of $\Omega(\log n)$ in the randomized SMP model of communication and $\Omega(\log \log n)$ in the public-coin two-way model of communication (see references in Appendix B), and therefore any graph class which encodes the Greater-Than problem cannot have constant-size adjacency sketches. We formalize this by introducing the *chain number*.

For any graph G, the *chain number* $\mathsf{ch}(G)$ is the maximum number k such that there exist disjoint sets of vertices $\{a_1, \ldots, a_k\}$ and $\{b_1, \ldots, b_k\}$ in $V(G)$ such that $a_i b_j \in E(G)$ if and only if $i \leq j$. It is clear that the Greater-Than problem on domain $[k]$ can be reduced to adjacency in any graph with $\mathsf{ch}(G) \geq k$. We say that a class $\mathcal{F}$ is *stable* if there is some fixed k such that $\mathsf{ch}(G) \leq k$ for all $G \in \mathcal{F}$. Intuitively, a class is stable if arbitrarily large instances of Greater-Than cannot be found inside $\mathcal{F}$. Therefore, stability is a necessary condition for adjacency sketchability.

We now have two necessary conditions for sketchability of a hereditary graph class $\mathcal{F}$: the condition $|\mathcal{F}_n| = 2^{O(n \log n)}$, and stability. We will show that in many cases, these two conditions are also sufficient. However, these conditions are not *always* sufficient, which we discuss at the end.

Interval and Permutation Graphs (Section 7.1). Two typical examples of graph classes with adjacency labels of size $O(\log n)$ are the interval graphs (Example 1.3.3) and permutation graphs. A graph is a *permutation graph* if each vertex can be associated with a point in $\mathbb{R}^2$, such that two vertices $x, y \in \mathbb{R}^2$ are adjacent if and only if they are comparable in the natural partial order on $\mathbb{R}^2$ (where $x = (x_1, x_2) \leq (y_1, y_2) = y$ when $x_1 \leq y_1$ and

$x_2 \leq y_2$). Both of these classes are defined in such a way that demands comparison between numbers, so it is easy to see that the Greater-Than communication problem can be reduced to adjacency in graphs that belong to these classes, giving a lower bound of $\Omega(\log n)$ on the adjacency sketch size. Both of these classes have simple adjacency labeling schemes of size $O(\log n)$, so the sketch size is $\Theta(\log n)$; in other words, randomization does not give a significant advantage.

However, we find that eliminating the Greater-Than subproblems is sufficient to drop the sketch size to $O(1)$. In other words, if we fix a constant k and let $\mathcal{F}$ be the hereditary subclass of the interval or permutation graphs with $\mathsf{ch}(G) \leq k$ for all $G \in \mathcal{F}$, we obtain constant-size sketches. We remark that the proofs for these two closely-related classes are actually quite different. In both cases, our sketches are equality-based (i.e. they reduce the problem to a constant number of instances of Equality).

Theorem 1.3.25. *Let $\mathcal{F}$ be any hereditary subclass of interval or permutation graphs. Then $\mathcal{F}$ is adjacency sketchable if and only if it is stable.*

Bipartite Graphs (Section 7.2). A Boolean-valued communication problem is defined by a Boolean matrix, and may therefore be interpreted as a bipartite graph. To characterize the adjacency sketchable classes, it is sufficient to consider only the classes of bipartite classes. Any hereditary class $\mathcal{F}$ of graphs can be defined by a unique (possibly infinite) set of *forbidden induced subgraphs*: a set of graphs $\mathcal{H}$ which may not appear as an induced subgraphs of any $G \in \mathcal{F}$.

Therefore, a natural step towards understanding adjacency sketchability is to consider the bipartite graph classes defined by a single forbidden induced subgraph. These are called *monogenic* bipartite graph classes. For a bipartite graph H, we will refer to the class of bipartite graphs that forbid H as an induced subgraph[10] as the *H-free bipartite graphs*. Recall that adjacency sketchable classes must have at most $2^{O(n \log n)}$ unique n-vertex graphs, and that these graph classes are said to have *factorial speed*, so we must restrict our attention to the graphs H where the H-free bipartite graphs have factorial speed. As was the case for interval and permutation graphs, we find that stability is equivalent to adjacency sketchability:

Theorem 1.3.26. *Let H be a bipartite graph such that the class $\mathcal{H}$ of H-free bipartite graphs has at most factorial speed. Then any hereditary subclass $\mathcal{F}$ of $\mathcal{H}$ is adjacency sketchable if and only if it is stable.*

[10] Bipartite graphs require a more careful definition of *induced subgraph*; see Chapter 7.

To prove this theorem, we require new structural results for some classes of bipartite graphs. Previous work [All09, LZ17] has shown that a class of H-free bipartite graphs is factorial only when H is an induced subgraph of $P_7, S_{1,2,3}$, or one of the infinite set $\{F^*_{p,q}\}_{p,q\in\mathbb{N}}$ (defined in Section 7.2). We construct a new decomposition scheme for the $F^*_{p,q}$-free graphs whose depth is controlled by the chain number, and we show that the chain number controls the depth of the decomposition of [LZ17] for P_7-free graphs.

Remarks on the Stability Condition. The stability condition has in interesting relationship to the structure of hereditary graph classes, due to [Ale92, BT95, SZ94, Ale97], which we discuss in more detail in Chapter 4. Briefly, the hereditary graph classes $\mathcal{F}$ which satisfy $\log|\mathcal{F}_n| = \Theta(n\log n)$ are sometimes called the *factorial layer* of the lattice of hereditary graph classes. This layer is separated from the layers below: if a hereditary graph class has $\log|\mathcal{F}_n| = o(n\log n)$ then it must have $\log|\mathcal{F}_n| = O(n)$; there is nothing in between. There are several other "jumps" in the quantity $|\mathcal{F}_n|$. There are 9 hereditary graph classes $\mathcal{M}$ with the property that any class in the factorial layer must contain at least one of these classes $\mathcal{M}$ as a subclass. Three of these classes, the *chain graphs*, *co-chain graphs*, and *threshold graphs* are not stable, and in fact a class is stable if and only if it does not contain any of these. The other 6 minimal classes admit constant-size adjacency sketches. A similar result holds for another "jump" in $|\mathcal{F}_n|$ which occurs at the Bell numbers, where the chain graphs, co-chain graphs, and threshold graphs are again minimal classes above this jump and the remaining minimal classes admit constant-size sketches.

Due to the fundamental role of stability in the structure of hereditary graph classes, and the fact that stability was (surprisingly) sufficient for constant-size sketches in the interval and permutation graphs, monogenic bipartite graph classes, and classes of structurally bounded expansion, we conjectured in an early version of the paper [HWZ22] that stability is a sufficient condition for sketchability among classes with $|\mathcal{F}_n| = 2^{O(n\log n)}$.

We now have several examples of stable classes with $|\mathcal{F}_n| = 2^{O(n\log n)}$, but which are not sketchable. One example was quickly provided by Hambardzumyan, Hatami, & Hatami [HHH21a] using a construction from their concurrent work [HHH21b]. Hatami & Hatami extended this construction to refute the Implicit Graph Conjecture [HH21]. More examples can be found with the help of Theorem 1.3.19, which showed that any monotone class has a constant-size adjacency sketch if and only if it has bounded degeneracy. For a monotone graph class, having bounded degeneracy is equivalent to the condition that each n-vertex graph has at most $O(n)$ edges. Also, for any *hereditary* graph class $\mathcal{F}$ with $O(n\log n)$ edges and $\log|\mathcal{F}_n| = O(n\log n)$ has the property that when we take the monotone closure $\mathcal{G} = \mathsf{mon}(\mathcal{F})$, we get $|\mathcal{G}_n| \le 2^{O(n\log n)}\cdot 2^{O(n\log n)}$. This is because any graph $G \in \mathcal{G}$ is a

spanning subgraph of some $H \in \mathcal{F}$ (i.e. $G \subset H$ and $V(G) = V(H)$) and each $H \in \mathcal{F}$ with $O(n \log n)$ edges has at most $2^{O(n \log n)}$ spanning subgraphs. Any such class must also be stable, since any monotone class that is not stable contains *all* bipartite graphs.

Choosing any hereditary class $\mathcal{F}$ with $|\mathcal{F}_n| = 2^{O(n \log n)}$ and with the number of edges between $\omega(n)$ and $O(n \log n)$ then produces a class $\mathsf{mon}(\mathcal{F})$ which is stable and has $|\mathsf{mon}(\mathcal{F})_n| = 2^{O(n \log n)}$, but which has $\omega(n)$ edges and therefore is not sketchable. Examples include the subgraphs of the hypercube and subgraphs of the $K_{2,2}$-free incidence graphs between points and dyadic boxes studied in [BCS+21], discussed at the bottom of Section 1.3.3.

Chapter 2

Distribution-Free Sample-Based Testing

It is a fact, beyond comprehension,
Accepted by all, without abstention,
Studied by many, with much extension,
That all things relate to VC dimension.

We begin Part I with results on the testing vs. learning question, which appeared in the paper [BFH21], coauthored with Eric Blais and Renato Ferreira Pinto Jr. Recall from Section 1.2.2 the definitions of property testing and learning. As discussed in Section 1.2.3, we are interested in identifying the hypothesis classes where the labeled-sample complexity of testing is much less than that of learning. The most important example to keep in mind is the class of *halfspaces*. In this chapter, for a given hypothesis class $\mathcal{H}$ (which will be clear from context), we will write simply $\mathsf{test}(\epsilon) := \mathsf{test}^{\mathcal{O}}_{\mathcal{H}}(\epsilon)$ where $\mathcal{O}$ is just the labeled-sample oracle, and we will similarly write $\mathsf{plearn}(\epsilon) := \mathsf{plearn}^{\mathcal{O}}_{\mathcal{H}}(\epsilon)$.

The fundamental result of PAC learning (see e.g. [SB14]) is that the VC dimension of $\mathcal{H}$ determines the labeled-sample complexity of learning. A set $T \subseteq \mathcal{X}$ is *shattered* by $\mathcal{H}$ if for every $\ell : T \to \{0,1\}$ there is a function $f \in \mathcal{H}$ that agrees with ℓ on all points in T. The *VC dimension of $\mathcal{H}$ with respect to $S \subseteq \mathcal{X}$* is

$$\mathsf{VC}_S(\mathcal{H}) := \max\{k : \exists T \subseteq S \text{ of size } |T| = k \text{ that is shattered by } \mathcal{H}\}.$$

We will write $\mathsf{VC}(\mathcal{H}) := \mathsf{VC}_{\mathcal{X}}(\mathcal{H})$. When $\epsilon > 0$ is constant, $\mathsf{plearn}(\epsilon) = \Theta(\mathsf{VC}(\mathcal{H}))$, so to understand the relationship between $\mathsf{test}(\epsilon)$ and $\mathsf{plearn}(\epsilon)$, it is necessary to understand the relationship between $\mathsf{test}(\epsilon)$ and the VC dimension.

The VC dimension has appeared in the property testing literature, but its use has mostly been limited to upper bounds (e.g. [GGR98, ADPR03, BBBY12, AFZ19]). Since $\mathsf{plearn}(\epsilon) = O(\mathsf{VC}(\mathcal{H})/\epsilon)$ (Theorem 1.2.9) and $\mathsf{test}(\epsilon) = O(\mathsf{plearn}(\epsilon/2)+1/\epsilon)$ (Theorem 1.2.5), we may conclude that $\mathsf{test}(\epsilon) = O(\mathsf{VC}(\mathcal{H})/\epsilon)$. This can be improved to one-sided error (at a small cost in terms of ϵ) by using a direct proof that does not go through the testing-to-learning reduction of [GGR98]: for the sake of completeness, we show in Theorem 2.1.1 that $\mathsf{otest}(\epsilon) = O\left(\frac{\mathsf{VC}(\mathcal{H})}{\epsilon} \log \frac{1}{\epsilon}\right)$.

We are interested in determining the cases where the VC dimension is also a *lower bound* on $\mathsf{test}(\epsilon)$. Such lower bounds would be desirable not only for understanding the relationship between testing and learning, but also because they would be *combinatorial* in nature, obtained via an analysis of the structure of the function class, whereas nearly all known lower bounds in sample-based property testing (e.g., [GGR98, KR00, BBBY12, BY19, RR20]) are *distributional*: a probability distribution specific to the problem is constructed and shown to be hard to test.

The VC dimension cannot, in general, be a lower bound on the sample complexity of testing: consider the following example from [GGR98]. Let $\mathcal{H}$ be the set of all Boolean functions $f : \{0,1\}^n \to \{0,1\}$ that satisfy $f(x) = 1$ for all $x \in \{0,1\}^n$ with $x_1 = 1$. $\mathsf{VC}(\mathcal{H}) = \Theta(2^n)$ since the 2^{n-1} points x with $x_1 = 0$ are shattered, while $\mathsf{test}(\epsilon) = O(1/\epsilon)$. Therefore, the relationship of VC dimension to (distribution-free sample-based) property testing is more complicated than to (PAC) learning, and we must introduce some new ideas.

This thesis introduces the *lower VC (LVC)* dimension and uses it to obtain lower bounds. We introduce and motivate the LVC dimension in the next section and present our main theorem of this chapter in Section 2.2. The applications of our theorem to halfspaces, unions of intervals, intersections of halfspaces, etc. are in Sections 2.3 to 2.6.

Our lower bounds show that many of the most natural hypothesis classes in learning theory cannot be tested significantly more efficiently than they can be learned. It is also necessary to study the converse question: whether there are any natural hypothesis classes that are more efficiently testable than learnable. We present two examples in Section 2.7: monotone Boolean functions and k-juntas. We also give an example where our main lower bound is tight, and where giving the tester access to *queries* can achieve significantly better results than the lower bound for labeled samples given by our main theorem.

2.1 LVC Dimension and One-Sided Error Testing

We relate the labeled-sample complexity of testing to the VC dimension by introducing the *lower VC (LVC) dimension*. Although our goal is to obtain lower bounds for two-sided error testers, we motivate this notion by studying the labeled-sample complexity of testing with one-sided error. First, an upper bound on one-sided error testing in terms of the VC dimension: We remark that this result is likely not new, though we have not found a reference for it (a similar proof with a weaker bound was presented in [ADPR03]).

Theorem 2.1.1. *Let $\mathcal{H}$ be a set of functions $\mathcal{X} \to \{0,1\}$ with finite $d = \mathsf{VC}(\mathcal{H}) > 0$. Then for any $\epsilon > 0$, $\mathsf{otest}(\epsilon) = O\left(\frac{d}{\epsilon}\log\frac{1}{\epsilon}\right)$.*

Proof. Let $f : \mathcal{X} \to \{0,1\}$ be the input function, and let $\mathcal{D}$ be the input distribution over $\mathcal{X}$. The algorithm is as follows. Draw a set S of $m = O\left(\frac{d}{\epsilon}\log\frac{1}{\epsilon}\right)$ labelled examples from $\mathcal{D}$ and accept if there exists $h \in \mathcal{H}$ such that $f(x) = h(x)$ for all $x \in S$; otherwise, reject.

If $f \in \mathcal{H}$ then this algorithm accepts with probability 1, so assume that f is ϵ-far from $\mathcal{H}$. Define the class $f \oplus \mathcal{H} := \{f \oplus h : h \in \mathcal{H}\}$ and observe that a set is shattered by $f \oplus \mathcal{H}$ iff it is shattered by $\mathcal{H}$. By standard VC dimension arguments (e.g. [SB14] Theorem 28.3), with probability at least $2/3$ a sample S of size m is an ϵ-net for $f \oplus \mathcal{H}$, meaning that for every $h \in \mathcal{H}$, if $\mathbb{P}_{x\sim\mathcal{D}}[f(x) \neq h(x)] = \mathbb{P}_{x\sim\mathcal{D}}[(f \oplus h)(x) = 1] \geq \epsilon$ then there exists $x \in S$ such that $(f \oplus h)(x) = 1$, i.e. $f(x) \neq h(x)$. Since $\mathbb{P}_{x\sim\mathcal{D}}[f(x) \neq h(x)] \geq \epsilon$ for every $h \in \mathcal{H}$, this implies that the algorithm rejects. □

Now we consider the problem of obtaining a lower bound for one-sided testers. A one-sided tester must obtain *proof* that the input function $f : \mathcal{X} \to \{0,1\}$ does not belong to $\mathcal{H}$ before it can reject. The LVC dimension quantifies the minimum size of this proof. Observe that a set $T \subseteq \mathcal{X}$ that is shattered *cannot* be a proof that $f \notin \mathcal{H}$. The VC dimension is the largest number d such that there *exists* a shattered set of size d. A reasonable attempt to define the minimum "proof size" is to take the largest number d' such that *all* sets of size d' are shattered. But this is not sufficient: any 3 points on a line in $\mathbb{R}^n$ are not shattered by the class of halfspaces, so we would have $d' = 2$, which is not helpful for getting strong lower bounds for testing halfspaces.

Therefore we define LVC dimension with respect to a set $S \subseteq \mathcal{X}$, which we think of as the largest subset of the domain that forbids "small" proofs, as appear in the bad example above. For example, to choose a good set S for halfspaces, we would choose S to avoid any 3 points on a line; e.g. we choose S to be in general position. Formally, we define the following.

Definition 2.1.2 (Lower VC Dimension)**.** For any class $\mathcal{H}$ of Boolean-valued functions over $\mathcal{X}$ and any subset $S \subseteq \mathcal{X}$, define the *LVC dimension* (or *Lower Vapnik-Chervonenkis dimension*) of $\mathcal{H}$ with respect to S to be

$$\mathsf{LVC}_S(\mathcal{H}) := \max\{k : \forall T \subseteq S \text{ of size } |T| = k,\ T \text{ is shattered by } \mathcal{H}\}.$$

The definition of LVC dimension differs from that of the VC dimension only by the replacement of the existential quantifier with a universal one. This immediately implies that $\mathsf{LVC}(\mathcal{H}) \leq \mathsf{VC}(\mathcal{H})$ for every class $\mathcal{H}$ and motivates our choice to call this measure "lower" VC dimension. And in some cases, the LVC dimension of a class can be much smaller than its VC dimension. (See Section 2.5.1 for a discussion of some concepts in learning theory related to LVC dimension.) It is not hard to prove that the LVC dimension is a lower bound on one-sided testing.

Proposition 2.1.3. *Let $\mathcal{H}$ be a set of functions $\mathcal{X} \to \{0,1\}$, let $\epsilon > 0$, and let $\mathcal{D}$ be any distribution over $\mathcal{X}$ such that there exists $f : \mathcal{X} \to \{0,1\}$ with $\mathsf{dist}_{\mathcal{D}}(f, \mathcal{H}) > \epsilon$. Then any one-sided ϵ-tester for $\mathcal{H}$ over $\mathcal{D}$ requires at least $\mathsf{LVC}_S(\mathcal{H})$ queries, where S is the support of $\mathcal{D}$, under the assumption that all queries fall within S.*

Remark 2.1.4. The above proposition holds even even for adaptive testers using queries. The final assumption holds in particular for testers in the labeled-sample model.

Proof. Suppose A is any algorithm that makes at most q queries, where $q \leq \mathsf{LVC}_S(\mathcal{H})$, and for any function $f : \mathcal{X} \to \{0,1\}$ let $\mathcal{Q}_f$ be the distribution of query sequences $((x_1, f(x_1)), \ldots, (x_q, f(x_q)))$ made by the algorithm on input f. Since the algorithm has one-sided error, it must accept every sequence $Q_h \sim \mathcal{Q}_h$ with probability 1 when $h \in \mathcal{H}$. Consider any $f : \mathcal{X} \to \{0,1\}$ and any sequence $Q_f \in \mathrm{supp}(\mathcal{Q}_f)$. Since $q \leq \mathsf{LVC}_S(\mathcal{H})$ and each $x_i \in S$, the set $\{x_1, \ldots, x_q\}$ is shattered by $\mathcal{H}$, so there exists $h \in \mathcal{H}$ such that $h(x_i) = f(x_i)$ for each i; therefore there is $Q_h \in \mathrm{supp}(\mathcal{Q}_h)$ such that $Q_h = Q_f$. Then for every $f, Q_f \sim \mathcal{Q}_f$ is accepted with probability 1, a contradiction. □

The LVC dimension does not, however, immediately lead to a lower bound for testers with *two-sided* error. Whereas one-sided testers must find *proof* of $f \notin \mathcal{H}$ before they reject, a two-sided tester must only find sufficiently strong *evidence*. The main result of this chapter will show that whenever the necessary *proof* is sufficiently large, the number of samples required to obtain *evidence* is also large.

We can obtain a weaker (but more general) result towards this goal by applying a result of Goldreich & Ron [GR16] that relates one- to two-sided error testers.

Theorem 2.1.5 ([GR16], Theorem 1.3 part 1)**.** *For every class $\mathcal{H}$ of functions $\mathcal{X} \to \{0,1\}$ with $\mathcal{X}$ finite, if there is a distribution-free sampling ϵ-tester for $\mathcal{H}$ using $q(\epsilon)$ samples, then there is a one-sided error sampling ϵ-tester for $\mathcal{H}$ over the uniform distribution on $\mathcal{X}$ using at most $\widetilde{O}(q(\epsilon)^2)$ samples.*

Using this result, we get a general relationship between two-sided testers and LVC dimension. We will significantly strengthen this lower bound in the next section.

Corollary 2.1.6. *Let $\mathcal{H}$ be a class of functions $\mathcal{X} \to \{0,1\}$ where $\mathcal{X}$ is finite, let $\epsilon > 0$, and let $S \subseteq \mathcal{X}$ be such that there exists a distribution $\mathcal{D}$ supported on S and a function $f : \mathcal{X} \to \{0,1\}$ satisfying $\mathsf{dist}_{\mathcal{D}}(f, \mathcal{H}) > \epsilon$. Then*

$$\mathsf{test}_{\mathcal{H}}(\epsilon) = \widetilde{\Omega}(\sqrt{\mathsf{LVC}_S(\mathcal{H})})\,.$$

Proof. This follows from Theorem 2.1.5 and Proposition 2.1.3. □

2.2 General Lower Bound

Our main theorem gives a general lower bound on the labeled-sample complexity of two-sided error testing in terms of the VC and LVC dimensions of $\mathcal{H}$.

Theorem 2.2.1. *There are constants $C, \epsilon_0 > 0$ such that for any class $\mathcal{H}$ of Boolean-valued functions over $\mathcal{X}$ and any $S \subseteq \mathcal{X}$, if $|S| \geq 5 \cdot \mathsf{VC}_S(\mathcal{H})$ and $\mathsf{LVC}_S(\mathcal{H}) \geq C \cdot \mathsf{VC}_S(\mathcal{H})^{3/4}\sqrt{\log \mathsf{VC}_S(\mathcal{H})}$, then for all $\epsilon \leq \epsilon_0$,*

$$\mathsf{test}_{\mathcal{H}}(\epsilon) = \Omega\left(\frac{\mathsf{LVC}_S(\mathcal{H})^2}{\mathsf{VC}_S(\mathcal{H}) \log \mathsf{VC}_S(\mathcal{H})}\right).$$

This bound is tight in the sense that there are classes $\mathcal{H}$ for which $\mathsf{test}_{\mathcal{H}}(\epsilon) = \Theta\left(\frac{\mathsf{VC}(\mathcal{H})}{\log \mathsf{VC}(\mathcal{H})}\right)$ (for any constant ϵ), while $\mathsf{LVC}_S(\mathcal{H}) = \mathsf{VC}(\mathcal{H})$ for some S with $|S| \geq 5 \cdot \mathsf{VC}(\mathcal{H})$.

Before we begin, we discuss some examples that illuminate why the conditions in the theorem are important, i.e. the choice of a subset $S \subseteq \mathcal{X}$ with large $\mathsf{LVC}_S(\mathcal{H})$ and the condition $|S| \geq 5 \cdot \mathsf{VC}_S(\mathcal{H})$. Unlike a learning algorithm, a property tester can halt and reject as soon as it sees proof that the unknown function $f : \mathcal{X} \to \{0,1\}$ does not belong to the class. Therefore, we aim to find subsets $S \subseteq \mathcal{X}$ where small "certificates" of non-membership cannot exist. This motivates the definition of $\mathsf{LVC}_S(\mathcal{H})$: any subset $T \subseteq S$ of

size $|T| \leq \mathsf{LVC}_S(\mathcal{H})$ cannot contain any certificates of non-membership, for any function $f \notin \mathcal{H}$. So we want to find sets where $\mathsf{LVC}_S(\mathcal{H})$ is as large as possible relative to $\mathsf{VC}(\mathcal{H})$. On the other hand, if $\mathsf{LVC}_S(\mathcal{H}) = \mathsf{VC}(\mathcal{H})$ but $|S| = \mathsf{VC}(\mathcal{H})$, then the class $\mathcal{H}$ restricted to S is trivial: it contains all possible functions on S, so testing is still easy. $|S|$ must be large enough so that most functions are far from $\mathcal{H}$, and this will be guaranteed in general when $|S| > 5 \cdot \mathsf{VC}_S(\mathcal{H})$ (the constant 5 is somewhat arbitrary). The following examples illustrate these phenomena. In the first example, $\mathsf{LVC}_{\mathcal{X}}(\mathcal{H})$ is constant, but a careful choice of large S allows $\mathsf{LVC}_S(\mathcal{H}) = \mathsf{VC}(\mathcal{H})$, and we will obtain lower bounds for this class:

Example 2.2.2. Let $\mathcal{L}_n$ be the set of halfspaces $\mathbb{R}^n \to \{\pm 1\}$. As is well-known, $\mathsf{VC}_{\mathbb{R}^n}(\mathcal{L}_n) = n + 1$. But $\mathsf{LVC}_{\mathbb{R}^n}(\mathcal{L}_n) = 2$, since any 3 colinear points cannot be shattered. On the other hand, if $S \subseteq \mathbb{R}^n$ is a set of points in general position and $|S| > n + 1$, then $\mathsf{LVC}_S(\mathcal{L}_n) = \mathsf{VC}_S(\mathcal{L}_n) = n + 1$.

In the second example, the conditions of our theorem fail: finding a good set S is impossible, and indeed there is an efficient distribution-free sample-based tester; see Theorem 2.7.9.

Example 2.2.3. Let $\mathcal{M}$ be the set of monotone functions $P \to \{0, 1\}$ where P is any partial order ($f : P \to \{0, 1\}$ is monotone if $f(x) \leq f(y)$ whenever $x < y$). Recall that an *antichain* is a set of points $x \in P$ that are incomparable. Observe that a set T is shattered by $\mathcal{M}$ if and only if it is an antichain: a monotone function can take arbitrary values on an antichain, whereas if $x, y \in T$ are comparable, say $x < y$, then $f(x) \leq f(y)$ so T cannot be shattered. Therefore $\mathsf{LVC}_S(\mathcal{M}) = \mathsf{VC}_S(\mathcal{M}) = |S|$ if S is an antichain, and if S is not an antichain then $\mathsf{LVC}_S(\mathcal{M}) = 2$ while $\mathsf{VC}_S(\mathcal{M})$ is the size of the largest antichain in S.

We now turn to the proof of Theorem 2.2.1. The proof uses two main ingredients: lower bounds on the support size estimation problem, and the Sauer–Shelah–Perles theorem.

2.2.1 Ingredient 1: Support Size Distinction

A fundamental problem in the field of distribution testing is *support size estimation*: Given sample access to an unknown finitely-supported distribution $\mathcal{D}$ where each element occurs with probability at least $1/n$ (for some n), estimate the size of the support up to an additive ϵn error. Valiant & Valiant [VV11a, VV11b] showed that for constant ϵ, the number of samples required for this problem is $\Theta\left(\frac{n}{\log n}\right)$. We will adapt this lower bound (in fact an improved version of Wu and Yang [WY19]) to give lower bounds on distribution-free property testing.

Definition 2.2.4 (Support-Size Distinction Problem)**.** For any $n \in \mathbb{N}$ and $0 < \alpha < \beta \leq 1$, define $\mathsf{SSD}(n, \alpha, \beta)$ as the minimum number $m \in \mathbb{N}$ such that there exists an algorithm that for any input distribution p over $[n]$, takes m samples from p and distinguishes with probability at least $2/3$ between the cases:

1. $|\operatorname{supp}(p)| \leq \alpha n$ and $\forall i \in \operatorname{supp}(p), p_i \geq 1/n$; and,
2. $|\operatorname{supp}(p)| \geq \beta n$ and $\forall i \in \operatorname{supp}(p), p_i \geq 1/n$.

Valiant & Valiant [VV11a] and Wu & Yang [WY19] each prove lower bounds on support-size *estimation* and they do so essentially by proving lower bounds on support-size distinction. We note that the bound of [VV11a] holds for $\mathsf{SSD}(n, \alpha, \beta)$ when $1/2 < \alpha < \beta < 1$, but this gap of at most $1/2$ is not sufficient for our purposes, so we use the improved version of [WY19]. However, their lower bound on SSD is not stated explicitly, and therefore we state and prove the following bound explicitly in Appendix A.

Theorem 2.2.5 ([WY19])**.** *There exists a constant C such that, for any $\delta \geq C\frac{\sqrt{\log n}}{n^{1/4}}$ and $\delta \leq \alpha < \beta \leq 1 - \delta$,*

$$\mathsf{SSD}(n, \alpha, \beta) = \Omega\left(\frac{n}{\log n} \log^2 \frac{1}{1-\delta}\right).$$

2.2.2 Ingredient 2: Sauer–Shelah–Perles Lemma

We will need the Sauer–Shelah–Perles lemma (see e.g. [SB14]), for which we recall the following definitions:

Definition 2.2.6. Let $\mathcal{H}$ be a set of functions $\mathcal{X} \to \{0, 1\}$ and let $S \subseteq \mathcal{X}$. We will define the *shattering number* as

$$\mathsf{sh}(\mathcal{H}, S) := |\{T \subseteq S \mid T \text{ is shattered by } \mathcal{H}\}|.$$

We define the *growth function* as

$$\Phi(\mathcal{H}, S) := |\{\ell : S \to \{0, 1\} \mid \exists h \in \mathcal{H}\ \forall x \in S, \ell(x) = h(x)\}|.$$

We state a version of the Sauer–Shelah–Perles lemma that follows from the so-called Sandwich Theorem, rediscovered by numerous authors (see e.g. [Mor12]):

Lemma 2.2.7 (Sauer–Shelah–Perles)**.** *Let $\mathcal{H}$ be a class of functions $\mathcal{X} \to \{0, 1\}$ and let $S \subseteq \mathcal{X}$ with $\mathsf{VC}_S(\mathcal{H}) = d$. Then $\Phi(\mathcal{H}, S) \leq \mathsf{sh}(\mathcal{H}, S) \leq \sum_{i=0}^{d} \binom{|S|}{i}$.*

This lemma gives us a bound on the probability that a random function over a large set is far from the hypothesis class $\mathcal{H}$.

Lemma 2.2.8. *There is a constant $K > 1$ (in particular, $K = 3.04$ suffices) and constants $L > 0, \epsilon_0 > 0$ (depending on K) such that, if $\mathcal{H}$ is a class of functions $\mathcal{X} \to \{0,1\}$ with $\mathsf{VC}(\mathcal{H}) = d$ and $T \subseteq \mathcal{X}$ has size $|T| \geq Kd$, then a uniformly random labelling $\ell : T \to \{0,1\}$ satisfies, with probability at least $1 - e^{-Ld}$, $\forall h \in \mathcal{H} : \mathop{\mathbb{P}}_{x \sim T}[h(x) \neq \ell(x)] > \epsilon_0$.*

Proof. For any $T \subseteq \mathcal{X}$ of size $|T| = m$, and each $h \in \mathcal{H}$, the number of functions $\ell : T \to \{0,1\}$ that differ from h on at most ϵm points of T is at most $\sum_{i=1}^{\epsilon m} \binom{m}{i}$. Therefore, by the Sauer-Shelah-Perles lemma, the number of labellings $\ell : T \to \{0,1\}$ that differs on at most ϵm points from the closest $h \in \mathcal{H}$ is at most

$$\left(\sum_{i=0}^{d} \binom{m}{i}\right) \cdot \left(\sum_{i=0}^{\epsilon m} \binom{m}{i}\right) \leq \left(\frac{em}{d}\right)^d \cdot \left(\frac{em}{\epsilon m}\right)^{\epsilon m} = \left(\frac{em}{d}\right)^d \cdot \left(\frac{e}{\epsilon}\right)^{\epsilon m}.$$

The probability that a uniformly random $\ell : T \to \{0,1\}$ satisfies this condition is therefore at most

$$\begin{aligned}\left(\frac{em}{d}\right)^d \cdot \left(\frac{e}{\epsilon}\right)^{\epsilon m} \cdot 2^{-m} &= (Ke)^d (e/\epsilon)^{K\epsilon d} 2^{-Kd} = 2^{d(\log(Ke) + K\epsilon \log(e/\epsilon) - K)} \\ &= e^{d(\ln(Ke) + K\epsilon \ln(e/\epsilon) - K\ln(2))}.\end{aligned}$$

For any $K > 1$ satisfying $K \ln(2) > 1 + \ln(K)$, there is $L > 0, \epsilon_0 > 0$ such that the exponent $d\left(\ln(Ke) + K\epsilon \ln(e/\epsilon) - K\ln(2)\right) < -Ld$ for all $\epsilon < \epsilon_0$. □

2.2.3 Main Reduction

We now present the main reduction for the proof of Theorem 2.2.1. This reduction is inspired by a proof in the recent work of Epstein & Silwal [ES20]. The reduction can be described intuitively as follows. Suppose there is a class $\mathcal{H}$ of functions $\mathcal{X} \to \{0,1\}$ and a set $S \subseteq \mathcal{X}$ such that are two thresholds $t_1 < t_2$ where:

1. Any set $T \subset S$ of size $|T| \leq t_1$ is shattered by $\mathcal{H}$; and,
2. A random function on any subset $T \subset S$ of size $|T| \geq t_2$ is far from $\mathcal{H}$ with high probability.

Then a distribution-free tester must accept any function (with high probability) when the distribution has support size at most t_1, and reject a random function (with high probability) when the distribution has support size at least t_2. This is made formal in our main lemma:

Lemma 2.2.9. *Let $\mathcal{H}$ be a set of functions $\mathcal{X} \to \{0,1\}$. Suppose $S \subseteq \mathcal{X}$ has size $|S| = n$ and $0 < \alpha < \beta \leq 1$ satisfy the following conditions:*

1. *$\forall T \subset S$ such that $|T| \leq \alpha n$, T is shattered by $\mathcal{H}$; and,*
2. *$\forall T \subseteq S$ such that $|T| \geq \beta n$, a uniformly random labelling $\ell : T \to \{0,1\}$ satisfies with probability at least $9/10$ the condition*

$$\forall h \in \mathcal{H} : \mathop{\mathbb{P}}_{x \sim T} [\ell(x) \neq h(x)] \geq \epsilon/\beta \,.$$

Then $\mathsf{test}_{\mathcal{H}}(\epsilon) = \Omega(\mathsf{SSD}(n, \alpha, \beta))$.

Proof. Let $f : S \to \{0,1\}$ be a uniformly random function, let $\phi : [n] \to S$ be any bijection, and let $\mathcal{D}$ be any distribution over $[n]$ with $\mathcal{D}(x) \geq 1/n$ for all $x \in \mathrm{supp}(\mathcal{D})$. Write $\phi\mathcal{D}$ for the distribution over S of $\phi(x)$ when $x \sim \mathcal{D}$. We make two claims.

First, if $\mathcal{D}$ has support size at most αn then $\mathsf{dist}_{\phi\mathcal{D}}(f, \mathcal{H}) = 0$. Let $T = \mathrm{supp}(\phi\mathcal{D})$. Then since $|T| \leq \alpha n$, by the first condition there exists $h \in \mathcal{H}$ such that $h(x) = f(x)$ on all $x \in T$. So $\mathsf{dist}_{\phi\mathcal{D}}(f, h) = 0$.

Second, if $\mathcal{D}$ has support size at least βn then with probability at least $9/10$ over the choice of f, $\mathsf{dist}_{\phi\mathcal{D}}(f, \mathcal{H}) \geq \epsilon$. Let $T = \mathrm{supp}(\phi\mathcal{D})$ and for any $h \in \mathcal{H}$ write $\Delta(f, h) = \{x \in T : f(x) \neq h(x)\}$. Since $|T| \geq \beta n$ we have by assumption that, with probability at least $9/10$ over the choice of f, for uniform $x \sim T$, $\mathbb{P}[x \in \Delta(f, h)] \geq \epsilon/\beta$. Therefore $|\Delta(f, h)| \geq \frac{\epsilon}{\beta}|T| \geq \epsilon n$. Since $\phi\mathcal{D}(x) = \mathcal{D}(\phi^{-1}(x)) \geq 1/n$ for every $x \in T$, this means that for every $h \in \mathcal{H}$, $\mathop{\mathbb{P}}_{x \sim \phi\mathcal{D}} [f(x) \neq h(x)] \geq \frac{1}{n}|\Delta(f, h)| \geq \epsilon$.

Assume there is a distribution-free tester A that uses m samples. The algorithm for support-size distinction is as follows. Given input distribution $\mathcal{D}$ over $[n]$, choose a uniformly random $f : S \to \{0,1\}$, draw m samples $Q = (x_1, \ldots, x_m)$ from $\phi\mathcal{D}$ and let $Q_f = ((x_1, f(x_1)), \ldots, (x_m, f(x_m)))$; run A on the samples Q_f and accept $\mathcal{D}$ iff A outputs 1.

First suppose that $\mathcal{D}$ has support size at most αn. There exists a function $h \in \mathcal{H}$ with $\mathsf{dist}_{\phi\mathcal{D}}(f, h) = 0$, so $f(x) = h(x)$ for all $x \in \mathrm{supp}(\phi\mathcal{D})$. Therefore the samples Q_f and Q_h

have the same distribution, and the algorithm must output 1 on Q_h with probability at least 5/6, so it must output 1 on Q_f, and therefore accept $\mathcal{D}$, with probability at least 5/6.

Next suppose that $\mathcal{D}$ has support size at least βn. Then the uniformly random function $f : S \to \{0,1\}$ is ϵ-far from $\mathcal{H}$ with respect to $\phi\mathcal{D}$ with probability at least 9/10. Assuming this occurs, algorithm A must output 0 with probability at least 5/6, so $\mathcal{D}$ is rejected with probability at least 2/3. We conclude

$$\mathsf{test}_{\mathcal{H}}(\epsilon) = \Omega(\mathsf{SSD}(n, \alpha, \beta)) . \qquad \square$$

2.2.4 Proof of the Main Lower Bound

Combining Theorem 2.2.5 with Lemma 2.2.8, we obtain the most general form of our main theorem:

Theorem 2.2.10. *Let $\mathcal{H}$ be a class of functions $\mathcal{X} \to \{0,1\}$ and suppose there is a set $S \subseteq \mathcal{X}$ and a value $\delta \in (0, 1/2)$ such that, for $n = |S|$, the following hold:*

1. *$K \cdot \mathsf{VC}_S(\mathcal{H}) \leq (1-\delta)n$, where K is the constant from Lemma 2.2.8; and,*
2. *$\mathsf{LVC}_S(\mathcal{H}) \geq \delta n$; and,*
3. *$\delta \geq C\frac{\sqrt{\log n}}{n^{1/4}}$ where C is the constant from Theorem 2.2.5.*

Let $d := \mathsf{VC}_S(\mathcal{H})$. Then for some constant $\epsilon_0 > 0$ and all $0 < \epsilon < \epsilon_0$,

$$\mathsf{test}_{\mathcal{H}}(\epsilon) = \Omega\left(\frac{n}{\log n}\log^2\frac{1}{1-\delta}\right) .$$

Proof. Let $\alpha := \frac{1}{n}\mathsf{LVC}_S(\mathcal{H}), \beta := \frac{1}{n}K \cdot \mathsf{VC}_S(\mathcal{H})$, so that $\alpha \geq \delta$ and $\beta \leq 1-\delta$. Then from Theorem 2.2.5,

$$\mathsf{SSD}(n, \alpha, \beta) = \Omega\left(\frac{n}{\log n}\log^2\frac{1}{1-\delta}\right) .$$

By definition of LVC, any set $T \subseteq S$ with $|T| \leq \alpha n$ satisfies condition 1 of Lemma 2.2.9, and by Lemma 2.2.8, any set $T \subseteq S$ such that $|T| \geq \beta n = K \cdot \mathsf{VC}_S(\mathcal{H})$ satisfies condition 2 for sufficiently small (constant) $\epsilon > 0$, so by Lemma 2.2.9,

$$\mathsf{test}_{\mathcal{H}}(\epsilon) = \Omega(\mathsf{SSD}(n, \alpha, \beta)) = \Omega\left(\frac{n}{\log n}\log^2\frac{1}{1-\delta}\right) .$$

Finally, since $\frac{1}{1-\delta} \geq 1$ and $n = \Omega(d/(1-\delta)) = \Omega(d)$, we get a bound of $\Omega\left(\frac{d}{\log d}\log^2\frac{1}{1-\delta}\right)$. $\square$

The following simplified bound proves Theorem 2.2.1 from the introduction and will also be used in most of our applications.

Corollary 2.2.11. *There is a constant $L > 0$ such that the following holds. Let $S \subseteq \mathcal{X}$ satisfy $n := |S| \geq 5 \cdot \mathsf{VC}_S(\mathcal{H})$. If $\mathsf{LVC}_S(\mathcal{H}) > L \cdot \mathsf{VC}_S(\mathcal{H})^{3/4}\sqrt{\log \mathsf{VC}_S(\mathcal{H})}$, then*

$$\mathsf{test}_{\mathcal{H}}(\epsilon) = \Omega\left(\frac{\mathsf{LVC}_S(\mathcal{H})^2}{\mathsf{VC}_S(\mathcal{H}) \log \mathsf{VC}_S(\mathcal{H})}\right).$$

Proof. We may assume $n = |S| = 5 \cdot \mathsf{VC}_S(\mathcal{H})$ since by taking subsets of a set S of size larger than $\mathsf{VC}_S(\mathcal{H})$, we do not decrease the LVC dimension and do not increase the VC dimension; we can choose a subset that also does not decrease the VC dimension. We may set $K = 4$ in Theorem 2.2.10. Let $\delta = \frac{\mathsf{LVC}_S(\mathcal{H})}{2K\mathsf{VC}_S(\mathcal{H})}$, so

$$\delta n = \frac{\mathsf{LVC}_S(\mathcal{H})}{2K\mathsf{VC}_S(\mathcal{H})} \cdot 5\mathsf{VC}_S(\mathcal{H}) \leq \mathsf{LVC}_S(\mathcal{H}).$$

We also have $(1-\delta)n \geq \left(1 - \frac{1}{8}\right) 5 \cdot \mathsf{VC}_S(\mathcal{H}) \geq 4\mathsf{VC}_S(\mathcal{H}) = K \cdot \mathsf{VC}_S(\mathcal{H})$. Finally,

$$\delta = \frac{\mathsf{LVC}_S(\mathcal{H})}{8\mathsf{VC}_S(\mathcal{H})} \geq \frac{L\sqrt{\log \mathsf{VC}_S(\mathcal{H})}}{8\mathsf{VC}_S(\mathcal{H})^{1/4}} = \frac{5^{1/4} L \sqrt{\log(n/5)}}{8n^{1/4}},$$

so for large enough constant $L > 0$ this is at least $C\frac{\sqrt{\log n}}{n^{1/4}}$ for the constant C in Theorem 2.2.5, so the conditions for Theorem 2.2.10 are satisfied, and we obtain a lower bound of

$$\Omega\left(\frac{\mathsf{VC}_S(\mathcal{H})}{\log \mathsf{VC}_S(\mathcal{H})} \log^2 \frac{1}{1-\delta}\right).$$

Finally, using the inequality $\log^2 \frac{1}{1-\delta} \geq \log^2(e^\delta) = \Omega(\delta^2)$ we get the conclusion. □

2.3 Application: Geometric Classes

In this section, we use Theorem 2.2.1 to prove lower bounds on the number of samples required to test unions of intervals, halfspaces, and intersections of halfspaces.

Technical note: For the domain $\mathbb{R}^n$, the tester may assume that the distribution $\mathcal{D}$ is defined on the same σ-algebra as the Lebesgue measure. The distributions arising from the above reduction are finitely supported but for the functions considered in this paper, one may replace finitely supported distributions with distributions that are absolutely continuous with respect to the Lebesgue measure without changing the results, by replacing each point in the support with an arbitrarily small ball.

2.3.1 Unions of Intervals

A function $f : \mathbb{R} \to \{0, 1\}$ is a *union of k intervals* if there are k intervals $[a_1, b_1], \ldots, [a_k, b_k]$, where we allow $a_i = -\infty$ and $b_i = \infty$, such that $f(x) = 1$ iff x is contained in some interval $[a_i, b_i]$. Let $\mathcal{I}_k$ denote the class of such functions.

The analysis of the LVC dimension of $\mathcal{I}_k$ is a straightforward variant of the standard analysis of the VC dimension of the class and serves as a good introduction to the high-level structure of the arguments that will be used in later proofs as well.

Proposition 2.3.1. $\mathsf{LVC}_{\mathbb{R}}(\mathcal{I}_k) = \mathsf{VC}_{\mathbb{R}}(\mathcal{I}_k) = 2k$.

Proof. Let $S \subset \mathbb{R}$ have size $2k$ and let $\ell : S \to \{0, 1\}$ be arbitrary. Write $S = \{s_1, \ldots, s_{2k}\}$ where $s_1 < \cdots < s_{2k}$ and partition S into k consecutive pairs (s_i, s_{i+1}) for odd i. Then for each pair (s_i, s_{i+1}) we can choose a single interval that contains exactly the points in s_i, s_{i+1} labelled 1 by ℓ. Therefore S is shattered by k intervals.

On the other hand, let $S \subset \mathbb{R}$ have size $|S| = 2k + 1$, let $s_1 < \cdots < s_{2k+1}$ be the points in S, and suppose $\ell(i) = 1$ iff i is odd. Then any interval can contain at most 1 point of S labelled 1, unless it also contains a 0-point. Therefore S is not shattered. So a set S is shattered iff $|S| \leq 2k$, implying the conclusion. □

Applying Corollary 2.2.11, we obtain:

Theorem 2.3.2. *For some constant $\epsilon > 0$,* $\mathsf{test}_{\mathcal{I}_k}(\epsilon) = \Omega\left(\frac{k}{\log k}\right)$.

2.3.2 Halfspaces

A halfspace is a function $f : \mathbb{R}^n \to \{\pm 1\}$ of the form $f(x) = \operatorname{sign}\left(w_0 + \sum_{i=1}^{n} w_i x_i\right)$ where each $w_i \in \mathbb{R}$. In this subsection, write $\mathcal{L}_n$ for the class of halfspaces (or *Linear threshold functions*) with domain $\mathbb{R}^n$.

The analysis of the LVC dimension follows immediately from the following well-known shattering properties of halfspaces. (See, e.g., [SB14].)

Proposition 2.3.3. *Any set $S \subset \mathbb{R}^n$ of size $n + 1$ in general position can be shattered by $\mathcal{L}_n$, and any set $T \subset \mathbb{R}^n$ of n linearly independent vectors can be shattered by $\mathcal{L}_n$. No set of size $n + 2$ is shattered by $\mathcal{L}_n$.*

Applying Corollary 2.2.11, we obtain our lower bound for domain $\mathbb{R}^n$:

Theorem 2.3.4. *For all small enough constants $\epsilon > 0$, the number of samples required to test the class $\mathcal{L}_n$ of halfspaces over $\mathbb{R}^n$ satisfies*

$$\mathsf{test}_{\mathcal{L}_n}(\epsilon) = \Omega\left(\frac{n}{\log n}\right).$$

Proof. This holds by Corollary 2.2.11, since we may choose any set $S \subset \mathbb{R}^n$ of size $|S| \geq 5(n+1)$ in general position, which by the above proposition satisfies $\mathsf{LVC}_S(\mathcal{L}_n) = \mathsf{VC}_S(\mathcal{L}_n) = n + 1$. □

2.3.3 Intersections of Halfspaces

Let $\mathcal{L}_n^{\cap k}$ denote the class of all Boolean-valued functions obtained by taking the intersections of k halfspaces over $\mathbb{R}^n$. Formally, $\mathcal{L}_n^{\cap k}$ is the set of functions

$$f(x) = h_1(x) \wedge h_2(x) \wedge \cdots \wedge h_k(x)$$

where each h_i is a halfspace. It was recently shown by Csikós, Mustafa, & Kupavskii [CMK19] that the VC dimension of this class is

$$\mathsf{VC}(\mathcal{L}_n^{\cap k}) = \Omega(nk \log k),$$

matching the upper bound given in [BEHW89]. Csikós *et al.* remark that it was long assumed (incorrectly) that the VC dimension of the class was $\Theta(nk)$, which is what one might intuitively expect. We exhibit an infinite set S on which $\mathsf{VC}_S(\mathcal{L}_n^{\cap k}) = \mathsf{LVC}_S(\mathcal{L}_n^{\cap k}) = \Theta(nk)$. We do so with an analysis of alternating functions and polynomial threshold functions.

For any n, define the mapping $\psi : \mathbb{R} \to \mathbb{R}^n$ as follows:

$$\psi_n(x) := \begin{cases} (x, x^2, x^3, \ldots, x^n) & \text{if } n \text{ is even} \\ (0, x, x^2, \ldots, x^{n-1}) & \text{if } n \text{ is odd}. \end{cases}$$

Let $\mathcal{A}_m$ be the set of function $\mathbb{R} \to \{0, 1\}$ that alternate at most m times.

Proposition 2.3.5. *The set $\mathcal{P}$ of functions* $\operatorname{sign}(p(x))$ *on $\mathbb{R}$ where p is a polynomial of degree at most d is equal to the set $\mathcal{A}_d$.*

Proof. This follows from the fact that number of alternations of the function $\text{sign}(p)$ is exactly the number of zeroes of p, which is at most d. On the other hand, any function alternating at most d times may be represented by $\text{sign}(p)$ where p is a polynomial whose zeroes are exactly the points where the function alternates. □

Proposition 2.3.6. *For any even m and any k, $\mathcal{A}_m^{\cup k} = \mathcal{A}_{mk}$.*

Proof. It is clear that the union of k m-alternating functions will alternate at most mk times, so $\mathcal{A}_m^{\cup k} \subseteq \mathcal{A}_{mk}$, so we must show that $\mathcal{A}_{mk} \subseteq \mathcal{A}_m^{\cup k}$. We will do so by induction on k, where the base case $k = 1$ is trivial. For $k > 1$, let $f \in \mathcal{A}_{mk}$ and let $t_1 < \cdots < t_{mk}$ be the alternations (i.e. f is constant on each interval (t_i, t_{i+1}) and $(-\infty, t_1), (t_{mk}, \infty)$). There are two cases: First suppose that the first alternation of $f \in \mathcal{A}_{mk}$ alternates from 0 to 1; or, symmetrically, suppose that the last alternation of f alternates from 1 to 0. Then the function g equal to f on $x \leq t_m$ and 0 on $x > t_m$ is the union of $m/2$ intervals, and $g \in \mathcal{A}_m$. Let f' be 0 on $x \leq t_m$ and equal to f on $x > t_m$, so that f is the union of f' and g, and $f' \in \mathcal{A}_{m(k-1)}$. By induction f' is the union of $k-1$ m-alternating functions, so $f \in \mathcal{A}_m \cup \mathcal{A}_m^{\cup(k-1)} = \mathcal{A}_m^{\cup k}$.

In the second case, the first and last alternations of f alternate from 1 to 0 and 0 to 1, respectively. Let g take value 1 on $(-\infty, t_1], [t_{mk}, \infty)$ as well as on the first $m/2 - 1$ intervals $[t_2, t_3], [t_4, t_5], \ldots, [t_{m-2}, t_{m-1}]$, and 0 otherwise. Then $g \in \mathcal{A}_m$ and the function $f' = f - g$ is in $\mathcal{A}_{m(k-1)}$. So by induction $f' \in \mathcal{A}_m^{\cup(k-1)}$ and $f \in \mathcal{A}_m \cup \mathcal{A}_m^{\cup(k-1)} = \mathcal{A}_m^{\cup k}$. □

Proposition 2.3.7. *For any even m, any k, and any set $S \subseteq \mathbb{R}$ with $|S| > mk$, $\mathsf{VC}_S(\mathcal{A}_m^{\cap k}) = \mathsf{LVC}_S(\mathcal{A}_m^{\cap k}) = mk + 1$.*

Proof. For a class $\mathcal{H}$, write $\overline{\mathcal{H}}$ of the set of functions $f = -g$ where $g \in \mathcal{H}$ (i.e. the set of complements of functions in $\mathcal{H}$). Note that $\overline{\mathcal{A}}_m = \mathcal{A}_m$ since the complement preserves alternations. By De Morgan's laws, $(\overline{\mathcal{H}_n})^{\cap k} = \overline{\mathcal{H}_n^{\cup k}}$. Then $\mathcal{A}_m^{\cap k} = (\overline{\mathcal{A}_m})^{\cap k} = \overline{\mathcal{A}_m^{\cup k}} = \overline{\mathcal{A}_{mk}} = \mathcal{A}_{mk}$. The conclusion follows since $\mathsf{VC}_S(\mathcal{A}_{mk}) = \mathsf{LVC}_S(\mathcal{A}_{mk}) = mk + 1$ by the same argument as for unions of intervals. □

Lemma 2.3.8. *For any $k \geq 1$ and $S \subset \mathbb{R}$ with $|S| > nk + 1$, if n is even then $\mathsf{LVC}_{\psi_n(S)}(\mathcal{L}_n^{\cap k}) = \mathsf{VC}_{\psi_n(S)}(\mathcal{L}_n^{\cap k}) = nk+1$ and if n is odd then $\mathsf{LVC}_{\psi_n(S)}(\mathcal{L}_n^{\cap k}) = \mathsf{VC}_{\psi_n(S)}(\mathcal{L}_n^{\cap k}) = (n-1)k + 1$.*

Proof. First suppose that n is even and consider a halfspace $h(y) = \text{sign}(t + \sum_{i=1}^n w_i y_i)$, where $y = \psi_n(x)$ for some $x \in S$. Then $h(\psi_n(x)) = \text{sign}(t + \sum_{i=1}^n w_i x^i)$, which is the sign of a degree-n polynomial on x. Therefore the set of halfspaces h on the set $\psi(S)$ is equivalent

to the set of degree-n polynomials on S, which by Proposition 2.3.5 is equal to the set of n-alternating functions, so by Proposition 2.3.7 we have $\mathsf{LVC}_{\psi(S)}(\mathcal{L}_n^{\cap k}) = \mathsf{VC}_{\psi(S)}(\mathcal{L}_n^{\cap k}) = \mathsf{VC}(\mathcal{A}_n^{\cap k}) = nk + 1$. When n is odd, the same argument shows that $\mathsf{LVC}_{\psi_n(S)}(\mathcal{L}_n^{\cap k}) = \mathsf{VC}_{\psi_n(S)}(\mathcal{L}_n^{\cap k}) = (n-1)k + 1$. □

Applying Corollary 2.2.11 with a sufficiently large set $S \subset \mathbb{R}$, we obtain the theorem:

Theorem 2.3.9. *For any n, k and sufficiently small constant $\epsilon > 0$*

$$\mathsf{test}_{\mathcal{L}_n^{\cap k}}(\epsilon) = \Omega\left(\frac{nk}{\log(nk)}\right).$$

2.3.4 Decision Trees

For any parameters n and k, let $\mathcal{T}_{n,k}$ denote the set of functions $f : [0,1]^n \to \{0,1\}$ which can be computed by decision trees with at most k nodes, where each node is of the form "$x_i < t$?" for some $t \in \mathbb{R}$.

We can bound the LVC dimension of decision trees using the same argument as for unions of intervals.

Proposition 2.3.10. *Let $S \subset \mathbb{R}^n$ be any subset of the line $\{x \in \mathbb{R}^n : x_2 = \cdots = x_n = 0\}$ with $|S| > k$. Then $\mathsf{LVC}_S(\mathcal{T}_{n,k}) = \mathsf{VC}_S(\mathcal{T}_{n,k}) = k + 1$.*

Proof. Observe that on any sequence $s_1 < s_2 < \cdots < s_m$ in S, any function $f \in \mathcal{T}_{n,k}$ can alternate at most k times, since there are at most k nodes in the decision tree labelled "$x_1 < t$" for some values t. Therefore $T \subseteq S$ is shattered iff $|T| \le k + 1$. □

Combining this proposition with Corollary 2.2.11 completes the proof of the lower bound for testing decision trees:

Theorem 2.3.11. *For any k, n, and small enough constant $\epsilon > 0$, $\mathsf{test}_{\mathcal{T}_{n,k}}(\epsilon) = \Omega\left(\frac{k}{\log k}\right)$.*

2.4 Application: Boolean Functions

The techniques used in the last section do not carry over to classes of functions over the Boolean hypercube. This is because $\{\pm 1\}^n$ is very far from being in general position—indeed, up to 2^{n-1} points can belong to an affine subspace of dimension $n-1$, by, for

example, taking the subspace obtained by setting the first coordinate to 1. In this section, we will instead choose the set S uniformly at random from $\{\pm 1\}^n$ and show that the properties we need for the reduction in Lemma 2.2.9 hold with high probability.

2.4.1 Halfspaces

We first introduce some notation and a theorem that will be used also for PTFs in the next subsection. For a vector $a \in \{0,1\}^n$ and $x \in \mathbb{R}^n$ we will write $x^a = \prod_{i=1}^n x_i^{a(i)}$. Write $|a| = \sum_i a(i)$. Let $\psi_k : \mathbb{R}^n \to \mathbb{R}^{\binom{n}{\leq k}}$ be defined as follows:

$$\psi_k(x) = (x^a)_{a \in \{0,1\}^n : |a| \leq k}\,.$$

We will use the following theorem of Abbe, Shpilka, & Wigderson [ASW15]:

Theorem 2.4.1 ([ASW15])**.** *Let n, k, m be positive integers such that*

$$m < \binom{n - \log\binom{n}{\leq k} - t}{\leq k}\,.$$

Then for independent, uniformly random vectors $x_1, \ldots, x_m \sim \{\pm 1\}^n$, the vectors

$$\psi_k(x_1), \ldots, \psi_k(x_m) \in \{\pm 1\}^{\binom{n}{\leq k}}$$

are linearly independent with probability at least $1 - 2^{-t}$.

Let $\mathcal{L}_n^{\pm}$ denote the set of halfspaces (or linear threshold functions) over $\{\pm 1\}^n$.

Theorem 2.4.2. *For every n and all sufficiently small constant $\epsilon > 0$,*

$$\mathsf{test}_{\mathcal{L}_n^{\pm}}(\epsilon) = \Omega\left(\frac{n}{\log n}\right)\,.$$

Proof. Set $m := 5(n+1), \alpha := 1/11, \beta := 4/5$. We will repeat the reduction from $\mathsf{SSD}(m, \alpha, \beta)$ to testing $\mathcal{L}_n^{\pm}$ as in Lemma 2.2.9 and Theorem 2.2.10 with the fixed set S replaced by a random set S of size m drawn from $\{\pm 1\}^n$. First suppose that the input distribution $\mathcal{D}$ over $[m]$ has support size at most $\alpha m < n/2$. Then $T := \mathrm{supp}(\phi\mathcal{D})$ is a uniformly random subset of $\{\pm 1\}^n$ of size at most $n/2$, so since $|T| \leq n/2 < n - \log(1+n) - C$ for any constant C, by Theorem 2.4.1 (with $k = 1$), the points in T are linearly independent with probability at least $9/10$. In this case, T is shattered by $\mathcal{L}_n^{\pm}$, so the remainder of the proof goes through as in Lemma 2.2.9. When $\mathcal{D}$ has support size at least $\beta m = 4(n+1)$, the proof goes through as in Lemma 2.2.9 and Corollary 2.2.11 with the constant $K = 4$, and we obtain the lower bound. □

2.4.2 Polynomial Threshold Functions

Let $\mathcal{P}_{n,k}$ denote the class of polynomial threshold functions with degree k over $\{\pm 1\}^n$. The above mapping $\psi_k : \{\pm 1\}^n \to \{\pm 1\}^d$ with $d = \binom{n}{\leq k}$ establishes an equivalence between PTFs and halfspaces in a higher dimension:

Lemma 2.4.3. *Write $d := \binom{n}{\leq k}$. A set $S \subseteq \mathbb{R}^n$ is shattered by $\mathcal{P}_{n,k}$ if and only if $\psi_k(S)$ is shattered by $\mathcal{L}_d^{\pm}$.*

Proof. We shall index the coordinates of $\{\pm 1\}^d$ with vectors $a \in \{0,1\}^n$ satisfying $|a| \leq k$. Let $\ell : S \to \{\pm 1\}$ be any labelling of S. Note that ψ_k is a bijection (which can be seen just from the vectors a with $|a| = 1$. If there is a degree-k polynomial $p(x) = \sum_{a \in \{0,1\}^n, |a| \leq k} w_a x^a$ such that $\operatorname{sign}(p(x)) = \ell(x)$ for every $x \in S$, then for every $x \in S$ we have

$$\ell(x) = \operatorname{sign}(p(x)) = \operatorname{sign}\left(w_0 + \sum_{a \in \{0,1\}^n, |a| \leq k} w_a x^a\right) = \operatorname{sign}\left(w_0 + \sum_{a \in \{0,1\}^n, |a| \leq k} w_a \psi_k(x)_a\right).$$

Observe that the function on the right is an LTF in $\mathcal{L}_d^{\pm}$, so there is an LTF consistent with the labelling $\ell \circ \psi_k^{-1}$ on $\psi_k(S)$. So, if S is shattered by $\mathcal{P}_{n,k}$ then $\psi_k(S)$ is shattered by $\mathcal{L}_d^{\pm}$, because ψ_k acts also as a bijection between labellings of S and $\psi_k(S)$. On the other hand, the same equation shows that for any labelling $\ell : \psi_k(S) \to \{\pm 1\}$, if there is an LTF $f : \mathbb{R}^d$ such that $f(\psi_k(x)) = \ell(\psi_k(x))$ for each $x \in \psi_k(S)$ then there is a PTF $g : \mathbb{R}^n \to \{\pm 1\}$ such that $g(x) = f(\psi(x)) = \ell(\psi(x))$ for each $x \in S$. Therefore S is shattered by $\mathcal{P}_k$ iff $\psi_k(S)$ is shattered by $\mathcal{L}_d^{\pm}$. □

Theorem 2.4.4. *Write $\mathcal{P}_{n,k}^{\pm}$ for the set of degree-k PTFs with domain $\{\pm 1\}^n$. There exists some constant C' such that for all $k < n/C'$ and for sufficiently small constant $\epsilon > 0$,*

$$\mathsf{test}_{\mathcal{P}_{n,k}^{\pm}}(\epsilon) = \Omega\left(\frac{\binom{n - \log\binom{n}{k} - O(1)}{\leq k}^2}{\binom{n}{\leq k} \log\binom{n}{\leq k}}\right) = \Omega\left(\frac{(n/4ek)^k}{k \log(n/k)}\right).$$

Proof. Let $d := \binom{n}{\leq k}$ and set $m := 5d$. Let $\beta := 4/5$, $t := \log(10)$, and

$$\alpha := \frac{1}{5}\binom{n}{\leq k}^{-1}\binom{n - \log\binom{n}{\leq k} - t}{\leq k}$$

As was the case with halfspaces, we let S be a uniformly random set of m points drawn from $\{\pm 1\}^n$, let $\phi : [m] \to S$ be a random mapping obtained by assigning a uniform and

independently random $x \in S$ to each $i \in [m]$, and complete the reduction from $\mathsf{SSD}(m, \alpha, \beta)$ to testing $\mathcal{P}_{n,k}$ as in Lemma 2.2.9 and Theorem 2.2.10, which we verify below.

We must first verify that $\alpha \geq C\frac{\sqrt{\log m}}{m^{1/4}}$, where C is the constant in Theorem 2.2.5, for which it suffices to prove that $\alpha \geq \hat{C}\frac{\sqrt{\log d}}{d^{1/4}}$ for a slightly larger $\hat{C} > C$, since $m = 5d$. For an appropriately large choice of constant C', and sufficiently large $n > 2t$,

$$\log \binom{n}{\leq k} + t \leq \log \binom{n}{\leq n/C'} + t \leq \log \left(\left(\frac{en}{n/C'} \right)^{n/C'} \right) + t \leq \log \left((C')^{n/C'} \right) + t$$
$$= \frac{n}{C'} \log(eC') + t \leq n/2\,,$$

so

$$\alpha \geq \frac{1}{5} \binom{n}{\leq k}^{-1} \binom{n/2}{\leq k} \geq \frac{1}{5} \left(\frac{n}{2k} \right)^k \left(\frac{k}{en} \right)^k = \left(\frac{1}{2e} \right)^k .$$

For any constant $\eta > 0$, we may assume $C' > (\hat{C}2e)^{\frac{1}{1/4-\eta}}$, so that, using $\frac{k}{n} \leq \frac{1}{C'} \leq \frac{1}{(C2e)^{\frac{1}{1/4-\eta}}}$, we get

$$\hat{C}\frac{\sqrt{\log d}}{d^{1/4}} \leq C\frac{1}{d^{1/4-\eta}} \leq \hat{C} \left(\frac{k}{n} \right)^{k(1/4-\eta)} \leq \hat{C} \left(\frac{1}{(\hat{C}2e)^{\frac{1}{1/4-\eta}}} \right)^{k(1/4-\eta)} \leq \frac{1}{5} \left(\frac{1}{2e} \right)^k \leq \alpha\,.$$

Now we verify correctness. Suppose that the input distribution $\mathcal{D}$ over $[m]$ has support size at most αm and let $T := \mathrm{supp}(\phi\mathcal{D})$. T is a (multi)set of at most

$$\alpha m = d \binom{n}{\leq k}^{-1} \binom{n - \log \binom{n}{\leq k} - t}{\leq k} = \binom{n - \log \binom{n}{\leq k} - t}{\leq k}$$

uniformly random points from $\{\pm 1\}^n$, so by Theorem 2.4.1 the probability that the points $\psi_k(T)$ are linearly independent is at least $9/10$. In that case, $\psi_k(T)$ is shattered by the halfspaces $\mathcal{H}_d$ over $\{\pm 1\}^d$ so by Lemma 2.4.3, T is shattered by $\mathcal{P}_{n,k}$. Therefore, as in Lemma 2.2.9, the tester for $\mathcal{P}_{n,k}$ will output 1 with probability at least $5/6$, so the distribution $\mathcal{D}$ is accepted with probability at least $2/3$.

Now suppose that the input distribution $\mathcal{D}$ over $[m]$ has support size at least $\beta m = 4d$, and let $T = \mathrm{supp}(\phi\mathcal{D})$. Since ϕ is a random mapping (with replacement), we must first show that, with high probability, $|T| \geq Kd$ for the constant $K > 3.04$ in Lemma 2.2.8. Since $k \leq n/C'$ for a sufficiently large constant C', we have $4d = 4\binom{n}{\leq k} \leq 4(eC')^{n/C'} \leq 2^{cn}$ for constant $c < 1/3$. Therefore the probability that a random point x in T is unique is

at least $1 - \frac{4d}{2^n} \geq 1 - 2^{(c-1)n}$. By the union bound, the probability that any point fails to be unique is at most $4d2^{(c-1)n} = 4\binom{n}{\leq k}2^{(c-1)n} \leq 2^{(2c-1)n} < 2^{-n/3}$. When this occurs, the support of $\phi\mathcal{D}$ has size at least $4d$ so, as in Theorem 2.2.10, we may apply Lemma 2.2.8 to conclude that a random labelling $f : \{\pm 1\}^n \to \{\pm 1\}$ satisfies $\mathsf{dist}_{\phi\mathcal{D}}(f, \mathcal{P}_{n,k}) \geq \epsilon$ with probability at least 9/10, for some small enough constant $\epsilon > 0$. Then the tester for $\mathcal{P}_{n,k}$ will output 0 with probability at least 5/6, so the distribution $\mathcal{D}$ is rejected with probability at least 2/3.

We obtain a lower bound of $\Omega\left(\frac{d}{\log d}\log^2\frac{1}{1-\alpha}\right)$, since $1 - \beta \geq \alpha$. Using the inequality $\log^2\frac{1}{1-x} \geq \log^2\frac{1}{e^{-x}} = \log^2(e^x) = \Omega(x^2)$, we get

$$\frac{d}{\log d}\log^2\frac{1}{1-\alpha} = \Omega\left(\frac{d}{\log d}\alpha^2\right) = \Omega\left(\frac{\binom{n-\log(d)-t}{\leq k}^2}{d\log d}\right).$$

To obtain the simplified bound, use $n - \log(d) - t \leq n/2$ from above, and $\binom{n/2}{\leq k} \geq (n/2k)^k$ to get

$$\Omega\left(\frac{(n/2k)^{2k}}{d\log d}\right) = \Omega\left(\frac{(n/2k)^{2k}}{(en/k)^k k\log(en/k)}\right) = \Omega\left(\frac{(n/4ek)^k}{k\log(n/k)}\right). \qquad \square$$

2.4.3 Decision Trees

Let $\mathcal{B}_{n,k}$ be the set of functions $f : \{0,1\}^n \to \{0,1\}$ defined by decision trees with k nodes of the form "$x_i = 1$?". When $k \gg \log n$, fairly tight bounds on the VC dimension of $\mathcal{B}_{n,k}$ are known.

Lemma 2.4.5 (Mansour [Man97])**.** $\mathsf{VC}(\mathcal{B}_{n,k})$ *is between* $\Omega(k)$ *and* $O(k\log n)$.

A lower bound on the LVC dimension of $\mathcal{B}_{n,k}$ is also easily established.

Proposition 2.4.6. *Every subset* $T \subseteq \{0,1\}^n$ *of size at most* k *is shattered by* $\mathcal{B}_{n,k}$.

Proof. We prove by induction on k that any set $T \subseteq S$ of size k is shattered by a decision tree with at most k leaves. Clearly when $k = 1$, for any subset $T \subseteq S$ of size $|T| = 1$, decision trees with 0 nodes and 1 leaf shatter T. For $k > 1$, there exists a coordinate $i \in [n]$ such that $T_0 := \{x \in T : x_i = 0\} \neq \emptyset$ and $T_1 := \{x \in T : x_i = 1\} \neq \emptyset$. Now T_0 is a subset of size $k - |T_1| < k$ so by induction it is shattered by subtrees with at most $k - |T_1|$ leaves, while T_1 is shattered by subtrees with at most $|T_1|$ leaves. Therefore T is shattered by a tree with at most k leaves. Since the number of nodes is at most the number of leaves, T is shattered by $\mathcal{B}_{n,k}$. $\square$

We are now ready to bound the sample and tolerant-query complexities for testing decision trees.

Theorem 2.4.7. *For any k, $n \geq \log k + \log\log k + \Omega(1)$, and sufficiently small constant $\epsilon > 0$,*

$$\mathsf{test}_{\mathcal{B}_{n,k}}(\epsilon) = \Omega\left(\frac{k}{\log k \cdot \log\log k}\right).$$

Proof. Let $S \subset \{0,1\}^n$ be a subcube with dimension $m := \log(6C) + \log k + \log\log\log k$ and let $d = \mathsf{VC}_S(\mathcal{B}_{n,k})$. Then by Lemma 2.4.5, for some constant C and sufficiently large k,

$$d \leq Ck\log(m) = Ck\log\log(6Ck\log\log k) \leq Ck\log\log(k^2) = Ck(\log\log k + 1)\,,$$

so that

$$\begin{aligned}(1-\delta)|S| = (1-\delta)2^m &= 6Ck\log\log k - k = 5Ck\log\log k + Ck(\log\log k - 1/C)\\ &\geq 5Ck(\log\log k + 1) \geq 5d\,.\end{aligned}$$

By Proposition 2.4.6, $\mathsf{LVC}_S(\mathcal{B}_{n,k}) \geq k$, so for $\delta = \frac{1}{6C\log\log k}$,

$$\mathsf{LVC}_S(\mathcal{B}_{n,k}) \geq k = \delta 6Ck\log\log k = \delta|S|\,,$$

therefore the conditions for Theorem 2.2.10 are satisfied. We obtain a lower bound of

$$\Omega\left(\frac{k\log\log k}{\log k}\log^2\frac{1}{1-\frac{1}{\log\log k}}\right).$$

Using the inequality $\log^2\frac{1}{1-1/x} \geq \log^2\frac{1}{e^{-1/x}} = \log^2(e^{1/x}) = \Omega(1/x^2)$, we get

$$\begin{aligned}\Omega\left(\frac{k\log\log k}{\log k}\log^2\frac{1}{1-\frac{1}{\log\log k}}\right) &= \Omega\left(\frac{k\log\log k}{\log k}\log^2\frac{1}{1-\frac{1}{\log\log k}}\right)\\ &= \Omega\left(\frac{k\log\log k}{(\log k)(\log\log(k))^2}\right)\\ &= \Omega\left(\frac{k}{\log k\cdot\log\log k}\right).\end{aligned}$$

□

2.5 Application: Maximum Classes and Analytic Dudley Classes

In this section we discuss the relation between the LVC dimension and the Sauer–Shelah–Perles lemma, and specifically a refinement of this lemma known sometimes as the sandwich theorem. A special type of hypothesis class that has been well-studied in the literature is called a *maximum class*, which is one where the Sauer–Shelah–Perles lemma is tight in a certain way. It is interesting that the useful condition $\mathsf{LVC}(\mathcal{H}) = \mathsf{VC}(\mathcal{H})$ characterizes the cases where the lemma is tight in another way (explained below). Our main theorem leads to general lower bounds when we can find a large subset $S \subseteq \mathcal{X}$ on which the hypothesis class is maximum. Of special interest are results of Johnson [Joh14] showing that *analytic Dudley classes* are maximum on a large set S, leading us to easily-applicable lower bounds for natural algebraically-defined hypothesis classes. These include PTFs over $\mathbb{R}^d$ as well as trigonometric polynomial thresholds and balls (Theorem 2.5.10).

2.5.1 LVC and the Sauer-Shelah-Perles Lemma

Recall the Sauer-Shelah-Perles lemma and the associated definitions: Let $\mathcal{H}$ be a set of functions $\mathcal{X} \to \{0,1\}$ and let $S \subseteq \mathcal{X}$. The *shattering number* is

$$\mathsf{sh}(\mathcal{H}, S) := \left|\{T \subseteq S \mid T \text{ is shattered by } \mathcal{H}\}\right|,$$

and the *growth function* is

$$\Phi(\mathcal{H}, S) := \left|\{\ell : S \to \{0,1\} \mid \exists h \in \mathcal{H}\ \forall x \in S, \ell(x) = h(x)\}\right|.$$

Sauer-Shelah-Perles lemma. *Let $\mathcal{H}$ be a class of functions $\mathcal{X} \to \{0,1\}$ and let $S \subseteq \mathcal{X}$ with $\mathsf{VC}_S(\mathcal{H}) = d$. Then $\Phi(\mathcal{H}, S) \leq \mathsf{sh}(\mathcal{H}, S) \leq \sum_{i=0}^{d} \binom{|S|}{i}$.*

Much research has studied the cases where this inequality is tight in various ways: A class is called *maximum* on S ([GW94, FW95, KW07, Joh14, AMY16, MW16, CCMW19]) if the sequence of inequalities is tight, i.e. $\mathcal{H}$ is maximum on S if

$$\Phi(\mathcal{H}, S) = \mathsf{sh}(\mathcal{H}, S) = \sum_{i=0}^{d} \binom{|S|}{i}.$$

A class is called *shatter-extremal* on S (see e.g. [Mor12, MW16, CCMW19]) if the first inequality is tight, i.e.

$$\Phi(\mathcal{H}, S) = \mathsf{sh}(\mathcal{H}, S).$$

We are not aware of any studies of the case where the second inequality $\mathsf{sh}(\mathcal{H}, S) \leq \sum_{i=0}^{d} \binom{|S|}{i}$ is tight; our requirement $\mathsf{LVC}_S(\mathcal{H}) = \mathsf{VC}_S(\mathcal{H})$ fills in the gap:

Proposition 2.5.1. *A set $\mathcal{H}$ of functions $\mathcal{X} \to \{0,1\}$ satisfies $\mathsf{LVC}_S(\mathcal{H}) = \mathsf{VC}_S(\mathcal{H})$ on a set $S \subseteq \mathcal{X}$ if and only if $\mathsf{sh}(\mathcal{H}, S) = \sum_{i=0}^{d} \binom{|S|}{i}$, for $d = \mathsf{VC}_S(\mathcal{H})$.*

Proof. This follows from the fact that $\sum_{i=0}^{d} \binom{|S|}{i}$ is exactly the number of sets of size at most d; if the equality holds, all such sets are shattered, so $\mathsf{LVC}_S(\mathcal{H}) = d$. On the other hand if $\mathsf{LVC}_S(\mathcal{H}) = d$ then all sets of size at most d are shattered, so the equality holds. □

We can therefore conclude:

Proposition 2.5.2. *A set $\mathcal{H}$ of functions $\mathcal{X} \to \{0,1\}$ is maximum on $S \subseteq \mathcal{X}$ if and only if it is both shatter-extremal on S and $\mathsf{LVC}_S(\mathcal{H}) = \mathsf{VC}_S(\mathcal{H})$.*

Then we easily obtain lower bounds for maximum classes using Corollary 2.2.11.

Theorem 2.5.3. *Let $\mathcal{H}$ be a set of functions $\mathcal{X} \to \{0,1\}$. Suppose there is $S \subseteq \mathcal{X}$ such that $\mathcal{H}$ is maximum on S and $d := \mathsf{VC}_S(\mathcal{H})$ satisfies $|S| \geq 5d$. Then for sufficiently small constant $\epsilon > 0$,*

$$\mathsf{test}_{\mathcal{H}}(\epsilon) = \Omega\left(\frac{d}{\log d}\right).$$

Examples of maximum classes include the set of functions $f : [n] \to \{0,1\}$ with at most $n/5$ 1-valued points [MW16] (studied in Section 2.7.1), unions of k intervals [Flo89], and positive halfspaces (halfspaces with normal vectors $w \in \mathbb{R}^n$ satisfying $x_i \geq 0$) [FW95]. Another standard example is the set of sign vectors arising from an arrangement of hyperplanes:

Example 2.5.4 ([GW94])**.** Let H be a set of $n > d$ hyperplanes in $\mathbb{R}^d$ and write $H = \{h_1, \ldots, h_n\}$ where each $h_i : \mathbb{R}^d \to \{\pm 1\}$ is of the form $h_i(x) = \mathrm{sign}(t + \sum_{j=1}^{d} w_j x_j)$ for some $t, w_j \in \mathbb{R}$. Assume that the hyperplanes are in general position. Let $\mathcal{H}$ be the set of functions $f_x : [n] \to \{\pm 1\}$ obtained by choosing $x \in \mathbb{R}^d$ obtained by setting $f_x(i) = h_i(x)$. Then $\mathsf{VC}_{[n]}(\mathcal{H}) = d$ and $\mathcal{H}$ is maximum on $[n]$, as proved by Gartner & Welzl [GW94]. Therefore, for any such set $\mathcal{H}$ where $n \geq 5d$ we obtain via Theorem 2.5.3 that $\mathsf{test}_{\mathcal{H}}(\epsilon) = \Omega(d/\log d)$.

2.5.2 Analytic Dudley Classes

Some examples of maximum classes and classes with $\mathsf{LVC}_S(\mathcal{H}) = \mathsf{VC}_S(\mathcal{H})$ that are arguably more pertinent to property testing can be obtained from a family of classes called *Dudley classes* [BL98].

Definition 2.5.5 (Dudley Class)**.** A class $\mathcal{H}$ of functions $\mathcal{X} \to \{\pm 1\}$ is a *Dudley class* if there exists a set $\mathcal{F}$ of functions $\mathcal{X} \to \mathbb{R}$ and a function $h : \mathcal{X} \to \mathbb{R}$ such that:

- $\mathcal{F}$ is a vector space, i.e. $\forall f, g \in \mathcal{F}, \lambda \in \mathbb{R}$, $f + g \in \mathcal{F}$ and $\lambda f \in \mathcal{F}$;
- Every $g \in \mathcal{H}$ can be written as $g(x) = \text{sign}(f(x) + h(x))$.

We will refer to $\mathcal{F}$ as the vector space of $\mathcal{H}$ and h as the threshold of $\mathcal{H}$.

The VC dimension of Dudley classes is equal to the dimension of the vector space $\mathcal{F}$:

Theorem 2.5.6 ([WD81] Theorem 3.1)**.** *Let $\mathcal{H}$ be any Dudley class with vector space $\mathcal{F}$. Then $\mathsf{VC}(\mathcal{H}) = \dim(\mathcal{F})$.*

This theorem implies that $\mathsf{LVC}_S(\mathcal{H}) = \mathsf{VC}_S(\mathcal{H})$ on a set $S \subseteq \mathcal{X}$ if and only if the dimension of the vector space remains the same when restricted to any subset of S:

Corollary 2.5.7. *Let $\mathcal{H}$ be a Dudley class of functions $\mathcal{X} \to \{\pm 1\}$ with vector space $\mathcal{F}$ of functions $\mathcal{X} \to \mathbb{R}$ and threshold h. Then for any set $S \subseteq \mathcal{X}$, $\mathsf{VC}_S(\mathcal{H}) = \mathsf{LVC}_S(\mathcal{H})$ if and only if the vector space $\mathcal{F}$ restricted to any $T \subseteq S$ of size $|T| = d = \mathsf{VC}_S(\mathcal{H})$ has dimension d.*

Proof. This follows from the above theorem, since for any $T \subseteq S$ of size $|T| = d$ on which $\mathcal{F}$ has dimension d, $\mathsf{VC}_T(\mathcal{H}) = d$, so T is shattered. □

A useful condition on Dudley classes that guarantees the above condition was described by Johnson [Joh14]. Recall that a function $f : \mathbb{R}^n \to \mathbb{R}$ is *analytic* if it is infinitely differentiable and for every x in the domain, there is an open set $U \ni x$ such that f is equal to its Taylor series expansion on U. We will call a Dudley class *analytic* if its threshold h and each f in the basis of $\mathcal{F}$ is analytic. Johnson proves the following (rewritten in our terminology):

Theorem 2.5.8 ([Joh14])**.** *Let $\mathcal{H}$ be any analytic Dudley class on domain $[0,1]^n$ with $\mathsf{VC}(\mathcal{H}) = d$. Then for any $N > n$ there exists a set $S \subset [0,1]^n$ of size $|S| = N$ such that $\mathcal{H}$ is maximum on S with $\mathsf{VC}_S(\mathcal{H}) = d$.*

Then by taking $N \geq 5d$ in the above theorem and applying Theorem 2.5.3, we obtain:

Corollary 2.5.9. *Let $\mathcal{H}$ be any analytic Dudley class and suppose $\mathsf{VC}(\mathcal{H}) = d$. Then for some constant $\epsilon > 0$,*

$$\mathsf{test}_{\mathcal{H}}(\epsilon) = \Omega\left(\frac{d}{\log d}\right).$$

Examples of analytic Dudley classes include halfspaces (for which we have already proved the lower bound) and PTFs. Other examples due to [Joh14] are balls in $\mathbb{R}^n$ and trigonometric polynomial threshold functions in $\mathbb{R}^d$:

Theorem 2.5.10. *For sufficiently small constant $\epsilon > 0$, the following classes $\mathcal{H}$ satisfy the given lower bounds for both $\mathsf{test}_{\mathcal{H}}(\epsilon)$:*

1. *Degree-k PTFs on domain $\mathbb{R}^n$ satisfy the lower bound $\Omega\left(\frac{\binom{n+k}{k}}{\log\binom{n+k}{k}}\right)$.*
2. *Balls in $\mathbb{R}^n$, i.e. functions $f : \mathbb{R}^n \to \{\pm 1\}$ of the form $f(x) = \mathrm{sign}(t - \|x - z\|_2)$, satisfy the lower bound $\Omega\left(\frac{n}{\log n}\right)$.*
3. *Signs of trigonometric polynomials, i.e. functions $\mathbb{R}^2 \to \{\pm 1\}$ of the form:*

$$f(x, y) = \mathrm{sign}\left(t + \sum_{k=1}^{d} a_k \cos(kx) + \sum_{k=1}^{d} b_k \sin(kx) - y\right),$$

which satisfy the lower bound $\Omega\left(\frac{d}{\log d}\right)$.

2.6 Application: Other Models of Testing

Now we apply our techniques to a couple other models of testing. Specifically, we obtain the first lower bounds for two-sided error testers for testing k-Clusterability (as introduced in [ADPR03]) and testing feasibility of linear programs (as introduced in [ES20]).

2.6.1 Testing Clusterability

For a point $x \in \mathbb{R}^n$ and radius $r > 0$, define $B_r(x) = \{y \in \mathbb{R}^n : \|x - y\|_2 \leq r\}$. Alon, Dar, Parnas, & Ron [ADPR03] introduced the problem of testing clusterability with radius cost:

Definition 2.6.1 (Radius Clustering)**.** Say that a probability distribution $\mathcal{D}$ over $\mathbb{R}^n$ is *k-clusterable* if there exist k centers $c_1, \ldots, c_k \in \mathbb{R}^n$ such that $\mathrm{supp}(\mathcal{D}) \subseteq \cup_{i=1}^k B_1(c_i)$. An ϵ-tester for k-clusterability is a randomized algorithm A that is given sample access to $\mathcal{D}$ and must satisfy the following:

1. If $\mathcal{D}$ is k-clusterable then $\mathbb{P}[A(\mathcal{D}) = 1] \geq 2/3$; and,
2. If $\mathcal{D}$ is ϵ-far from being k-clusterable in total variation distance, then $\mathbb{P}[A(\mathcal{D}) = 0] \geq 2/3$.

Alon *et al.* [ADPR03] prove an upper bound of $O\left(\frac{nk\log(nk)}{\epsilon}\right)$ samples for one-sided testing of k-clusterability when the distribution is uniform over an unknown set of points. The following theorem updates the upper bound of [ADPR03] using modern VC dimension results; it follows from the same ϵ-net argument found in Theorem 2.1.1 (see also [Har14]), and the fact that the VC dimension of unions of k balls is at most $O(nk\log k)$ [CMK19].

Theorem 2.6.2 (Improved version of [ADPR03])**.** *There is a one-sided, distribution-free ϵ-tester for k-clusterability in $\mathbb{R}^n$ with sample complexity $O\left(\frac{nk\log k}{\epsilon}\log\frac{1}{\epsilon}\right)$.*

We give a lower bound for two-sided error testers that is tight up to poly-log factors for all values of k up to $2^{n/6}$. Let $S_r^n = \{x \in \mathbb{R}^n : \|x\|_2 = r\}$ be the points on the hypersphere of radius r.

Proposition 2.6.3. *For every $\delta > 0$ there is $\eta > 0$ such that a uniformly random set of n points P drawn from $S_{1+\eta}^n$ is contained within some ball $B_1(x)$ with probability at least $1 - \delta$.*

Proof. Unless all n points in P lie on a hyperplane through the origin (which occurs with probability 0), there is a hyperplane through the origin such that all points in P lie on one side. Consider the distribution of P conditional on this event, and without loss of generality assume that the hyperplane is $\{x : x_1 = 0\}$ so that all points $x \in P$ satisfy $x_1 > 0$. Let $\eta > 0$ and consider the ball B of radius 1 centered at $z = (\sqrt{(1+\eta)^2 - 1}, 0, \ldots, 0)$. Let $x \in S_{1+\eta}^n$ satisfy $x_1 \geq z_1 = \sqrt{(1+\eta)^2 - 1} = \sqrt{\eta(2-\eta)}$. Then since $\|x\|_2^2 = (1+\eta)^2$,

$$\begin{aligned}\|x - z\|_2^2 &= (x_1 - z_1)^2 + \sum_{i=2}^n x_i^2 = (x_1 - z_1)^2 + (1+\eta)^2 - x_1^2 \\ &= z_1^2 - 2x_1 z_1 + (1+\eta)^2 \leq (1+\eta)^2 - z_1^2 = 1\,,\end{aligned}$$

so all points x with $x_1 \geq z_1$ are contained within the ball B. Conditioned on $x_1 > 0$, the probability that $x_1^2 \geq \eta(2-\eta)$ is at least the probability that $y_1^2 \geq \eta(2-\eta)$ for y drawn uniformly randomly from S_1^n. This probability goes to 1 as $\eta \to 0$, so the probability that $x_1^2 \geq \eta(2-\eta)$ also approaches 1 as $\eta \to 0$. The conclusion follows. □

Proposition 2.6.4. *For every constant $\delta, \eta > 0$, there is a constant $\epsilon_0 > 0$ such that, for all $\epsilon < \epsilon_0$ and for a uniformly random set P of $m = 2n$ points drawn from $S_{1+\eta}^n$, with probability at least $1 - e^{-\delta n}$, no subset $T \subset P$ of size $(1-\epsilon)m$ is contained within a ball of radius 1.*

Proof. Let $t = (1-\epsilon)m > n$ and let $T \subset P$ have size $|T| = t$. If the points T are contained within a ball of radius 1 then they are contained within a centered halfspace, because the intersection of the ball with $S_{1+\eta}^n$ is equal to the intersection of some halfspace with $S_{1+\eta}^n$. The probability that t uniformly random points on the surface of the sphere lie within some hemisphere is $2^{1-t}\sum_{k=0}^{n-1}\binom{t-1}{k}$ [Wen62]. There are at most $\binom{m}{t}$ subsets of size t, so the probability that any of these subsets lie within a hemisphere is at most

$$\begin{aligned}
\binom{m}{m-t} 2^{1-t} \left(\frac{et}{n}\right)^n &\leq 2^{1-(1-\epsilon)m} \left(\frac{e}{\epsilon}\right)^{\epsilon m} \left(\frac{et}{n}\right)^n \\
&= 2^{1+\epsilon m \log(e/\epsilon) - (1-\epsilon)m + n\log\left(\frac{e(1-\epsilon)m}{n}\right)} \\
&= 2^{1+\epsilon 2n \log(e/\epsilon) - (1-\epsilon)2n + n\log(e(1-\epsilon)2)} \\
&\leq 2^{1-2n(1-\epsilon\log(4e^2/\epsilon))}.
\end{aligned}$$

The conclusion holds since $\epsilon \log(4e^2/\epsilon) \to 0$ as $\epsilon \to 0$. □

Proposition 2.6.5 (Balls and bins)**.** *Fix $C > 0$, $0 < \delta \leq 1$, and let n, k be positive integers with $k \leq \frac{1}{10}e^{\delta^2 Cn/3}$. Then if Cnk balls are deposited into k bins uniformly at random, the following hold:*

1. *With probability at least $9/10$, every bin receives at most $(1+\delta)Cn$ balls;*
2. *With probability at least $9/10$, every bin receives at least $(1-\delta)Cn$ balls.*

Proof. Let X_{ij} be the indicator variable for the event that the i-th ball goes into the j-th bin, and let the random variable $L_j = \sum_{i=1}^{Cnk} X_{ij}$ denote the final load on the j-th bin. Note that $\mathbb{E}[L_j] = Cn$. By the multiplicative Chernoff bound, we have:

1. $\mathbb{P}[L_j \geq (1+\delta)Cn] \leq e^{-\delta^2 Cn/3}$; and

2. $\mathbb{P}[L_j \leq (1-\delta)Cn] \leq e^{-\delta^2 Cn/3}$.

In both cases, by the union bound, the probability that the respective event occurs for any L_j $(1 \leq j \leq k)$ is at most $k \cdot e^{-\delta^2 Cn/3} \leq 1/10$, as desired. □

Lemma 2.6.6. *For $k < \frac{1}{10}e^{n/6}$, let $A_1, \ldots, A_k$ be spheres in $\mathbb{R}^n$ of radius $1+\eta$ for sufficiently small $\eta > 0$, such that the minimum distance between any two spheres is 3. Define the following distribution $\mathcal{S}$ over $\bigcup_{i=1}^n A_i$: Draw $i \in [k]$ uniformly at random and then draw $x \sim A_i$ uniformly at random. Then:*

1. *If S is a set of $m \leq nk/2$ independent points drawn from $\mathcal{S}$, then with probability at least $9/10$, there are k balls of radius 1 whose union contains S;*
2. *If S is a set of $4nk \leq m \leq 8nk$ independent points drawn from $\mathcal{S}$ and $\epsilon > 0$ is a sufficiently small constant, then with probability at least $81/100$, no union of k balls of radius 1 contains more than $(1-\epsilon)m$ points of S.*

Proof. First suppose that $m \leq nk/2$. If each sphere A_i receives at most n sample points then by Proposition 2.6.3, setting $\delta, \eta > 0$ arbitrarily small in the statement of that proposition, for each sphere A_i there is a ball B_i of radius 1 containing all points $S \cap A_i$ with probability arbitrarily close to 1, so there are k balls containing all points of S. Proposition 2.6.5 (with $C = 1/2$ and $\delta = 1$) shows that the maximum load of any sphere is at most n with probability at least $9/10$, so the first conclusion holds.

Now suppose that $4nk \leq m \leq 8nk$. Note that no ball of radius 1 can contain points from more than 1 sphere A_i. Proposition 2.6.5 (with $C = 4$ and $\delta = 1/2$) shows that the minimum load of any sphere is at least $2n$ with probability at least $9/10$. Assume that this occurs for the rest of this argument.

Let $S_i = S \cap A_i$ for $i = 1, \ldots, k$, and say that S_i is *difficult* if no ball of radius 1 contains at least $(1-\epsilon')|S_i|$ points in S_i, for constant ϵ' to be defined. Since $|S_i| \geq 2n$, Proposition 2.6.4 gives that $\mathbb{P}[S_i \text{ is difficult}] \geq 1 - e^{-\delta n}$. Setting $\delta = 1/6$ and by the union bound, the probability that every S_i is difficult is at least $1 - k \cdot e^{-\delta n} \geq 1 - \frac{1}{10}e^{n/6}e^{-\delta n} = 9/10$. Fix ϵ' corresponding to $\delta = 1/6$ in Proposition 2.6.4.

Assume that every S_i is difficult, and consider any set of k balls $B_1, \ldots, B_k$. Denote their union by $B = \bigcup_i B_i$. Then for each S_i, we have that $|B \cap S_i| \geq (1-\epsilon')|S_i|$ only if at least two balls B_{j_1}, B_{j_2} intersect S_i. Thus, this can only happen for at most $k/2$ such

S_i's. Assume without loss of generality that $S_1, \ldots, S_\ell$ have at least $(1-\epsilon')$-fraction of their points covered by B, so that $\ell \le k/2$. It follows that

$$|S \setminus B| \ge \sum_{i=\ell+1}^{k} \epsilon' |S_i| \ge \frac{k}{2} \cdot \epsilon' \cdot 2n \ge \frac{\epsilon' m}{8}.$$

Which satisfies the second claim for $\epsilon = \epsilon'/8$, and this happens with probability at least $9/10 \cdot 9/10 = 81/100$ over the choice of S. □

Theorem 2.6.7. *For sufficiently small constant $\epsilon > 0$, any ϵ-tester for k-clusterability in $\mathbb{R}^n$ requires at least $\Omega\left(\frac{nk}{\log(nk)}\right)$ samples.*

Proof. Let $N = 8nk$ and let $\alpha = 1/16, \beta = 1/2$. We will prove a reduction from support-size distinction to k-clusterability; we may assume that the tester for k-clusterability has success probability at least $5/6$ due to standard boosting techniques. For an input distribution $\mathcal{D}$ over $[N]$ with densities at least $1/N$, construct spheres $A_1, \ldots, A_k$ as in Lemma 2.6.6. Construct the map $\phi : [N] \to \bigcup_{i=1}^k A_i$ by sampling $s_1, \ldots, s_N \sim \mathcal{S}$, where $\mathcal{S}$ is the distribution from Lemma 2.6.6, and setting $\phi(i) = s_i$. Then simulate the tester for k-clusterability by giving the tester samples $\phi(i)$ for $i \sim \mathcal{D}$. We will write $\phi\mathcal{D}$ for the distribution over $\bigcup_{i=1}^k A_i$ obtained by sampling $i \sim \mathcal{D}$ and returning $\phi(i)$.

First suppose that $|\operatorname{supp}(\mathcal{D})| \le \alpha N$. Then $\operatorname{supp}(\phi\mathcal{D})$ is a set of at most $\alpha N = nk/2$ points sampled from $\mathcal{S}$, so by Lemma 2.6.6, with probability at least $9/10$ over the choice of ϕ the distribution $\phi\mathcal{D}$ is k-clusterable, so the tester will output 1 with probability at least $5/6$, so the total probability of success is at least $2/3$.

Next suppose that $|\operatorname{supp}(\mathcal{D})| \ge \beta N$ so $\operatorname{supp}(\phi\mathcal{D})$ is a set of between $\beta N = 4nk$ and $N = 8nk$ points sampled from $\mathcal{S}$. Then by Lemma 2.6.6, for sufficiently small constant $\epsilon > 0$, with probability at least $81/100$ over the choice of ϕ, $X := \operatorname{supp}(\phi\mathcal{D})$ is at least ϵ/β-far from k-clusterable according to the uniform distribution over X. Since $\mathcal{D}$ (and therefore $\phi\mathcal{D}$) has densities at least $1/N$ on X, any k-clusterable distribution $\phi\mathcal{D}$ must be at least $\frac{(\epsilon/\beta)|X|}{N} \ge \epsilon$-far from $\phi\mathcal{D}$. Therefore the ϵ-tester will output 0 with probability at least $5/6$, so the total probability to output 0 is at least $2/3$. So the algorithm solves support-size distinction with parameters $N = 8nk, \alpha = 1/16, \beta = 1/2$. Finally, by Theorem 2.2.5, the number of samples required is at least $\Omega\left(\frac{N}{\log N}\right) = \Omega\left(\frac{nk}{\log(nk)}\right)$. □

2.6.2 Uniform Distributions and Testing LP-Type Problems

Epstein & Silwal [ES20] recently introduced property testing for LP-Type problems, which are problems that generalize linear-programming. The algorithm has query access to a set

S of constraints and must determine with high probability whether an objective function ϕ satisfies $\phi(S) \leq k$ or if at least an ϵ-fraction of constraints must be removed in order to satisfy $\phi(S) \leq k$. We refer the reader to their paper for the definition of their model and results in full generality, and describe only a special case here.

Definition 2.6.8 (Testing Feasibility [ES20])**.** A tester for feasibility of a set of linear equations is an algorithm that performs as follows. On an input set S of linear equations over $\mathbb{R}^n$, the algorithm samples equations $s \sim S$ uniformly at random, and must satisfy the following:

1. If S is feasible, i.e. there exists $x \in \mathbb{R}^n$ that satisfies all equations S, then the algorithm outputs 1 with probability at least 2/3;
2. If at least $\epsilon|S|$ equations must be removed or flipped for the system to be feasible, then the algorithm outputs 0 with probability at least 2/3.

Epstein & Silwal obtain a two-sided tester for this problem.

Theorem 2.6.9 ([ES20])**.** *There is a tester for feasibility in $\mathbb{R}^n$ with two-sided error and sample complexity $O(n/\epsilon)$.*

Testing if a set $X \subseteq \mathbb{R}^n$ with labels $\ell : X \to \{\pm 1\}$ is realizable by a halfspace can be solved by their algorithm, since for each $x \in X$ one can add the constraint $\ell(x) \cdot (w_0 + \sum_{i=1}^n w_i x_i) \geq 1$ to S, with variables $w_0, w_1, \ldots, w_n$. On the other hand, they prove a lower bound for one-sided error:

Theorem 2.6.10 ([ES20])**.** *Testing with one-sided error whether a set $X \subseteq \mathbb{R}^n$ with labels $\ell : X \to \{\pm 1\}$ is realizable by a halfspace or whether at least $\epsilon|X|$ labels must be changed to become realizable by a halfspace requires at least $\Omega(d/\epsilon)$ samples.*

Remark 2.6.11. [ES20] does not specify that their lower bound is for one-sided error; however, their proof relies on a claim that is true only for one-sided error [Sil20], namely that distinguishing between uniform distributions with support size d and uniform distributions with support size $3d$ requires at least $d + 1$ samples – with two-sided error, this can be done with only $O(\sqrt{d})$ samples via a birthday paradox argument.

We would like to prove lower bounds on two-sided error algorithms. However, our reduction from support-size distinction will not work for this, because the model of LP testing uses the uniform distribution as its distance measure, and the distributions that occur in the reduction are not uniform. We can fix this by replacing the lower bound of Wu & Yang [WY19] with a weaker lower bound of [RRSS09] that uses distributions $\mathcal{D}$ over $[n]$ with densities that are integer multiples of $1/n$:

Theorem 2.6.12 ([RRSS09] Theorem 2.1)**.** *Let* $\mathsf{SSD}^{\mathbb{Z}}(n, \delta, 1-\delta)$ *be the support-size distinction problem under the promise that the input distribution* $\mathcal{D}$ *has densities that are integer multiples of* $1/n$*. Then for every* $\delta \geq 2\frac{\sqrt{\log n}}{n^{1/4}}$*,*

$$\mathsf{SSD}^{\mathbb{Z}}(n, \delta, 1-\delta) = \Omega(n^{1-\gamma}),$$

where $\gamma = 2\sqrt{\frac{\log(1/\delta)+\frac{1}{2}\log\log(n)+1}{\log n}}$*. In particular, for constant* δ*, the lower bound is* $n^{1-o(1)}$*.*

We can now prove the following lower bound on testing linear separability:

Theorem 2.6.13. *Testing with two-sided error whether a set* $X \subseteq \mathbb{R}^n$ *with labels* $\ell : X \to \{\pm 1\}$ *is realizable by a halfspace or whether at least* $\epsilon|X|$ *labels must be changed to become realizable by a halfspace requires at least* $n^{1-o(1)}$ *samples.*

Proof. Repeat the proof of Theorem 2.2.10 and Lemma 2.2.9 with input distributions $\mathcal{D}$ over $[n]$ where for each $i \in \mathrm{supp}(\mathcal{D}), \mathcal{D}(i)$ is an integer multiple of $1/n$. We obtain a set of points $X = \mathrm{supp}(\phi\mathcal{D}) \subseteq \mathbb{R}^n$ and labels $\ell : X \to \{\pm 1\}$ with integer probabilities, and we let S be the set of linear constraints constructed from X, ℓ as above, with each $x \in S$ occurring with multiplicity t when $\mathcal{D}(\phi^{-1}(x)) = t/n$. We may simulate samples from S by samples from $\mathcal{D}$. Therefore we obtain a lower bound of $n^{1-o(1)}$ by the theorem of [RRSS09]. □

2.7 Upper Bounds

In this section we will prove upper bounds to complement the above lower bounds. First we study symmetric classes of functions, which have been noted by Sudan [Sud10] and Goldreich & Ron [GR16] to be closely related to support size estimation. These classes establish the optimality of Theorem 2.2.1, and show that the lower bound cannot in general be extended to testers in the query model (on the other hand, follow-up work of Chen & Patel [CP22] has shown that the lower bound *can* be extended to the query model for halfspaces).

Next, we show that there are natural classes of Boolean-valued functions, k-juntas and monotone functions for which efficient distribution-free sample-based testing is possible.

2.7.1 Symmetric Classes

We first show that our lower bound in Theorem 2.2.10 is optimal, in the sense that there exists a property $\mathcal{H}$ where $d = \mathsf{LVC}_S(\mathcal{H}) = \mathsf{VC}_S(\mathcal{H}) \leq |S|/5$ and the sample complexity

of distribution-free testing is $\Theta(d/\log d)$. The upper bound will follow from a theorem of Goldreich & Ron [GR16] for symmetric properties.

Definition 2.7.1. A set $\mathcal{H}$ of functions $[n] \to \{0,1\}$ is *symmetric* if for any permutation $\sigma : [n] \to [n]$, for any function $f \in \mathcal{H}$, it is also the case that $f \circ \sigma \in \mathcal{H}$. Equivalently, $\mathcal{H}$ is symmetric iff there is a function $\phi : [n] \to \{0,1\}$ such that $f \in \mathcal{H}$ iff $\phi(k) = 1$ when $k = |\{i \in [n] : f(i) = 1\}|$.

Proposition 2.7.2. *Let $\mathcal{H}$ be any symmetric class of functions $[n] \to \{0,1\}$. Then for any set $S \subseteq [n]$, $\mathsf{LVC}_S(\mathcal{H}) = \mathsf{VC}_S(\mathcal{H})$.*

Proof. This follows from the fact that if $T \subseteq S$ is shattered by $\mathcal{H}$, then every $T' \subseteq S$ with $|T'| = |T|$ is also shattered. □

Symmetric properties are interesting because, as observed by Goldreich & Ron [GR16], there is a distribution-free testing upper bound for these sets that can be obtained by the support-size estimation algorithm of Valiant & Valiant [VV11a, VV11b]. Together with our lower bound, this shows that distribution-free testing symmetric sets $\mathcal{H}$ is essentially equivalent to deciding support size.

Theorem 2.7.3 (Goldreich & Ron [GR16], Claim 7.4.2)**.** *For any symmetric class $\mathcal{H}$ of functions $[n] \to \{0,1\}$, $\mathsf{test}_{\mathcal{H}}(\epsilon) = \mathrm{poly}(1/\epsilon) \cdot O\left(\frac{n}{\log n}\right)$.*

On the other hand, consider the class $\mathcal{S}_n$ of functions $[n] \to \{0,1\}$ such that $f : [n] \to \{0,1\}$ is in $\mathcal{S}_n$ iff $|\{i \in [n] : f(i) = 1\}| \leq n/5$.

Theorem 2.7.4. *$\mathsf{LVC}_{[n]}(\mathcal{S}_n) = \mathsf{VC}_{[n]}(\mathcal{S}_n) = n/5$ and for small enough constant $\epsilon > 0$,*

$$\mathsf{test}_{\mathcal{S}_n}(\epsilon) = \Theta\left(\frac{n}{\log n}\right).$$

Proof. Any negative certificate for a function $f \notin \mathcal{S}_n$ must have size at least $n/5 + 1$ so $\mathsf{LVC}_{[n]}(\mathcal{S}_n) \geq n/5$. On the other hand, any set T of size $n/5$ is shattered since we may assign 0 to all values $[n] \setminus T$. Therefore Corollary 2.2.11 and Theorem 2.7.3 imply the conclusion. □

Next we show that the lower bound for $(0,\epsilon)$-tolerant adaptive testers cannot be extended to intolerant testers. Parnas, Ron, & Rubinfeld [PRR06] observed that, when

testing over the uniform distribution, any ϵ-tester with uniformly (but not necessarily independently) distributed queries is in fact (ϵ', ϵ)-tolerant for some $\epsilon' > 0$ (depending on the query cost). Since our lower bound in Theorem 2.2.10 holds even for $(0, \epsilon)$-tolerant testers, one might then wonder if it holds also for intolerant testers, in light of the observation of [PRR06]. However, this is not the case, and the counterexample is the same class of symmetric functions discussed above. This class exhibits a nearly-maximal separation between tolerant and intolerant testing in the distribution-free model, even when the intolerant tester has uniformly distributed and non-adaptive queries.

Theorem 2.7.5. *There is a two-sided non-adaptive query tester for $\mathcal{S}_n$ with query complexity $O\left(\frac{1}{\epsilon^2}\right)$, and yet every adaptive $(0, \epsilon)$-tolerant tester for $\mathcal{S}_n$ has query complexity $\Omega\left(\frac{n}{\log n}\right)$ for small enough constant $\epsilon > 0$.*

Proof. The lower bound follows from Corollary 2.2.11. For the upper bound, consider the algorithm that makes $m = \frac{50}{\epsilon^2}\ln(3)$ uniformly random samples, sets X equal to the number of sample points with value 1, and rejects iff $X > (1 + \epsilon/2)\frac{m}{5}$.

Let $\mathcal{D}, f$ be the input distribution and function, and suppose that $f \in \mathcal{S}_n$. Since the queries are made uniformly at random, it follows that $\mathbb{E}[X] \leq n/5$. Thus by Hoeffding's inequality,

$$\mathbb{P}\left[X > (1 + \epsilon/2)\frac{m}{5}\right] \leq \mathbb{P}\left[X > \mathbb{E}[X] + \frac{\epsilon m}{10}\right] \leq \exp\left(-\frac{m\epsilon^2}{50}\right) \leq 1/3\,.$$

Now suppose that $\mathsf{dist}_{\mathcal{D}}(f, \mathcal{H}) > \epsilon$. Let $N = \{x \in [n] : f(x) = 1\}$ and observe that $|N| > n/5$. Write $\mathcal{D}(x)$ for the probability density of x according to $\mathcal{D}$, let $A \subset N$ be the $n/5$ points $x \in N$ with largest value $\mathcal{D}(x)$, and let $B = N \setminus A$. Observe that for all $x \in A, y \in B, \mathcal{D}(x) > \mathcal{D}(y)$, so the average $\mathcal{D}(x)$ in A is larger than the average $\mathcal{D}(x)$ in B. Write $\mathcal{D}(A) := \sum_{x \in A} \mathcal{D}(x), \mathcal{D}(B) := \sum_{x \in B} \mathcal{D}(x)$. Since $\mathsf{dist}_{\mathcal{D}}(f, \mathcal{H}) > \epsilon$ it must be that $\mathcal{D}(B) > \epsilon$ since otherwise the function f' obtained by flipping the values in B is in $\mathcal{H}$ and satisfies $\mathsf{dist}_{\mathcal{D}}(f, \mathcal{H}) \leq \mathsf{dist}_{\mathcal{D}}(f, f') \leq \epsilon$.

$$\frac{5}{n} \geq \frac{\mathcal{D}(A)}{|A|} \geq \frac{\mathcal{D}(B)}{|B|} \geq \frac{\epsilon}{|B|}$$

so $|B| \geq \epsilon n/5$. Therefore the number of 1-valued points according to f is $|N| = |A| + |B| \geq (1 + \epsilon)\frac{n}{5}$, so $\mathbb{E}[X] \geq (1 + \epsilon)\frac{m}{5}$. By Hoeffding's inequality:

$$\mathbb{P}\left[X \leq (1 + \epsilon/2)\frac{m}{5}\right] \leq \mathbb{P}\left[X \leq \mathbb{E}[X] - \frac{\epsilon m}{10}\right] \leq \exp\left(-\frac{m\epsilon^2}{50}\right) \leq 1/3\,. \qquad \square$$

2.7.2 k-Juntas

A k-junta $\{0,1\}^n \to \{0,1\}$ on n variables is a function that depends on only k of the n variables; these are of great interest in testing and learning because if a function depends on $k \ll n$ variables then the complexity of learning may be significantly reduced. Blais [Bla09] gave a nearly optimal tester in the query model for product distributions, and Bshouty [Bsh19] recently presented a tester in the distribution-free query model, with query cost $\widetilde{O}(k/\epsilon)$, but there are no known upper bounds in the sample-based distribution-free model. The VC dimension of k-juntas is at least 2^k since any subcube of dimension k can be shattered. We prove a polynomial improvement over the VC dimension for distribution-free sample-based testers when $k > \log\log n$ using the following version of the birthday problem.

Proposition 2.7.6. *Let p be any distribution over $[n]$. The probability that m independent samples drawn from p are all distinct is at most $e^{-\frac{(m-1)^2}{2n}}$.*

Proof. It is known that the worst case probability distribution p is uniform over $[n]$ [Mun77]. For the uniform distribution over $[n]$, the probability that all m independent samples are distinct is at most

$$\prod_{i=0}^{m-1}\left(1-\frac{i}{n}\right) \le \prod_{i=0}^{m-1} e^{-\frac{i}{n}} = \exp\left(-\frac{1}{n}\sum_{i=0}^{m-1} i\right) = \exp\left(-\frac{m(m-1)}{2n}\right). \qquad \square$$

Theorem 2.7.7. *There is a distribution-free sample-based ϵ-tester for k-juntas on domain $\{0,1\}^n$ with one-sided error and sample complexity $O\left(\frac{k2^{k/2}\log(n/k)}{\epsilon}\right)$.*

Proof. For a set $S \subseteq [n]$ of size $n-k$ we will arrange the points $x \in \{0,1\}^n$ into "rows" and "columns"; for every partial assignment $\rho : \overline{S} \to \{0,1\}$ let row R_ρ be the set of points $x \in \{0,1\}^n$ such that $\forall i \notin S, x_i = \rho(i)$, and for every partial assignment $\gamma : S \to \{0,1\}$ let column C_γ be the set of points $x \in \{0,1\}^n$ such that $\forall i \in S, x_i = \gamma(i)$.

The tester is as follows: On input $f : \{0,1\}^n \to \{0,1\}$ and distribution p, sample a set Q of $s \cdot m$ points, where $s = O\left(\log\binom{n}{k}\right)$ and $m = O\left(\frac{2^{k/2}}{\epsilon}\right)$; since $\binom{n}{k} \le \left(\frac{en}{k}\right)^k$, the sample complexity is $sm = O\left(\frac{k2^{k/2}\log(n/k)}{\epsilon}\right)$. Reject if for every set $S \subset [n]$ of $n-k$ variables, there exists a row $\rho : \overline{S} \to \{0,1\}$ that contains $x, y \in Q \cap R_\rho$ such that $f(x) \neq f(y)$; we will call such a pair x, y a *witness* for S. This has one-sided error because a k-junta has a set S of variables such that f is constant on every row.

Let $p : \{\pm 1\}^n \to \mathbb{R}$ be a probability distribution over $\{\pm 1\}^n$. Let $f : \{0,1\}^n \to \{0,1\}$ and let $S \subset [n]$ be a set of $n-k$ variables. For each $\rho : \overline{S} \to \{0,1\}$ write

$$r^0_\rho := \sum_{x \in R_\rho : f(x)=0} p(x) \qquad\qquad r^1_\rho := \sum_{x \in R_\rho : f(x)=1} p(x)$$

Suppose that

$$\sum_{\rho : \overline{S} \to \{0,1\}} \min(r^0_\rho, r^1_\rho) < \epsilon \,.$$

Then f is ϵ-close to a k-junta, because we can define the k-junta h as follows. For each x we can set $h(x) = 0$ if $r^0_\rho \geq r^1_\rho$ and $h(x) = 1$ if $r^0_\rho < r^1_\rho$, where ρ is the partial assignment defining the row R_ρ containing x. Since h is constant on each row, it does not depend on any of the variables that are not assigned by $\rho : \overline{S} \to \{0,1\}$, i.e. it does not depend on any of the $n-k$ variables in S. And by definition,

$$\mathsf{dist}_p(f,h) = \sum_{\rho : \overline{S} \to \{0,1\}} \min(r^0_\rho, r^1_\rho) < \epsilon \,.$$

If f is ϵ-far, then every set S of $n-k$ variables satisfies

$$\sum_{\rho : \overline{S} \to \{0,1\}} \min(r^0_\rho, r^1_\rho) \geq \epsilon \,.$$

For any fixed S, we can bound the probability that the set Q does not contain any witness as follows. Without loss of generality assume that $r^0_\rho \leq r^1_\rho$ for every ρ, and choose a set $T_\rho \subseteq R_\rho$ such that

$$r^0_\rho = \sum_{x \in T_\rho : f(x)=0} p(x) = \sum_{x \in T_\rho : f(x)=1} p(x) \,,$$

which we may do since, without loss of generality, we may adjust the probabilities $p(x)$ in each row without changing the probability of finding a witness, as long as the totals r^0_ρ, r^1_ρ are invariant. Note that if two random points $x, y \sim p$ fall in T_ρ, then with probability $1/2$ we will have $f(x) \neq f(y)$. Therefore

$$\mathop{\mathbb{P}}_{Q}[\exists x, y, \rho : f(x) \neq f(y), x, y \in R_\rho] \geq \frac{1}{2} \cdot \mathop{\mathbb{P}}_{Q}[\exists \rho : T_\rho \text{ contains} \geq 2 \text{ points}] \,.$$

Let $T = \cup_{\rho : \overline{S} \to \{0,1\}} T_\rho$ and observe that $\sum_{x \in T} p(x) = \sum_\rho r^0_\rho \geq \epsilon$, so in expectation there are ϵm points in $Q \cap T$. By the Chernoff bound,

$$\mathbb{P}\left[|Q \cap T| < \frac{\epsilon m}{2}\right] \leq \exp\left(-\frac{\epsilon m}{8}\right) = o(1) \,.$$

Assume there are at least $\epsilon m/2$ points in T. By Proposition 2.7.6, the probability that no T_ρ contains at least 2 points is, for $N = 2^k$ being the number of rows, at most

$$\exp\left(-\frac{\left(\frac{\epsilon m}{2}-1\right)^2}{2N}\right) < \frac{1}{2},$$

since $m = \Omega\left(\frac{\sqrt{N}}{\epsilon}\right)$. Therefore the probability of finding a witness for S is at least

$$\frac{1}{2}\cdot\mathop{\mathbb{P}}_{Q}\left[\exists\rho : T_\rho \text{ contains} \geq 2 \text{ points}\right] \geq \frac{1}{2}\cdot(1-o(1))\cdot\frac{1}{2} = (1-o(1))\frac{1}{4} > \frac{1}{5}.$$

By repeating the sampling procedure $s = O\left(\log\binom{n}{k}\right)$ times, the probability of failing to find a witness is at most $(4/5)^s < \frac{1}{3}\binom{n}{k}^{-1}$. Then by the union bound, the probability that there exists S on which the tester fails to find a witness is at most $1/3$, since there are at most $\binom{n}{k}$ such sets. □

2.7.3 Monotonicity in General Posets

A basic result in testing monotonicity of Boolean functions over the uniform distribution is that at most $O(\sqrt{n/\epsilon})$ uniform samples are necessary for any partial order of size n [FLN+02]. We extend this result to the distribution-free setting. The VC dimension of the class of monotone functions over any poset P is the width, i.e. the size of the largest antichain in P. For example, the standard partial ordering of the hypercube $\{0,1\}^n$ has width $\Theta(2^n/\sqrt{n})$ since the set of points $x \in \{0,1\}^n$ with Hamming weight $n/2$ is an antichain of size $\binom{n}{n/2}$. Therefore, for the hypercube, distribution-free sample-based testing can be done with sample complexity $O(2^{n/2}) = \widetilde{O}(\sqrt{\mathsf{VC}})$.

For sets X, Y and a set of order pairs $E \subseteq X \times Y$, we call the triple (X, Y, E) a *bipartite* partial order, where the edges E define the following partial order on $X \cup Y$: $x < y$ iff $(x, y) \in E$. Fischer *et al.* [FLN+02] observed that for the uniform distribution, monotonicity on general finite posets reduces to testing on bipartite posets; we generalize their reduction to the distribution-free setting:

Lemma 2.7.8. *If for every bipartite partial order (X, Y, E) of size $|X| = |Y| = n$ there is a distribution-free sample-based ϵ-tester for monotonicity with sample complexity $m(n, \epsilon)$ then for every partial order P of size $|P| = n$ there is a distribution-free sample-based ϵ-tester for monotonicity with sample complexity $m(n, \epsilon/2)$.*

Proof. On any partial order P with distribution p and input function $f : P \to \{0,1\}$, consider the following reduction: let X, Y be separate copies of P and for each $x \in P$ write x_1, x_2 for the copies of x in X, Y respectively. Define a set of edges $E \subset X \times Y$ where $(x_1, y_2) \in E$ iff $x < y$ in P. Define the distribution q over $X \cup Y$ as $q(x_1) = \frac{1}{2}p(x)$ for each $x \in X$ and $q(y_2) = \frac{1}{2}p(y_2)$ for each $y \in Y$. Define the function $g : X \cup Y \to \{0,1\}$ as $g(x_1) = f(x), g(y_2) = f(y)$ for each $x_1 \in X, y_2 \in Y$. Observe that we can simulate a random sample from q labelled by g by sampling $x \sim p$ and taking x_1, x_2 with equal probability, labelling it with $f(x)$.

It is clear that if f is monotone on P then g is monotone on (X, Y, E), since $(x_1, y_2) \in E$ implies $x < y$ in P. Suppose now that g is ϵ-close to monotone in (X, Y, E) according to q, and let h be monotone on (X, Y, E) minimizing distance to g. Define $f' : P \to \{0, 1, *\}$ as follows: For $x \in P$, if $h(x_1) = h(x_2) = f(x)$ set $f'(x) = f(x)$, and otherwise set $f'(x) = *$. Then

$$\begin{aligned}
\sum_{x \in P : f'(x) = *} p(x) &= \sum_{x \in P} \mathbb{1}\left[h(x_1) \neq f(x) \vee h(x_2) \neq f(x)\right] p(x) \\
&\leq \sum_{x \in P} \left(\mathbb{1}\left[h(x_1) \neq f(x)\right] + \mathbb{1}\left[h(x_2) \neq f(x)\right]\right) p(x) \\
&= \sum_{x \in P} \mathbb{1}\left[h(x_1) \neq g(x_1)\right] p(x) + \sum_{x \in P} \mathbb{1}\left[h(x_2) \neq g(x_2)\right] p(x) \\
&= 2 \sum_{x \in P} \mathbb{1}\left[h(x_1) \neq g(x_1)\right] q(x_1) + 2 \sum_{x \in P} \mathbb{1}\left[h(x_2) \neq g(x_2)\right] q(x_2) \\
&= 2\mathsf{dist}_q(g, h) < 2\epsilon\,.
\end{aligned}$$

Now construct a monotone function $f'' : P \to \{0,1\}$ as follows. Take any total order $\prec$ consistent with the partial order on P. For each $x \in P$ in order of $\prec$, if $f'(x) = *$ set $f''(x) = \max_{y < x} f''(y)$, otherwise set $f''(x) = f'(x) = f(x)$. Then $\mathsf{dist}_p(f, f'') \leq \sum_{x \in P : f'(x) = *} p(x) < 2\epsilon$. Suppose that f'' is not monotone, so there are $x < y$ such that $f''(x) = 1, f''(y) = 0$; assume x is a minimal point where this occurs. Since $x \prec y$ it must be the case that $f'(y) \neq *$, so $f'(y) = f(y) = 0$. Since f' is monotone except on $*$-valued points, it must be that $f'(x) = *$, and $1 = f''(x) = \max_{z < x} f''(z)$. But then $z < x$ and $f''(z) = 1, f''(y) = 0$, so x was not minimal, a contradiction. Thus f'' is monotone and $\mathsf{dist}(f, f'') < 2\epsilon$.

We therefore conclude that if f is ϵ-far from monotone on P according to p then g is at least $(\epsilon/2)$-far from monotone on (X, Y, E) according to q. Therefore, by simulating the distribution-free one-sided sample-based tester on (X, Y, E) with parameter $\epsilon/2$ we obtain a distribution-free one-sided tester for P. □

Theorem 2.7.9. *For any finite partial order P of size $|P| = n$, there is a distribution-free, one-sided, sample-based ϵ-tester for monotonicity with sample complexity $O\left(\frac{\sqrt{n}}{\epsilon}\right)$.*

Proof. By Lemma 2.7.8, it suffices to consider bipartite partial orders. Let (X, Y, E) be a bipartite partial order. On input $f : X \cup Y \to \{0, 1\}$ and distribution $p : X \cup Y \to \mathbb{R}$, the tester will sample a set Q of $m = O\left(\frac{\sqrt{n}}{\epsilon}\right)$ points from p and reject if there exist $x \in X \cap Q, y \in Y \cap Q$ such that $x < y$ and $f(x) = 1, f(y) = 0$; we call such a pair a *violating pair*.

Suppose f is ϵ-far from monotone. Let $X_1 := \{x \in X : f(x) = 1\}$ and $Y_0 := \{y \in Y : f(y) = 0\}$. For each $x \in X_1$ let $V_x := \{y \in Y_0 : x < y\}$ be the set of points $y \in Y_0$ such that (x, y) is a violating pair, and define $q(x) := \sum_{y \in V_x} p(y)$ be the total probability mass of all the points y such that (x, y) is a violating pair. Suppose for contradiction that

$$\sum_{x \in X_1} \min(p(x), q(x)) < \epsilon \,.$$

Construct a monotone function h as follows. For each $x \in X_1$ (in arbitrary order), if $p(x) < q(x)$ set $h(x) = 0$, otherwise set $h(y) = 1$ for all $y \in V_x$. The resulting function is now monotone since each violating pair (x, y) now has either $h(x) = 0$ or $h(y) = 1$. The distance between f and h increases by at most $\min(p(x), q(x))$ for each $x \in X_1$, so $\mathsf{dist}_p(f, h) < \epsilon$, a contradiction. Therefore we must have $\sum_{x \in X_1} \min(p(x), q(x)) \geq \epsilon$.

Define a new distribution r on $X \cup Y$ that is initialized to $r = p$ but then is updated to set $r(x) = \min(p(x), q(x))$ for each $x \in X_1$, reassigning the remaining probability mass $p(x) - r(x)$ to an arbitrary point not in $X_1 \cup Y_0$ (we may assume such a point exists since otherwise f is the trivial function where every pair $x < y$ is a violating pair). This reassignment can only decrease the probability of finding a violating pair in the sample Q. Under the new distribution $r(X_1) = \sum_{x \in X_1} r(x) = \sum_{x \in X_1} \min(p(x), q(x)) \geq \epsilon$, and for each $x \in X_1, r(x) \leq r(V_x)$. Let R be a set of m independent points drawn from r, so $\mathbb{P}[Q \text{ contains a violating pair}] \geq \mathbb{P}[R \text{ contains a violating pair}]$.

Now observe that for each $x \in X_1$, the probability that R contains some violating pair (x, y) is at least the probability that x occurs twice in R; this is because $r(V_x) = q(x) \geq r(x)$. We will now bound the probability that there exists $x \in X_1$ that occurs twice in R. The expected number of points in $R \cap X_1$ is $\mathbb{E}[|R \cap X_1|] = m \sum_{x \in X_1} r(x) \geq \epsilon m$. By the Chernoff bound,

$$\mathbb{P}\left[|R \cap X_1| < \frac{\epsilon m}{2}\right] \leq \exp\left(-\frac{\epsilon m}{8}\right) = o(1) \,.$$

Assuming $|R \cap X_1| \geq \frac{\epsilon m}{2}$, by Proposition 2.7.6, the probability that each $x \in R \cap X_1$ occurs at most once in R is at most

$$\exp\left(-\frac{(|R \cap X_1| - 1)^2}{2|X_1|}\right) \leq \exp\left(-\frac{((\epsilon m/2) - 1)^2}{2n}\right) < 1/4\,,$$

for sufficiently large $m = O\left(\frac{\sqrt{n}}{\epsilon}\right)$. Therefore

$$\mathbb{P}\left[\exists x \in X_1, x \text{ occurs at least twice in } R\right] \geq 1 - \frac{1}{4} - o(1) \geq \frac{2}{3}\,.$$

Finally,

$$\begin{aligned}\mathbb{P}\left[Q \text{ contains a violating pair}\right] &\geq \mathbb{P}\left[R \text{ contains a violating pair}\right] \\ &\geq \mathbb{P}\left[\exists x \in X_1, x \text{ occurs at least twice in } R\right] \geq \frac{2}{3}\,. \qquad \square\end{aligned}$$

Chapter 3

Testing and Learning under Product Distributions

After decades of school, I can attest,
After uncountable books consumed without rest,
After exercise problems at the teacher's request,
All I can do is to learn and to test.

In this chapter, we study the second direction of research in testing and learning: designing efficient algorithms that work under the weakest possible restrictions on the class of input distributions $\mathfrak{D}$. The previous chapter made progress towards a theory of testing vs. learning when $\mathfrak{D}$ is essentially unrestricted and the complexity of the algorithm is measured by the number of requests to the labeled-sample oracle. In the present chapter we are not explicitly concerned with theorizing about testing vs. learning. Instead we focus on optimizing the cost of the algorithms in terms of the number of requests to the query, sample, or labeled-sample oracles, as well as (for learning algorithms) their time complexity; and on eliminating restrictions on the class of input distributions $\mathfrak{D}$.

Compared to the previous chapter, two important differences for learning algorithms in this chapter are:

- When $\mathfrak{D}$ is restricted, the VC dimension no longer determines the labeled-sample complexity of a learning algorithm. Indeed, there are classes, like convex sets, which can be learned under product distributions (as we will show), even though they have infinite VC dimension and are therefore not learnable when $\mathfrak{D}$ is unrestricted.

- We are now concerned with time complexity, not only the number of requests to labeled-sample oracle; the importance of this was emphasized in Section 1.2.4.

Recall from Section 1.2.4 that the goal for this chapter is to design testing and learning algorithms for the case where $\mathfrak{D}$ is any *product distribution* over $\mathbb{R}^d$. This is the natural generalization of many of the special cases for which algorithms have been developed in the prior literature, such as the uniform distribution over $\{\pm 1\}^d$ or the standard Gaussian over $\mathbb{R}^d$.

This chapter provides a general framework for designing distribution-free testing and learning algorithms under product distributions on $\mathbb{R}^d$, which may be finite or continuous. This framework, which we call *downsampling*, improves upon previous methods (in particular, [BCS20, BOW10]), in a few ways. It is more general and does not apply only to a specific type of algorithm [BOW10] or a specific problem [BCS20], and we use it to obtain many other results. It is conceptually simpler. And it allows quantitative improvements over both [BOW10] and [BCS20].

See Table 1.2 in Chapter 1 for a summary of our results in property testing, and Table 1.3 for a summary of our results in learning. In Section 3.1, we give an informal description of our techniques, and we introduce the main definitions and lemmas in Section 3.2. The following sections present the proofs of the results. For simplicity, we first handle only continuous product distributions, and handle finite distributions separately in Section 3.8.

3.1 Techniques

What connects these diverse problems is a notion of rectilinear surface area or isoperimetry that we call "block boundary size". There is a close connection between learning & testing and various notions of isoperimetry or surface area (e.g. [CS16, KOS04, KOS08, KMS18]). We show that testing or learning a class $\mathcal{H}$ on product distributions over $\mathbb{R}^d$ can be reduced to testing and learning on the *uniform* distribution over $[r]^d$, where r is determined by the block boundary size, and we call this reduction "downsampling". The name *downsampling* is used in image and signal processing for the process of reducing the resolution of an image or reducing the number of discrete samples used to represent an analog signal. We adopt the name because our method can be described by analogy to image or signal processing as the following 2-step process:

1. Construct a "digitized" or "pixellated" image of the function $f : \mathbb{R}^d \to \{\pm 1\}$ by sampling from the distribution and constructing a grid in which each cell has roughly equal probability mass; and

2. Learn or test the "low-resolution" pixellated function.

As long as the function f takes a constant value in the vast majority of "pixels", the low resolution version seen by the algorithm is a good enough approximation for testing or learning. The block boundary size is, informally, the number of pixels on which f is not constant.

This technique reduces distribution-free testing and learning problems to the uniform distribution in a way that is conceptually simpler than in the prior work [BOW10, BCS20]. However, some technical challenges remain. The first is that it is not always easy to bound the number of "pixels" on which a function f is not constant – for example, for PTFs. Second, unlike in the uniform distribution, the resulting downsampled function class on $[r]^d$ is not necessarily "the same" as the original class – for example, halfspaces on $\mathbb{R}^d$ are not downsampled to halfspaces on $[r]^d$, since the "pixels" are not of equal size. Thus, geometric arguments may not work, unlike the case for actual images.

A similar technique of constructing "low-resolution" representations of the input has been used and rediscovered ad-hoc a few times in the property testing literature; in prior work, it was restricted to the uniform distribution over $[n]^d$ [KR00, Ras03, FR10, BY19, CGG+19] (or the Gaussian in [CFSS17]). In this thesis, we aim to provide a unified and generalized study of this simple and powerful technique.

3.1.1 Block Boundary Size

Informally, we define the *r-block boundary size* $\mathsf{bbs}(\mathcal{H}, r)$ of a class $\mathcal{H}$ of functions $\mathbb{R}^d \to \{0, 1\}$ as the maximum number of grid cells on which a function $f \in \mathcal{H}$ is non-constant, over all possible $r \times \cdots \times r$ grid partitions of $\mathbb{R}^d$ (which are not necessarily evenly spaced) – see Section 3.2 for formal definitions. Whether downsampling can be applied to $\mathcal{H}$ depends on whether

$$\lim_{r \to \infty} \frac{\mathsf{bbs}(\mathcal{H}, r)}{r^d} \to 0\,,$$

and the complexity of the algorithms depends on how large r must be for the non-constant blocks to vanish relative to the whole r^d grid. A general observation is that any function class $\mathcal{H}$ where downsampling can be applied can be learned under unknown product distributions with a finite number of samples; for example, this holds for convex sets even though the VC dimension is infinite.

Proposition 3.1.1 (Consequence of Lemma 3.4.8)**.** *Let $\mathcal{H}$ be any set of functions $\mathbb{R}^d \to \{0,1\}$ (measurable with respect to continuous product distributions) such that*

$$\lim_{r\to\infty} \frac{\mathsf{bbs}(\mathcal{H}, r)}{r^d} = 0\,.$$

Then there is some function $\delta(d,\epsilon)$ such that $\mathcal{H}$ is distribution-free learnable under product distributions, up to error ϵ, with $\delta(d,\epsilon)$ samples.

For convex sets, monotone functions, k-alternating functions, and halfspaces, $\mathsf{bbs}(\mathcal{H}, r)$ is easy to calculate. For degree-k PTFs, it is more challenging. We say that a function $f : \mathbb{R}^d \to \{0,1\}$ induces a connected component S if for every $x, y \in S$ there is a continuous curve in $\mathbb{R}^d$ from x to y such that $f(z) = f(x) = f(y)$ for all z on the curve, and S is a maximal such set. Then we prove a general lemma that bounds the block boundary size by the number of connected components induced by functions $f \in \mathcal{H}$.

Lemma 3.1.2 (Informal, see Lemma 3.6.6)**.** *Suppose that for any axis-aligned affine subspace A of affine dimension $n \le d$, and any function $f \in \mathcal{H}$, f induces at most k^n connected components in A. Then for $r = \Omega(dk/\epsilon)$, $\mathsf{bbs}(\mathcal{H}, r) \le \epsilon \cdot r^d$.*

This lemma in fact generalizes all computations of block boundary size in this thesis (up to constant factors in r). Using a theorem of Warren [War68], we get that any degree-k polynomial $\mathbb{R}^d \to \{\pm 1\}$ achieves value 0 in at most ϵr^d grid cells, for sufficiently large $r = \Omega(dk/\epsilon)$ (Corollary 3.6.8).

3.1.2 Polynomial Regression

The second step of downsampling is to find a testing or learning algorithm that works for the uniform distribution over the (not necessarily evenly-spaced) hypergrid. Most of our learning results use *polynomial regression*. This is a powerful technique introduced in [KKMS08] that performs linear regression over a vector space of functions that approximately spans the hypothesis class. This method is usually applied by using Fourier analysis to construct such an approximate basis for the hypothesis class [BOW10, DHK+10, CGG+19]. This was the method used, for example, by Blais, O'Donnell, & Wimmer [BOW10] to achieve the poly(dn)-time algorithms for intersections of halfspaces.

We take the same approach but we use the Walsh basis for functions on domain $[n]^d$ (see e.g. [BRY14]) instead of the bases used in the prior works. We show that if one can establish bounds on the noise sensitivity in the Fourier basis for the hypothesis class

restricted to the uniform distribution over $\{\pm 1\}^d$, then one gets a bound on the number of Walsh functions required to approximately span the "downsampled" hypothesis class. In this way, we establish that if one can apply standard Fourier-analytic techniques to the hypothesis class over the *uniform* distribution on $\{\pm 1\}^d$ and calculate the block boundary size, then the results for the hypercube essentially carry over to the distribution-free setting for product distributions on $\mathbb{R}^d$.

An advantage of this technique is that both noise sensitivity and block boundary size grow at most linearly during function composition: for functions $f(x) = g(h_1(x), \ldots, h_k(x))$ where each h_i belongs to the class $\mathcal{H}$, the noise sensitivity and block boundary size grow at most linearly in k. Therefore learning results for $\mathcal{H}$ obtained in this way are easy to extend to arbitrary compositions of $\mathcal{H}$, which is how we get our result for intersections of halfspaces.

3.2 Downsampling

We will now introduce the main definitions, notation, and lemmas required by our main results. The purpose of this section is to establish the main conceptual component of the downsampling technique: that functions with small enough block boundary size can be efficiently well-approximated by a "coarsened" version of the function that is obtained by random sampling. See Figure 3.1 for an illustration of the following definitions.

Definition 3.2.1 (Block Partitions)**.** An *r-block partition* of $\mathbb{R}^d$ is a pair of functions $\mathsf{block} : \mathbb{R}^d \to [r]^d$ and $\mathsf{blockpoint} : [r]^d \to \mathbb{R}^d$ obtained as follows. For each $i \in [d], j \in [r-1]$ let $a_{i,j} \in \mathbb{R}$ such that $a_{i,j} < a_{i,j+1}$ and define $a_{i,0} := -\infty$, $a_{i,r} := \infty$ for each i. For each $i \in [d], j \in [r]$ define the interval $B_{i,j} := (a_{i,j-1}, a_{i,j}]$ and pick a point $b_{i,j} \in B_{i,j}$. The function $\mathsf{block} : \mathbb{R}^d \to [r]^d$ is defined by setting $\mathsf{block}(x)$ to be the unique vector $v \in [r]^d$ such that $x_i \in B_{i,v_i}$ for each $i \in [d]$. The function $\mathsf{blockpoint} : [r]^d \to \mathbb{R}^d$ is defined by setting $\mathsf{blockpoint}(v) = (b_{1,v_1}, \ldots, b_{d,v_d})$; note that $\mathsf{blockpoint}(v) \in \mathsf{block}^{-1}(v)$ where $\mathsf{block}^{-1}(v) := \{x \in \mathbb{R}^d : \mathsf{block}(x) = v\}$.

Definition 3.2.2 (Block Functions and Coarse Functions)**.** For a function $f : \mathbb{R}^d \to \{\pm 1\}$, we define $f^{\mathsf{block}} : [r]^d \to \{\pm 1\}$ as $f^{\mathsf{block}} := f \circ \mathsf{blockpoint}$ and $f^{\mathsf{coarse}} : \mathbb{R}^d \to \{\pm 1\}$ as $f^{\mathsf{coarse}} := f^{\mathsf{block}} \circ \mathsf{block} = f \circ \mathsf{blockpoint} \circ \mathsf{block}$. For any set $\mathcal{H}$ of functions $\mathbb{R}^d \to \{\pm 1\}$, we define $\mathcal{H}^{\mathsf{block}} := \{f^{\mathsf{block}} \mid f \in \mathcal{H}\}$. For a distribution μ over $\mathbb{R}^d$ and an r-block partition $\mathsf{block} : \mathbb{R}^d \to [r]^d$ we define the distribution $\mathsf{block}(\mu)$ over $[r]^d$ as the distribution of $\mathsf{block}(x)$ for $x \sim \mu$.

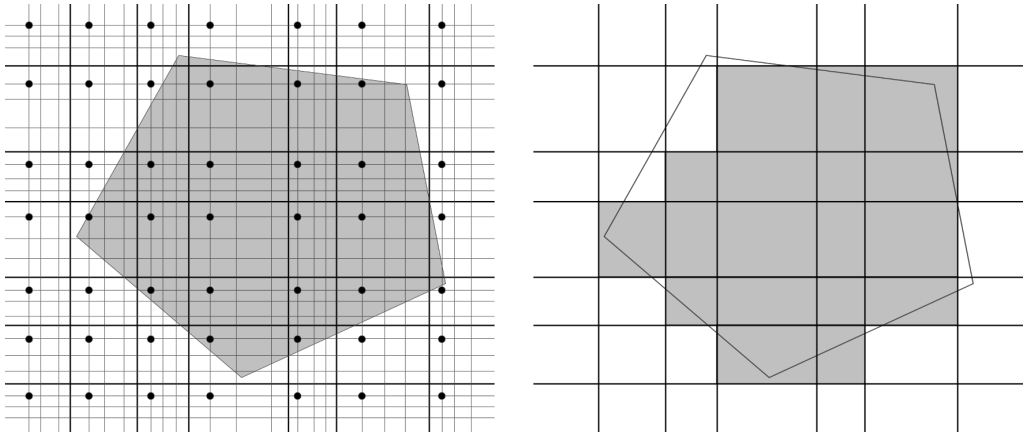


Figure 3.1: Left: Random grid X (pale lines) with induced block partition (thick lines) and $\mathsf{blockpoint}$ values (dots), superimposed on $f^{-1}(1)$ (gray polygon). Right: f^{coarse} (grey) compared to f (polygon outline).

Definition 3.2.3 (Induced Block Partitions)**.** When μ is a product distribution over $\mathbb{R}^d$, a *random grid* X of length m is the grid obtained by sampling m points $x_1, \ldots, x_m \in \mathbb{R}^d$ independently from μ and for each $i \in [d], j \in [m]$ defining $X_{i,j}$ to be the j^{th}-smallest coordinate in dimension i among all sampled points. For any r that divides m we define an r-block partition depending on X by defining for each $i \in [d], j \in [r-1]$ the point $a_{i,j} := X_{i,mj/r}$ so that the intervals are $B_{i,j} := (X_{i,m(j-1)/r}, X_{i,mj/r}]$ when $j \in \{2, \ldots, r-1\}$ and $B_{i,1} = (-\infty, X_{i,m/r}], B_{i,r} = (X_{i,m(r-1)/r}, \infty)$; we let the points $b_{i,j}$ defining $\mathsf{blockpoint}$ be arbitrary. This is the r-block partition *induced* by X.

Definition 3.2.4 (Block Boundary Size)**.** For a block partition $\mathsf{block} : \mathbb{R}^d \to [r]^d$, a distribution μ over $\mathbb{R}^d$, and a function $f : \mathbb{R}^d \to \{\pm 1\}$, we say f is *non-constant* on a block $v \in [r]^d$ if there are sets $S, T \subset \mathsf{block}^{-1}(v)$ such that $\forall s \in S, t \in T : f(s) = 1, f(t) = -1$; and S, T have positive measure (in the product of Lebesgue measures). For a function $f : \mathbb{R}^d \to \{\pm 1\}$ and a number r, we define the r-block boundary size $\mathsf{bbs}(f, r)$ as the maximum number of blocks on which f is non-constant, where the maximum is taken over all r-block partitions $\mathsf{block} : \mathbb{R}^d \to [r]^d$. For a set $\mathcal{H}$ of functions $\mathbb{R}^d \to \{\pm 1\}$, we define $\mathsf{bbs}(\mathcal{H}, r) := \max\{\mathsf{bbs}(f, r) \mid f \in \mathcal{H}\}$.

The *total variation distance* between two distributions μ, ν over a finite domain $\mathcal{X}$ is defined as

$$\|\mu - \nu\|_{\mathsf{TV}} := \frac{1}{2} \sum_{x \in \mathcal{X}} |\mu(x) - \nu(x)| = \max_{S \subseteq \mathcal{X}} |\mu(S) - \nu(S)| .$$

The essence of downsampling is apparent in the next proposition. It shows that the distance of f to its coarsened version f^{coarse} is bounded by two quantities: the fraction of blocks in the r-block partition on which f is not constant, and the distance of the distribution $\mathsf{block}(\mu)$ to uniform. When both quantities are small, testing or learning f can be done by testing or learning f^{coarse} instead. The uniform distribution over a set S is denoted $\mathsf{unif}(S)$:

Proposition 3.2.5. *Let μ be a continuous product distribution over $\mathbb{R}^d$, let X be a random grid, and let $\mathsf{block} : \mathbb{R}^d \to [r]^d$ be the induced r-block partition. Then, for any measurable $f : \mathbb{R}^d \to \{\pm 1\}$, the following holds with probability 1 over the choice of X:*

$$\mathop{\mathbb{P}}_{x\sim\mu}[f(x) \neq f^{\mathsf{coarse}}(x)] \leq r^{-d} \cdot \mathsf{bbs}(f, r) + \|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}}\,.$$

Proof. We first establish that, with probability 1 over X and $x \sim \mu$, if $f(x) \neq f^{\mathsf{coarse}}(x)$ then f is non-constant on $\mathsf{block}(x)$. Fix X and suppose there exists a set Z of positive measure such that for each $x \in Z$, $f(x) \neq f^{\mathsf{coarse}}(x)$ but f is not non-constant on $\mathsf{block}(x)$, i.e. for $V = \mathsf{block}^{-1}(\mathsf{block}(x))$, either $\mu(V \cap f^{-1}(1)) = \mu(V)$ or $\mu(V \cap f^{-1}(-1)) = \mu(V)$. Then there is $v \in [r]^d$ such that for $V = \mathsf{block}^{-1}(v)$, $\mu(Z \cap V) > 0$. Let $y = \mathsf{blockpoint}(v)$. If $\mu(V \cap f^{-1}(f(y)) = \mu(V)$ then $\mu(Z \cap V) = 0$, so $\mu(V \cap f^{-1}(f(y)) = 0$. But for random X, the probability that there exists $v \in [r]^d$ such that $\mu(V \cap f^{-1}(\mathsf{blockpoint}(v))) = 0$ is 0, since $\mathsf{blockpoint}(v)$ is random within V.

Assuming that the above event occurs,

$$\begin{aligned}\mathop{\mathbb{P}}_{x\sim\mu}[f(x) \neq f^{\mathsf{coarse}}(x)] &\leq \mathop{\mathbb{P}}_{x\sim\mu}[f \text{ is non-constant on } \mathsf{block}(x)] \\ &\leq \mathop{\mathbb{P}}_{v\sim[r]^d}[f \text{ is non-constant on } v] + \|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}}\,.\end{aligned}$$

Since $v \sim [r]^d$ is uniform, the probability of hitting a non-constant block is at most $r^{-d} \cdot \mathsf{bbs}(f, r)$. □

Next we give a bound on the number of samples required to ensure that $\mathsf{block}(\mu)$ is close to uniform. We need the following lemma.

Lemma 3.2.6. *Let μ be continuous probability distribution over $\mathbb{R}$, $m, r \in \mathbb{N}$ such that r divides m, and $\delta \in (0, 1/2)$. Let X be a set of m points sampled independently from μ. Write $X = \{x_1, \ldots, x_m\}$ labeled such that $x_1 < \cdots < x_m$ (and write $x_0 = -\infty$). Then for any $i \in [r]$,*

$$\mathbb{P}\left[\mu\left(x_{(i-1)(m/r)}, x_{i(m/r)}\right] < \frac{1-\delta}{r}\right] \leq 4 \cdot e^{-\frac{\delta^2 m}{32r}}\,.$$

Proof. We assume that $i-1 \le r/2$. If $i-1 > r/2$ then we can repeat the following analysis with the opposite ordering on the points in X. Write $x^* = x_{(i-1)\frac{m}{r}}$ and $\beta = \mu(-\infty, x^*]$. First suppose that $(1-\delta/2)\frac{i-1}{r} < \beta < (1+\delta/2)\frac{i-1}{r} \le (1+\delta/2)/2$; we will bound the probability of this event later.

Let $t \in \mathbb{R}$ be the point such that $\mu(x^*, t] = (1-\delta)/r$ (which must exist since μ is continuous). Let $\eta = \frac{\delta}{1-\delta} \ge \delta$. Write $X^* = \{x \in X : x > x^*\}$. The expected value of $|X^* \cap (x^*, t]|$ is $|X^*|\frac{1-\delta}{r(1-\beta)} = \left(1 - \frac{i-1}{r}\right)\frac{1-\delta}{r(1-\beta)}$, where the factor $1-\beta$ in the denominator is due to the fact that each element of X^* is sampled from μ conditional on being larger than x^*. The event $\mu(x^*, x_{i(m/r)}] < (1-\delta)/r$ occurs if and only if $|X^* \cap (x^*, t]| > m/r$, which occurs with probability

$$\mathbb{P}\left[|X^* \cap (x^*, t]| > \frac{m}{r}\right] = \mathbb{P}\left[|X^* \cap (x^*, t]| > m\left(1 - \frac{(i-1)}{r}\right)\frac{1-\delta}{r(1-\beta)}(1+\eta)\right]$$

where

$$\begin{aligned} 1+\eta = \frac{(1-\beta)}{(1-\delta)\left(1-\frac{i-1}{r}\right)} &\ge \frac{\left(1-(1+\delta/2)\frac{i-1}{r}\right)}{(1-\delta)\left(1-\frac{i-1}{r}\right)} = \frac{1}{1-\delta}\left(1 - \frac{(\delta/2)(i-1)}{r-(i-1)}\right) \\ &\ge \frac{1-\delta/2}{1-\delta} = 1 + \frac{\delta}{2(1-\delta)} \ge 1 + \delta/2\,. \end{aligned}$$

Since the expected value satisfies

$$|X^*|\frac{1-\delta}{r(1-\beta)} \ge \frac{m}{r}(1 - \frac{i-1}{r})\frac{2(1-\delta)}{1-\delta/2} \ge \frac{m}{r}(1-\delta/2) \ge \frac{m}{2r}\,,$$

the Chernoff bound gives

$$\mathbb{P}\left[|X^* \cap (x^*, t]| > \frac{m}{r}\right] \le \exp\left(-\frac{\delta^2|X^*|(1-\delta)}{3 \cdot 4 \cdot r(1-\beta)}\right) \le e^{-\frac{\delta^2 m}{3\cdot 4\cdot 2r}}\,.$$

Now let $t \in \mathbb{R}$ be the point such that $\mu(x^*, t] = (1+\delta)/r$. The expected value of $|X^* \cap (x^*, t]|$ is now $|X^*|\frac{1+\delta}{r(1-\beta)}$. The event $\mu(x^*, x_{i(m/r)}] > (1+\delta)/r$ occurs if and only if $|X^* \cap (x^*, t]| < m/r$, which occurs with probability

$$\mathbb{P}\left[|X^* \cap (x^*, t]| < \frac{m}{r}\right] = \mathbb{P}\left[|X^* \cap (x^*, t]| < m\left(1 - \frac{i-1}{r}\right)\frac{1+\delta}{r(1-\beta)}(1-\eta)\right]$$

where

$$\begin{aligned} 1-\eta = \frac{1-\beta}{(1+\delta)(1-\frac{i-1}{r})} &\le \frac{1-(1+\delta/2)\frac{i-1}{r}}{(1+\delta)\left(1-\frac{i-1}{r}\right)} = \frac{1}{1+\delta}\left(1 + \frac{(\delta/2)(i-1)}{r-(i-1)}\right) \\ &\le \frac{1+\delta/2}{1+\delta} = 1 - \frac{\delta/2}{1+\delta} \le 1 - \frac{\delta}{4}\,. \end{aligned}$$

The expected value satisfies $|X^*|\frac{1+\delta}{r(1-\beta)} > m/r$, so the Chernoff bound gives

$$\mathbb{P}\left[|X^* \cap (x^*, t]| < \frac{m}{r}\right] \leq \exp\left(-\frac{\delta^2 |X^*|(1+\delta)}{2 \cdot 4^2 \cdot r(1-\beta)}\right) \leq e^{-\frac{\delta^2 m}{2 \cdot 4^2}}.$$

It remains to bound the probability that $(1-\delta/2)\frac{i-1}{r} < \beta < (1+\delta/2)\frac{i-1}{r}$. Define $t \in \mathbb{R}$ such that $\mu(-\infty, t] = (1+\delta/2)\frac{i-1}{r}$. $\beta = \mu(-\infty, x^*] \geq (1+\delta/2)\frac{i-1}{r}$ if and only if $x^* > t$, i.e. $|X \cap (-\infty, t]| < \frac{i-1}{r}$. The expected value of $|X \cap (-\infty, t]|$ is $m\frac{(1+\delta/2)(i-1)}{r}$, so for $\eta = \frac{\delta/2}{1+\delta/2} \geq \delta/3$, the Chernoff bound implies

$$\begin{aligned}\mathbb{P}\left[|X \cap (-\infty, t]| < m\frac{i-1}{r}\right] &= \mathbb{P}\left[|X \cap (-\infty, t]| < m\frac{(1+\delta/2)(i-1)}{r}(1-\eta)\right] \\ &\leq e^{-\frac{\delta^2 m(1+\delta/2)(i-1)}{18r}} \leq e^{-\frac{\delta^2 m}{18r}}.\end{aligned}$$

Now define $t \in \mathbb{R}$ such that $\mu(-\infty, t] = (1-\delta/2)\frac{i-1}{r}$. $\beta = \mu(-\infty, x^*] \leq (1-\delta/2)\frac{i-1}{r}$ if and only if $x^* < t$, i.e. $|X \cap (-\infty, t]| > \frac{i-1}{r}$. The expected value of $|X \cap (-\infty, t]|$ is $m\frac{(1-\delta/2)(i-1)}{r}$, so for $\eta = \frac{\delta}{2-\delta} \geq \delta/2$,

$$\begin{aligned}\mathbb{P}\left[|X \cap (-\infty, t]| > m\frac{i-1}{r}\right] &= \mathbb{P}\left[|X \cap (-\infty, t]| > m\frac{(1-\delta/2)(i-1)}{r}(1+\eta)\right] \\ &\leq e^{-\frac{\delta^2 m(1-\delta/2)(i-1)}{2 \cdot 4r}} \leq e^{-\frac{\delta^2 m}{4^2 r}}.\end{aligned}$$

The conclusion then follows from the union bound over these four events. □

Lemma 3.2.7. *Let $\mu = \mu_1 \times \cdots \times \mu_d$ be a product distribution over $\mathbb{R}^d$ where each μ_i is continuous. Let X be a random grid with length m sampled from μ, and let $\mathsf{block} : \mathbb{R}^d \to [r]^d$ be the r-block partition induced by X. Then*

$$\mathop{\mathbb{P}}_{X}\left[\|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} > \epsilon\right] \leq 4rd \cdot e^{-\frac{\epsilon^2 m}{18rd^2}}$$

Proof. For a fixed grid X and each $i \in [d]$, write $p_i : [r] \to [0,1]$ be the probability distribution on $[r]$ with $p_i(z) = \mu_i(B_{i,z})$. Then $\mathsf{block}(\mu) = p_1 \times \cdots \times p_d$.

Let $\delta = \frac{4\epsilon}{3d}$. Suppose that for every $i, j \in [d] \times [r]$ it holds that $\frac{1+\delta}{r} \leq p_i(j) \geq \frac{1-\delta}{r}$. Note that $d\delta = \frac{4\epsilon}{3} \leq \ln(1+2\epsilon) \leq 2\epsilon$. Then for every $v \in [r]^d$,

$$\mathop{\mathbb{P}}_{u \sim \mu}[\mathsf{block}(u) = v] = \prod_{i=1}^{d} p_i(v_i) \begin{cases} \leq (1+\delta)^d r^{-d} \leq e^{d\delta} r^{-d} \leq (1+2\epsilon) r^{-d} \\ \geq (1-\delta)^d r^{-d} \geq (1-d\delta) r^{-d} \geq (1-2\epsilon) r^{-d}. \end{cases}$$

So

$$\|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} = \frac{1}{2}\sum_{v\in[r]^d} |\mathop{\mathbb{P}}_{u\sim\mu}[\mathsf{block}(u) = v] - r^{-d}| \le \frac{1}{2}\sum_{v\in[r]^d} 2\epsilon r^{-d} = \epsilon\,.$$

By Lemma 3.2.6 and the union bound, the probability that there is some $i \in [d], j \in [r]$ that satisfies $p_i(j) < (1-\delta)/r$ is at most $4rd \cdot e^{-\frac{\epsilon^2 m}{18rd^2}}$. □

3.3 Testing Monotonicity

The main result of [BCS20] is a "domain reduction" theorem, allowing a change of domain from $[n]^d$ to $[r]^d$ where $r = \mathrm{poly}(d/\epsilon)$; by applying this theorem together with their earlier $\widetilde{O}(\frac{d^{5/6}}{\epsilon^{4/3}}\,\mathrm{poly}\log(dn))$-query tester in [**?**] for the uniform distribution on $[n]^d$, they obtain a tester for monotone functions with query complexity independent of n. Our result replaces this domain reduction method with a simpler two-page argument, and gives a different generalization to the distribution-free case. We present the short proof of the domain reduction result for $[n]^d$ in the next section, and follow it with the general tester monotonicity under product distributions in Section 3.3.2.

3.3.1 Testing Monotonicity on the Hypergrid

Our monotonicity tester will use as a subroutine the following tester for *diagonal* functions. For a hypergrid $[n]^d$, a *diagonal* is a subset of points $\{x \in [n]^d : x = v + \lambda\vec{1}, \lambda \in \mathbb{Z}\}$ defined by some $v \in [n]^d$. A function $f : [n]^d \to \{0,1\}$ is a *diagonal function* if it has at most one 1-valued point in each diagonal.

Lemma 3.3.1. *There is an ϵ-tester for diagonal functions on $[n]^d$, with one-sided error and query complexity $O\left(\frac{1}{\epsilon}\log^2(1/\epsilon)\right)$.*

Proof. For each $t \in [n]$ let D_t be the set of diagonals with size t. For any $x \in [n]^d$ let $\mathrm{diag}(x)$ be the unique diagonal that contains x. For input $f : [n]^d \to \{0,1\}$ and any $x \in [n]^d$, let $R(x) = \frac{|\{y\in\mathrm{diag}(x): f(y)=1\}|}{|\mathrm{diag}(x)|}$.

Suppose that f is ϵ-far from diagonal. Then f must have at least ϵn^d 1-valued points; otherwise we could set each 1-valued point to 0 to obtain the constant 0 function. Now

observe

$$\begin{aligned}
\mathop{\mathbb{E}}_{x\sim[n]^d}[R(x)] &= \mathop{\mathbb{E}}_{x\sim[n]^d}\left[\sum_{t=1}^{n}\sum_{L\in D_t}\mathbb{1}\left[\mathrm{diag}(x)=L\right]\frac{|\{y\in L: f(y)=1\}|}{t}\right]\\
&= \sum_{t=1}^{n}\sum_{L\in D_t}\mathop{\mathbb{P}}_{x\sim[n]^d}[x\in L]\frac{|\{y\in L: f(y)=1\}|}{t}\\
&= \sum_{t=1}^{n}\sum_{L\in D_t}\frac{t}{n^d}\frac{|\{y\in L: f(y)=1\}|}{t}\\
&= \frac{1}{n^d}|\{y\in[n]^d: f(y)=1\}| \geq \epsilon\,.
\end{aligned}$$

For each i, define $A_i = \left\{x\in[n]^d : \frac{1}{2^i} < R(x) \leq \frac{1}{2^{i-1}}\right\}$. Let $k = \log(4/\epsilon)$. Then

$$\begin{aligned}
\epsilon \leq \mathbb{E}\left[R(x)\right] &\leq \sum_{i=1}^{\infty}\frac{|A_i|}{n^d}\max_{x\in A_i}R(x) \leq \sum_{i=1}^{\infty}\frac{|A_i|}{n^d 2^{i-1}} \leq \sum_{i=1}^{k}\frac{|A_i|}{n^d 2^{i-1}} + \sum_{i=k+1}^{\infty}\frac{1}{2^{i-1}}\\
&\leq \sum_{i=1}^{k}\frac{|A_i|}{n^d 2^{i-1}} + \frac{1}{2^{k-1}} \leq \sum_{i=1}^{k}\frac{|A_i|}{n^d 2^{i-1}} + \frac{\epsilon}{2}\\
\implies \frac{\epsilon}{2} &\leq \sum_{i=1}^{k}\frac{|A_i|}{n^d 2^{i-1}}\,.
\end{aligned}$$

Therefore there is some $\ell\in[k]$ such that $|A_\ell| \geq \frac{\epsilon n^d 2^{\ell-1}}{2k}$.

The tester is as follows. For each $i\in[k]$:

1. Sample $p = \frac{k}{\epsilon 2^{i-2}}\ln(6)$ points $x_1,\ldots,x_p \sim [n]^d$.
2. For each $j\in[p]$, sample $q = 2^{i+2}\ln(12)$ points $y_1,\ldots,y_q$ from $\mathrm{diag}(x_i)$ and reject if there are two distinct 1-valued points in the sample.

The query complexity of the tester is $\sum_{i=1}^{k} 4^2\ln(6)\ln(12)\frac{k}{\epsilon 2^i}2^i = O\left(\frac{1}{\epsilon}\log^2(1/\epsilon)\right)$.

The tester will clearly accept any diagonal function. Now suppose that f is ϵ-far from having this property, and let $\ell\in[k]$ be such that $|A_\ell| \geq \frac{\epsilon n^d 2^{\ell-2}}{k}$. On iteration $i=\ell$, the algorithm samples $p = \frac{k}{\epsilon 2^{\ell-2}}\ln(6)$ points $x_1,\ldots,x_p$. The probability that $\forall j\in[p], x_j\notin A_\ell$ is at most

$$\left(1-\frac{|A_\ell|}{n^d}\right)^p \leq \left(1-\frac{\epsilon 2^{\ell-2}}{k}\right)^p \leq \exp\left(-\frac{\epsilon p 2^{\ell-2}}{k}\right) \leq 1/6\,.$$

Now assume that there is some $x_j \in A_\ell$, so that $R(x_j) > 2^{-\ell}$. Let $A, B \subset \mathrm{diag}(x_j)$ be disjoint subsets that partition the 1-valued points in $\mathrm{diag}(x_i)$ into equally-sized parts. Then for y sampled uniformly at random from $\mathrm{diag}(x_j)$, $\mathbb{P}[y \in A], \mathbb{P}[y \in B] \geq 2^{-(\ell+1)}$. The probability that there are at least 2 distinct 1-valued points in $y_1, \ldots, y_q$ sampled by the algorithm is at least the probability that one of the first $q/2$ samples is in A and one of the last $q/2$ samples is in B. This fails to occur with probability at most $2(1-2^{-(\ell+1)})^{q/2} \leq 2e^{-q2^{-(\ell+2)}} \leq 1/6$. So the total probability of failure is at most $2/6 = 1/3$. □

Theorem 3.3.2. *There is a non-adaptive monotonicity tester on domain $[n]^d$ with one-sided error and query complexity $\widetilde{O}\left(\frac{d^{5/6}}{\epsilon^{4/3}}\right)$.*

Proof. Set $r = \lceil 4d/\epsilon \rceil$, and assume without loss of generality that r divides n. Partition $[n]$ into r intervals $B_i = \{(i-1)(n/r)+1, \ldots, i(n/r)\}$. For each $v \in [r]^d$ write $B_v = B_{v_i} \times \cdots \times B_{v_d}$. Define $\mathsf{block} : [n]^d \to [r]^d$ where $\mathsf{block}(x)$ is the unique vector $v \in [r]^d$ such that $x \in B_v$. Define $\mathsf{block}^{-\downarrow}(v) = \min\{x \in B_v\}$ and $\mathsf{block}^{-\uparrow}(v) = \max\{x \in B_v\}$, where the minimum and maximum are with respect to the natural ordering on $[n]^d$. For $f : [n]^d \to \{0,1\}$, write $f^{\mathsf{block}} : [r]^d \to \{0,1\}$, $f^{\mathsf{block}}(v) = f(\mathsf{block}^{-\downarrow}(v))$. We may simulate queries v to f^{block} by returning $f(\mathsf{block}^{-\downarrow}(v))$. We will call $v \in [r]^d$ a *boundary block* if $f(\mathsf{block}^{-\downarrow}(v)) \neq f(\mathsf{block}^{-\uparrow}(v))$.

The test proceeds as follows: On input $f : [n]^d \to \{0,1\}$ and a block $v \in [r]^d$, define the following functions:

$$g : [n]^d \to \{0,1\}, \qquad g(x) = \begin{cases} f^{\mathsf{block}}(\mathsf{block}(x)) & \text{if } \mathsf{block}(x) \text{ is not a boundary block} \\ f(x) & \text{if } \mathsf{block}(x) \text{ is a boundary block.} \end{cases}$$

$$b : [r]^d \to \{0,1\}, \qquad b(v) = \begin{cases} 0 & \text{if } v \text{ is not a boundary block} \\ 1 & \text{if } v \text{ is a boundary block.} \end{cases}$$

$$h : [r]^d \to \{0,1\}, \qquad h(v) = \begin{cases} f^{\mathsf{block}}(v) & \text{if } v \text{ is not a boundary block} \\ 0 & \text{if } v \text{ is a boundary block.} \end{cases}$$

Queries to each of these functions can be simulated by 2 or 3 queries to f. The tester performs:

1. Test whether $g = f$, or whether $\mathsf{dist}(f, g) > \epsilon/4$, using $O(1/\epsilon)$ queries.
2. Test whether b is diagonal, or is $\epsilon/4$-far from diagonal, using Lemma 3.3.1, with $O\left(\frac{1}{\epsilon}\log^2(1/\epsilon)\right)$ queries.

3. Test whether h is monotone or $\epsilon/4$-far from monotone, using the tester of [**?**] with $\widetilde{O}\left(\frac{d^{5/6}}{\epsilon^{4/3}}\right)$ queries.

Claim 3.3.3. *If f is monotone, the tester passes all 3 tests with probability 1.*

Proof of claim. To see that $g = f$, observe that if $v = \mathsf{block}(x)$ is not a boundary block then $f(\mathsf{block}^{-\downarrow}(v)) = f(\mathsf{block}^{-\uparrow}(v))$. If $f(x) \neq f^{\mathsf{block}}(\mathsf{block}(x))$ then $f(x) \neq f(\mathsf{block}^{-\downarrow}(v))$ and $f(x) \neq f(\mathsf{block}^{-\uparrow}(v))$ while $\mathsf{block}^{-\downarrow}(v) \preceq x \preceq \mathsf{block}^{-\uparrow}(v)$, and this is a violation of the monotonicity of f. Therefore f will pass the first test with probability 1.

To see that f passes the second test with probability 1, observe that if f had 2 boundary blocks in some diagonal, then there are boundary blocks $u, v \in [r]^d$ such that $\mathsf{block}^{-\uparrow}(u) \prec \mathsf{block}^{-\downarrow}(v)$. But then there is $x, y \in [n]^d$ such that $\mathsf{block}(x) = u, \mathsf{block}(y) = v$ and $f(x) = 1, f(y) = 0$; since $x \preceq \mathsf{block}^{-\uparrow}(u) \prec \mathsf{block}^{-\downarrow}(v) \preceq y$, this contradicts the monotonicity of f. So f has at most 1 boundary block in each diagonal.

To see that h is monotone, it is sufficient to consider the boundary blocks, since all other values are the same as f^{block}. Let $v \in [r]^d$ be a boundary block, so there exist $x, y \in [n]^d$ such that $\mathsf{block}(x) = \mathsf{block}(y)$ and $f(x) = 1, f(y) = 0$. Suppose $u \prec v$ is not a boundary block (if it is a boundary block then $h(u) = h(v) = 0$). If $h(u) = 1$ then $f(\mathsf{block}^{-\downarrow}(u)) = 1$, but $\mathsf{block}^{-\downarrow}(u) \prec \mathsf{block}^{-\downarrow}(v) \preceq y$ while $f(\mathsf{block}^{-\downarrow}(u)) > f(y)$, a contradiction. So it must be that $h(u) = 0$ whenever $u \prec v$. For any block $u \in [r]^d$ such that $v \prec u$, we have $0 = h(v) \leq h(u)$, so monotonicity holds. Since the tester of Black, Chakrabarty, & Seshadhri has one-sided error, the test passes with probability 1. □

Claim 3.3.4. *If g is $\epsilon/4$-close to f, b is $\epsilon/4$-close to diagonal, and h is $\epsilon/4$-close to monotone, then f is ϵ-close to monotone.*

Proof of claim. Let $h^{\mathsf{coarse}} : [n]^d \to \{0, 1\}$ be the function $h^{\mathsf{coarse}}(x) = h(\mathsf{block}(x))$. Suppose that $f(x) \neq h^{\mathsf{coarse}}(x)$. If $v = \mathsf{block}(x)$ is not a boundary block of f then $h^{\mathsf{coarse}}(x) = h(v) = f^{\mathsf{block}}(v) = g(x)$, so $f(x) \neq g(x)$. If v is a boundary block then $h^{\mathsf{coarse}}(x) = h(v) = 0$ so $f(x) = 1$, and $b(v) = 1$.

Suppose for contradiction that there are more than $\frac{\epsilon}{2}r^d$ boundary blocks $v \in [r]^d$, so there are more than $\frac{\epsilon}{2}r^d$ 1-valued points of b. Any diagonal function has at most dr^{d-1} 1-valued points. Therefore the distance of b to diagonal is at least

$$r^{-d}\left(\frac{\epsilon}{2}r^d - dr^{d-1}\right) = \frac{\epsilon}{2} - \frac{d}{r} = \frac{\epsilon}{2} - \frac{\epsilon}{4} = \frac{\epsilon}{4},$$

a contradiction. So f has at most $\frac{\epsilon}{2}r^d$ boundary blocks. Now

$$\begin{aligned}\mathsf{dist}(f, h^{\mathsf{coarse}}) &= \mathsf{dist}(f, g) + \mathop{\mathbb{P}}_{x\sim[n]^d}[f(x) = 1, \mathsf{block}(x) \text{ is a boundary block}] \\ &\leq \frac{\epsilon}{4} + r^{-d} \cdot \frac{\epsilon r^d}{2} = \frac{3}{4}\epsilon\,.\end{aligned}$$

Let $p : [r]^d \to \{0,1\}$ be a monotone function minimizing the distance to h, and let $p^{\mathsf{coarse}} : [n]^d \to \{0,1\}$ be the function $p^{\mathsf{coarse}}(x) = p(\mathsf{block}(x))$. Then

$$\mathsf{dist}(h^{\mathsf{coarse}}, p^{\mathsf{coarse}}) = \mathop{\mathbb{P}}_{x\sim[n]^d}[h(\mathsf{block}(x)) \neq p(\mathsf{block}(x))] = \mathop{\mathbb{P}}_{v\sim[r]^d}[h(v) \neq p(v)] \leq \epsilon/4\,.$$

Finally, the distance of f to the nearest monotone function is at most

$$\mathsf{dist}(f, p^{\mathsf{coarse}}) \leq \mathsf{dist}(f, h^{\mathsf{coarse}}) + \mathsf{dist}(h^{\mathsf{coarse}}, p^{\mathsf{coarse}}) \leq \frac{3}{4}\epsilon + \frac{1}{4}\epsilon = \epsilon\,. \qquad \square$$

These two claims suffice to establish the theorem. $\square$

3.3.2 Monotonicity Testing under Product Distributions

The previous section used a special case of downsampling, tailored for the uniform distribution over $[n]^d$. We will call a product distribution $\mu = \mu_1 \times \cdots \times \mu_d$ over $\mathbb{R}^d$ *continuous* if each of its factors μ_i are continuous (i.e. absolutely continuous with respect to the Lebesgue measure). The proof for discrete distributions is in Section 3.8.4.

Theorem 1.2.11. *There is a one-sided non-adaptive tester for monotonicity under product distributions over $\mathbb{R}^d$, with query complexity $\widetilde{O}(d^{5/6}/\epsilon^{4/3})$ and sample complexity $\widetilde{O}((d/\epsilon)^3)$.*

Proof. We follow the proof of Theorem 3.3.2, with some small changes. Let $r = \lceil 16d/\epsilon \rceil$. The tester first samples a grid X with length $m = O\left(\frac{rd^2}{\epsilon^2}\log(rd)\right)$ and constructs the induced $(r+2)$-block partition, with cells labeled $\{0,\ldots,r+1\}^d$. We call a block $v \in \{0,\ldots,r+1\}^d$ *upper extreme* if there is some $i \in [d]$ such that $v_i = r+1$, and we call it *lower extreme* if there is some $i \in [d]$ such that $v_i = 0$ but v is not upper extreme. Call the upper extreme blocks U and the lower extreme blocks L. Note that $[r]^d = \{0,\ldots,r+1\}^d \setminus (U \cup L)$.

For each $v \in [r]^d$, we again define $\mathsf{block}^{-\uparrow}(v), \mathsf{block}^{-\downarrow}(v)$ as, respectively, the supremal and infimal point $x \in \mathbb{R}^d$ such that $\mathsf{block}(x) = v$. The algorithm will ignore the extreme

blocks $U \cup L$, which do not have a supremal or an infimal point. Therefore it is not defined whether these blocks are boundary blocks.

By Lemma 3.2.7, with probability at least $5/6$, we will have $\|\mathsf{block}(\mu) - \mathsf{unif}(\{0, \dots, r+1\})\|_{\mathsf{TV}} \le \epsilon/8$. We define b, h as before, with domain $[r]^d$. Define g similarly but with domain $\mathbb{R}^d$ and values

$$g(x) = \begin{cases} 1 & \text{if } \mathsf{block}(x) \in U \\ 0 & \text{if } \mathsf{block}(x) \in L \\ f(x) & \text{if } \mathsf{block}(x) \in [n]^d \text{ is a boundary block} \\ f^{\mathsf{block}}(\mathsf{block}(x)) & \text{otherwise.} \end{cases}$$

If f is monotone, it may now be the case $f \neq g$, but we will have $f(x) = g(x)$ for all x with $\mathsf{block}(x) \in [r]^d$, where the algorithm will make its queries. The algorithm will test whether $f(x) = g(x)$ on all x with $\mathsf{block}(x) \in [r]^d$, or $\epsilon/8$-far from this property, which can be again done with $O(1/\epsilon)$ samples. Note that if f is $\epsilon/8$-close to having this property, then

$$\begin{aligned} \mathsf{dist}_\mu(f, g) &\le \mathop{\mathbb{P}}_{x \sim \mu}\left[\mathsf{block}(x) \notin [n]^d\right] + \epsilon/8 \\ &\le \frac{d(r+2)^{d-1}}{(r+2)^d} + \epsilon/8 + \|\mathsf{block}(\mu) - \mathsf{unif}([r]^d \cup U \cup L)\|_{\mathsf{TV}} \\ &\le \frac{\epsilon}{16} + \frac{\epsilon}{8} + \frac{\epsilon}{4} \le \frac{\epsilon}{2}\,. \end{aligned}$$

The algorithm then proceeds as before, with error parameter $\epsilon/2$. To test whether $g = f$, the algorithm samples from μ and throws away any sample $x \in \mathbb{R}^d$ with $\mathsf{block}(x) \notin [r]^d$. It then tests b and h using the uniform distribution on $[r]^d$. It suffices to prove the following claim, which replaces Claim 3.3.4.

Claim 3.3.5. *If g is $\epsilon/2$-close to f, b is $\epsilon/16$-close to diagonal, and h is $\epsilon/8$-close to monotone, then f is ϵ-close to monotone.*

Proof of claim. Let $p : [r]^d \to \{0, 1\}$ be a monotone function minimizing the distance to h. Then $p(v) \neq h(v)$ on at most $\frac{\epsilon r^d}{8}$ blocks $v \in [r]^d$. Define $p^{\mathsf{coarse}} : \mathbb{R}^d \to \{0, 1\}$ as $p^{\mathsf{coarse}}(x) = p(\mathsf{block}(x))$ when $\mathsf{block}(x) \in [r]^d$, and $p^{\mathsf{coarse}}(x) = g(x)$ when $\mathsf{block}(x) \in U \cup L$. Note that p^{coarse} is monotone.

By the triangle inequality,

$$\mathsf{dist}_\mu(f, p^{\mathsf{coarse}}) \le \mathsf{dist}_\mu(f, g) + \mathsf{dist}_\mu(g, p^{\mathsf{coarse}})\,.$$

From above, we know $\mathsf{dist}_\mu(f,g) \le \epsilon/2$. To bound the second term, observe that since b is $\epsilon/16$-close to diagonal, there are at most

$$\frac{\epsilon}{16}r^d + dr^{d-1} \le \frac{\epsilon}{16}r^d + \frac{d}{r}r^d \le \frac{\epsilon}{16}r^d + \frac{\epsilon}{16}r^d = \frac{\epsilon}{8}r^d$$

boundary blocks. Then observe that if $g(x) \neq p^{\mathsf{coarse}}(x)$ then $\mathsf{block}(x) \in [r]^d$ and either $\mathsf{block}(x)$ is a boundary block, or $g(x) = f^{\mathsf{block}}(\mathsf{block}(x)) = h(\mathsf{block}(x))$ and $h(\mathsf{block}(x)) \neq p(\mathsf{block}(x))$. Then

$$\begin{aligned}\mathsf{dist}_\mu(g, p^{\mathsf{coarse}}) &\le \left(\frac{1}{(r+2)^d}\sum_{v\in[r]^d} \mathbb{1}\left[v \text{ is a boundary block, or } h(v) \neq p(v)\right]\right) \\ &\quad + \|\mathsf{block}(\mu) - \mathsf{unif}(\{0,\dots,r+1\}^d)\|_{\mathsf{TV}} \\ &\le \frac{\epsilon r^d}{8r^d} + \frac{\epsilon r^d}{8r^d} + \frac{\epsilon}{4} \le \frac{\epsilon}{2}\,. \end{aligned}$$

□

This concludes the proof. □

3.4 Learning and Testing Functions of Convex Sets

Let $\mathcal{C}$ be the set of functions $f : \mathbb{R}^d \to \{\pm 1\}$ such that $f^{-1}(1)$ is convex. This class of functions was one of the first classes of functions in continuous space $\mathbb{R}^d \to \{0,1\}$ to be studied in the property testing literature. This problem has been studied in various models of testing [Ras03, RV04, CFSS17, BMR19, BB20]. In this section we consider the labeled-sample model, as in Chapter 2.

Chen *et al.* [CFSS17] gave a tester for $\mathcal{C}$ in this model under the Gaussian distribution on $\mathbb{R}^d$ with one-sided error and sample complexity $(d/\epsilon)^{O(d)}$, along with a lower bound (for one-sided testers) of $2^{\Omega(d)}$. We match their upper bound while generalizing the tester to be distribution-free under product distributions. This is proved in Section 3.4.1.

Theorem 3.4.1. *There is a sample-based distribution-free one-sided ϵ-tester for $\mathcal{C}$ under (finite or continuous) product distributions that uses at most $O\left(\left(\frac{6d}{\epsilon}\right)^d\right)$ samples.*

Recall the definition of an (ϵ_1, ϵ_2)-tolerant tester from Definition 1.2.3. Tolerantly testing convex sets has been studied by [BMR16] for the uniform distribution over the 2-dimensional grid, but not (to the best of our knowledge) in higher dimensions. We obtain

a sample-based tolerant tester (and distance approximator) for convex sets in high dimension. In fact, we get a tolerant tester for the class of all functions of k convex sets, defined as follows.

Definition 3.4.2 (Function Composition)**.** Write $\mathcal{B}_k$ for the set of all Boolean functions $g : \{\pm 1\}^k \to \{\pm 1\}$. For a set $\mathcal{H}$ of functions $h : \mathbb{R}^d \to \{\pm 1\}$, we will define the composition $\mathcal{B}_k \circ \mathcal{H}$ as the set of functions of the form $f(x) = g(h_1(x), \ldots, h_k(x))$ where $g \in \mathcal{B}_k$ and each h_i belongs to $\mathcal{H}$.

We obtain a distance approximator for $\mathcal{B}_k \circ \mathcal{C}$, proved in Section 3.4.2.

Theorem 3.4.3. *Let $\mathcal{B}' \subset \mathcal{B}_k$. There is a sample-based distribution-free algorithm under (finite or continuous) product distributions that approximates distance to $\mathcal{B}' \circ \mathcal{C}$ up to additive error ϵ using $O\left(\frac{1}{\epsilon^2}\left(\frac{3dk}{\epsilon}\right)^d\right)$ samples. Setting $\epsilon = (\epsilon_2 - \epsilon_1)/2$ we obtain an (ϵ_1, ϵ_2)-tolerant tester with sample complexity $O\left(\frac{1}{(\epsilon_2-\epsilon_1)^2}\left(\frac{6dk}{\epsilon_2-\epsilon_1}\right)^d\right)$.*

General distribution-free learning of convex sets is not possible, since this class has infinite VC dimension. However, they can be learned under the Gaussian distribution. Non-agnostic learning under the Gaussian was studied by Vempala [Vem10a, Vem10b]. Agnostic learning under the Gaussian was studied by Klivans, O'Donnell, & Servedio [KOS08] who presented a learning algorithm with complexity $d^{O(\sqrt{d}/\epsilon^4)}$, and a lower bound of $2^{\Omega(\sqrt{d})}$.

Unlike the Gaussian, there is a trivial lower bound of $\Omega(2^d)$ in arbitrary product distributions, because any function $f : \{\pm 1\}^d \to \{0, 1\}$ belongs to this class. However, unlike the general distribution-free case, we show that convex sets (or any functions of convex sets) can be learned under unknown product distributions. This is proved in Section 3.4.3.

Theorem 3.4.4. *There is an agnostic learning algorithm for $\mathcal{B}_k \circ \mathcal{C}$ under (finite or continuous) product distributions over $\mathbb{R}^d$, with time complexity $O\left(\frac{1}{\epsilon^2} \cdot \left(\frac{6dk}{\epsilon}\right)^d\right)$.*

All our algorithms will follow from more general results that actually hold for any class $\mathcal{H}$ with bounded r-block boundary size; i.e. bounded block-boundary size is sufficient to guarantee learnability in product distributions. We first state some important properties of the composed function class $\mathcal{B}_k \circ \mathcal{H}$ that make downsampling effective for these classes.

Proposition 3.4.5. *Let $\mathcal{H}$ be any class of functions $\mathbb{R}^d \to \{\pm 1\}$ and fix any r. Then* $\mathsf{bbs}(\mathcal{B}_k \circ \mathcal{H}, r) \leq k \cdot \mathsf{bbs}(\mathcal{H}, r)$.

Proof. If $f(\cdot) = g(h_1(\cdot), \ldots, h_k(\cdot))$ is not constant on $\mathsf{block}^{-1}(v)$ then one of the h_i is not constant on that block. Therefore $\mathsf{bbs}(f, r) \le \sum_{i=1}^{k} \mathsf{bbs}(h_i, r) \le k \cdot \mathsf{bbs}(\mathcal{H}, r)$. □

Lemma 3.4.6. *For any r, $\mathsf{bbs}(\mathcal{B}_k \circ \mathcal{C}, r) \le 2dkr^{d-1}$.*

Proof. We prove $\mathsf{bbs}(\mathcal{C}, r) \le 2dr^{d-1}$ by induction on d; the result will hold by Proposition 3.4.5. Let $\mathsf{bbs}(\mathcal{C}, r, d)$ be the r-block boundary size in dimension d. Recall that block $v \in [r]^d$ is the set $B_v = B_{1,v_1} \times \cdots \times B_{d,v_d}$ where $B_{i,j} = (a_{i,j-1}, a_{i,j}]$ for some $a_{i,j-1} < a_{i,j}$. Let $f \in \mathcal{C}$.

For $d = 1$, if there are 3 intervals $B_{1,i_1}, B_{1,i_2}, B_{1,i_3}$, $i_1 < i_2 < i_3$, on which f is not constant, then within each interval the function takes both values $\{\pm 1\}$. Thus, there are points $a \in B_{1,i_1}, b \in B_{1,i_2}, c \in B_{1,i_3}$ such that $f(a) = 1, f(b) = -1, f(c) = 1$, which is a contradiction.

For each block B_v, let $A_v = \{a_{1,v_1}\} \times B_{2,v_2} \times \ldots \times B_{d,v_d}$ be the "upper face". For $d > 1$, let $P \subseteq [r]^d$ be the set of non-constant blocks B_v such that f is constant on the upper face and let Q be the set of non-constant blocks that are non-constant on the upper face, so that $\mathsf{bbs}(f, r, d) = |P| + |Q|$. We argue that $|P| \le 2r^{d-1}$: for a vector $w \in [r]^{d-1}$ define the line $L_w := \{v \in [r]^d \mid \forall i > 1, v_i = w_i\}$. If $|P \cap L_w| \ge 3$ then there are $t, u, v \in L_w$ with $t < u < v$ such that f is constant on A_t, A_u, A_v but non-constant on B_t, B_u, B_v. Let x, y, z be points in B_t, B_u, B_v respectively such that $f(x) = f(y) = f(z) = 1$. If f is constant -1 on A_t or A_u then there is a contradiction since the lines through (x, y) and (y, z) pass through A_t, A_u; so f is constant 1 on A_t, A_u. But then there is a point $q \in A_u$ with $f(q) = -1$, which is a contradiction since it is within the convex hull of A_t, A_u. So $|L_w \cap P| < 3$; since there are at most r^{d-1} lines L_w, $|P| \le 2r^{d-1}$.

To bound $|Q|$, observe that for each block $v \in Q$, f is non-constant on the plane $\{a_{1,v_1}\} \times \mathbb{R}^{d-1}$, there are $(r-1)$ such planes, f is convex on each, and the r-block partition induces an r-block partition on the plane where f is non-constant on the corresponding block. Then, by induction $|Q| \le (r-1) \cdot \mathsf{bbs}(\mathcal{C}, r, d-1) \le 2(d-1)(r-1)r^{d-2}$. So

$$\mathsf{bbs}(\mathcal{C}, r, d) \le 2\left[(d-1)(r-1)r^{d-2} + r^{d-1}\right] < 2dr^{d-1}\,.$$ □

The above two lemmas combine to show that

$$r^{-d} \cdot \mathsf{bbs}(\mathcal{B}_k \circ \mathcal{C}, r) \le r^{-d}(2dkr^{d-1}) = 2dk/r \le \epsilon$$

when $r = \lceil 2dk/\epsilon \rceil$.

3.4.1 Sample-Based One-sided Tester

First, we prove a one-sided sample-based tester for convex sets.

Theorem 3.4.1. *There is a sample-based distribution-free one-sided ϵ-tester for $\mathcal{C}$ under (finite or continuous) product distributions that uses at most $O\left(\left(\frac{6d}{\epsilon}\right)^d\right)$ samples.*

Proof. We prove the result for continuous distributions. The proof for finite distributions is in Theorem 3.8.18.

On input distribution μ and function f, let $r = \lceil 6d/\epsilon \rceil$ so that $r^{-d} \cdot \mathsf{bbs}(\mathcal{C}, r) \leq \epsilon/3$.

1. Sample a grid X of size $m = O\left(\frac{rd^2}{\epsilon^2}\log(rd/\epsilon)\right)$ large enough that Lemma 3.2.7 guarantees $\|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} < \epsilon/9$ with probability $5/6$.
2. Take $q = O\left(\frac{r^d}{\epsilon}\right)$ samples Q and accept if there exists $h \in \mathcal{C}$ such that $f(x) = h^{\mathsf{coarse}}(x)$ on all $x \in Q$ that are not in a boundary block of h.

This tester is one-sided since for any $h \in \mathcal{C}$, $h(x) = h^{\mathsf{coarse}}(x)$ for all $x \in Q$ that are not in a boundary block, regardless of whether the r-block decomposition induced by X satisfies $\|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} \leq \epsilon/3$. Now suppose that $\mathsf{dist}_\mu(f, \mathcal{C}) > \epsilon$, and suppose that $\|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} \leq \epsilon$. For $h \in \mathcal{C}$, let $B_h \subseteq [r]^d$ be the set of non-constant blocks. If $\exists h \in \mathcal{C}$ such that $\mathop{\mathbb{P}}_{x\sim\mu}[h^{\mathsf{coarse}}(x) \neq f(x) \wedge \mathsf{block}(x) \notin B_h] < \epsilon/9$, then

$$\begin{aligned}
\mathsf{dist}_\mu(f, h^{\mathsf{coarse}}) &\leq \mathop{\mathbb{P}}_{x\sim\mu}[\mathsf{block}(x) \in B_h] + \mathop{\mathbb{P}}_{x\sim\mu}[h^{\mathsf{coarse}}(x) \neq f(x) \wedge \mathsf{block}(x) \notin B_h] \\
&\leq r^{-d} \cdot \mathsf{bbs}(\mathcal{C}, r) + \|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} + \frac{\epsilon}{9} \\
&\leq \left(\frac{1}{3} + \frac{2}{9}\right)\epsilon = \frac{5}{9} \cdot \epsilon\,.
\end{aligned}$$

Therefore

$$\begin{aligned}
\mathsf{dist}_\mu(f, h) &\leq \mathsf{dist}_\mu(f, h^{\mathsf{coarse}}) + \mathsf{dist}_\mu(h^{\mathsf{coarse}}, h) \\
&\leq \mathsf{dist}_\mu(f, h^{\mathsf{coarse}}) + r^{-d} \cdot \mathsf{bbs}(\mathcal{C}, r) + \|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} \\
&\leq \frac{5}{9}\epsilon + \frac{1}{3} \cdot \epsilon + \frac{1}{9}\epsilon = \epsilon\,,
\end{aligned}$$

a contradiction. So it must be that for every $h \in \mathcal{C}, \mathbb{P}[f(x) \neq h^{\mathsf{coarse}}(x) \wedge x \notin B_h] \geq \epsilon/9$. There are at most $\binom{r^d}{\frac{\epsilon}{3}r^d} \leq (3e/\epsilon)^{\epsilon r^d/3}$ choices of boundary set B. Because the 1-valued blocks must be the convex hull of the boundary points, for each boundary set B there are at most 2 choices of function h^{coarse} with boundary B (with a second choice occurring when the complement of h^{coarse} is also a convex set with the same boundary). Therefore, by the union bound, the probability that f is accepted is at most

$$\left(\frac{3e}{\epsilon}\right)^{\frac{\epsilon}{3}r^d} \cdot \left(1 - \frac{\epsilon}{9}\right)^q \leq e^{\epsilon\left(\frac{r^d}{3} - \frac{q}{9}\right)},$$

which is at most $1/6$ for sufficiently large $q = O\left(r^d + \frac{1}{\epsilon}\right)$. □

3.4.2 Sample-Based Distance Approximator

Our sample-based distance approximator follows from the following general result.

Lemma 3.4.7. *For any set $\mathcal{H}$ of functions $\mathbb{R}^d \to \{\pm 1\}$, $\epsilon > 0$, and r satisfying $r^{-d} \cdot \mathsf{bbs}(\mathcal{H}, r) \leq \epsilon/3$, there is a sample-based distribution-free algorithm for product distributions that approximates distance to $\mathcal{H}$ up to additive error ϵ using $O\left(\frac{r^d}{\epsilon^2}\right)$ samples.*

Proof. On input distribution μ and function $f : \mathbb{R}^d \to \{0, 1\}$, let $r = 3dk/\epsilon$, then:

1. Sample a grid X of size $m = O(\frac{rd^2}{\epsilon^2} \log \frac{rd}{\epsilon})$ large enough that Lemma 3.2.7 guarantees $\|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} < \epsilon/3$ with probability 5/6.
2. Let $\mathcal{H}^{\mathsf{coarse}}$ be the set of all functions h^{coarse} where $h \in \mathcal{H}$; note that $|\mathcal{H}^{\mathsf{coarse}}| \leq 2^{r^d}$.
3. Draw $q = O\left(\frac{r^d}{\epsilon^2}\right)$ samples Q and output the distance on Q to the nearest function in $\mathcal{H}^{\mathsf{coarse}}$.

We argue that with probability at least 5/6, $\mathcal{H}^{\mathsf{coarse}}$ is an $\frac{5}{6}\epsilon$-cover of $\mathcal{H}$. With probability at least 5/6, $\|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} < \epsilon/6$. Then by Proposition 3.2.5, for any $h \in \mathcal{H}$,

$$\mathop{\mathbb{P}}_{x\sim\mu}[h(x) \neq h^{\mathsf{coarse}}(x)] \leq r^{-d} \cdot \mathsf{bbs}(f, r) + \|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} \leq \left(\frac{2}{3} + \frac{1}{6}\right)\epsilon = \frac{5}{6}\epsilon,$$

so $\mathcal{H}^{\mathsf{coarse}}$ is a $\frac{5}{6}\epsilon$-cover; assume this event occurs.

Write $\mathsf{dist}_Q(f,g) := \frac{1}{q}\sum_{x\in Q} \mathbb{1}\left[f(x) \neq g(x)\right]$. By the union bound and Hoeffding's inequality, with q samples we fail to get an estimate of $\mathsf{dist}_\mu(f, \mathcal{H}^{\mathsf{coarse}})$ up to additive error $\frac{1}{6}\epsilon$ with probability at most

$$\begin{aligned}
&|\mathcal{H}^{\mathsf{coarse}}| \cdot \max_{h^{\mathsf{coarse}} \in \mathcal{H}^{\mathsf{coarse}}} \mathop{\mathbb{P}}_{Q}\left[\left|\mathsf{dist}_\mu(f, h^{\mathsf{coarse}}) - \mathsf{dist}_Q(f, h^{\mathsf{coarse}})\right| > \frac{1}{6}\epsilon\right] \\
&\quad \leq |\mathcal{H}^{\mathsf{coarse}}| \exp\left(-2\frac{q\epsilon^2}{36}\right) < \frac{1}{6}\,,
\end{aligned}$$

for appropriately chosen $q = O\left(\frac{1}{\epsilon^2}\log(|\mathcal{H}^{\mathsf{coarse}}|)\right) = O\left(\frac{r^d}{\epsilon^2}\right)$. Assume this event occurs. We want to show that $|\mathsf{dist}_Q(f, \mathcal{H}^{\mathsf{coarse}}) - \mathsf{dist}_\mu(f, \mathcal{H})| \leq \epsilon$. Let $h \in \mathcal{H}$ minimize $\mathsf{dist}_\mu(f, h)$ so $\mathsf{dist}_\mu(f, h) = \mathsf{dist}_\mu(f, \mathcal{H})$. Then

$$\begin{aligned}
\mathsf{dist}_Q(f, \mathcal{H}^{\mathsf{coarse}}) &\leq \mathsf{dist}_Q(f, h^{\mathsf{coarse}}) \leq \mathsf{dist}_\mu(f, h^{\mathsf{coarse}}) + \frac{\epsilon}{6} \\
&\leq \mathsf{dist}_\mu(f, h) + \mathsf{dist}_\mu(h, h^{\mathsf{coarse}}) + \frac{\epsilon}{6} \leq \mathsf{dist}_\mu(f, \mathcal{H}) + \epsilon\,.
\end{aligned}$$

Now let $g \in \mathcal{H}$ minimize $\mathsf{dist}_Q(f, g^{\mathsf{coarse}})$ so $\mathsf{dist}_Q(f, g^{\mathsf{coarse}}) = \mathsf{dist}_Q(f, \mathcal{H}^{\mathsf{coarse}})$. Then

$$\begin{aligned}
\mathsf{dist}_Q(f, \mathcal{H}^{\mathsf{coarse}}) = \mathsf{dist}_Q(f, g^{\mathsf{coarse}}) &\geq \mathsf{dist}_\mu(f, g^{\mathsf{coarse}}) - \frac{\epsilon}{6} \geq \mathsf{dist}_\mu(f, h^{\mathsf{coarse}}) - \frac{\epsilon}{6} \\
&\geq \mathsf{dist}_\mu(f, h) - \mathsf{dist}_\mu(h, h^{\mathsf{coarse}}) - \frac{\epsilon}{6} \geq \mathsf{dist}_\mu(f, h) - \epsilon\,,
\end{aligned}$$

which concludes the proof. □

Applying the bound on $\mathsf{bbs}(\mathcal{B}_k, r)$ we conclude:

Theorem 3.4.3. *Let $\mathcal{B}' \subset \mathcal{B}_k$. There is a sample-based distribution-free algorithm under (finite or continuous) product distributions that approximates distance to $\mathcal{B}' \circ \mathcal{C}$ up to additive error ϵ using $O\left(\frac{1}{\epsilon^2}\left(\frac{3dk}{\epsilon}\right)^d\right)$ samples. Setting $\epsilon = (\epsilon_2 - \epsilon_1)/2$ we obtain an (ϵ_1, ϵ_2)-tolerant tester with sample complexity $O\left(\frac{1}{(\epsilon_2-\epsilon_1)^2}\left(\frac{6dk}{\epsilon_2-\epsilon_1}\right)^d\right)$.*

3.4.3 Agnostic Learning

We begin our learning results with an agnostic learning algorithm for functions of k convex sets: the class $\mathcal{B}_k \circ \mathcal{C}$. For a distribution $\mathcal{D}$ over $\mathbb{R}^d \times \{\pm 1\}$ and an r-block partition $\mathsf{block} : \mathbb{R}^d \to [r]^d$, define the distribution $\mathcal{D}^{\mathsf{block}}$ over $[r]^d \times \{\pm 1\}$ as the distribution of $(\mathsf{block}(x), b)$ when $(x, b) \sim \mathcal{D}$.

Lemma 3.4.8. *Let $\mathcal{H}$ be any set of functions $\mathbb{R}^d \to \{\pm 1\}$, let $\epsilon > 0$, and suppose r satisfies $r^{-d} \cdot \mathsf{bbs}(\mathcal{H}, r) \le \epsilon/3$. Then there is an distribution-free agnostic learning algorithm for continuous product distributions that learns $\mathcal{H}$ in $O\left(\frac{r^d + rd^2 \log(rd/\epsilon)}{\epsilon^2}\right)$ samples and time.*

Proof. On input distribution $\mathcal{D}$:

1. Sample a grid X of size $m = O(\frac{rd^2}{\epsilon^2} \log(rd/\epsilon))$ large enough that Lemma 3.2.7 guarantees $\|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} < \epsilon/3$ with probability $5/6$, where $\mathsf{block} : \mathbb{R}^d \to [r]^d$ is the induced r-block partition.
2. Agnostically learn a function $g : [r]^d \to \{\pm 1\}$ with error $\epsilon/3$ and success probability $5/6$ using $O(r^d/\epsilon^2)$ samples from $\mathcal{D}^{\mathsf{block}}$. Output the function $g \circ \mathsf{block}$.

The second step is accomplished via standard learning results ([SB14] Theorem 6.8): the number of samples required for agnostic learning is bounded by $O(1/\epsilon^2)$ multiplied by the logarithm of the number of functions in the class, and the number of functions $[r]^d \to \{\pm 1\}$ is 2^{r^d}. Assume that both steps succeed, which occurs with probability at least $2/3$. Let $f \in \mathcal{H}$ minimize $\mathbb{P}_{(x,b)\sim\mathcal{D}}[f(x) \ne b]$. By Proposition 3.2.5,

$$\mathbb{P}_{x\sim\mu}[f(x) \ne f^{\mathsf{coarse}}] \le r^{-d} \cdot \mathsf{bbs}(f, r) + \|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} < 2\epsilon/3\,.$$

Then

$$\begin{aligned}\mathbb{P}_{(x,b)\sim\mathcal{D}}[g(\mathsf{block}(x)) \ne b] &= \mathbb{P}_{(v,b)\sim\mathcal{D}^{\mathsf{block}}}[g(v) \ne b] \le \mathbb{P}_{(v,b)\sim\mathcal{D}^{\mathsf{block}}}\left[f^{\mathsf{block}}(v) \ne b\right] + \epsilon/3 \\ &= \mathbb{P}_{(x,b)\sim\mathcal{D}}[f^{\mathsf{coarse}}(x) \ne b] + \epsilon/3 < \mathbb{P}_{(x,b)\sim\mathcal{D}}[f(x) \ne b] + \epsilon\,. \qquad \square\end{aligned}$$

Lemma 3.4.8 then gives the following result for continuous product distributions, with the result for finite distributions following from Theorem 3.8.18.

Theorem 3.4.4. *There is an agnostic learning algorithm for $\mathcal{B}_k \circ \mathcal{C}$ under (finite or continuous) product distributions over $\mathbb{R}^d$, with time complexity $O\left(\frac{1}{\epsilon^2} \cdot \left(\frac{6dk}{\epsilon}\right)^d\right)$.*

3.5 Polynomial Regression and Learning Functions of Halfspaces

Let $\mathcal{H}$ be the set of halfspaces and recall from Definition 3.4.2 that $\mathcal{B}_k \circ \mathcal{H}$ is the class of arbitrary functions of k halfspaces. Intersections of k halfspaces (the most important

subclass of $\mathcal{B}_k \circ \mathcal{H}$) have VC dimension $\Theta(dk \log k)$ [BEHW89, CMK19], so the sample complexity of learning is known, but it is not possible to efficiently find k halfspaces whose intersection is correct on the sample, unless $\mathsf{P} = \mathsf{NP}$ [BR92]. Therefore the goal is to find efficient "improper" algorithms that output a function other than an intersection of k halfspaces. Several learning algorithms for intersections of k halfspaces actually work for arbitrary functions of k halfspaces. Klivans, O'Donnell, & Servedio [KOS04] gave a (non-agnostic) learning algorithm for $\mathcal{B}_k \circ \mathcal{H}$ over the uniform distribution on $\{\pm 1\}^d$ with complexity $d^{O(k^2/\epsilon^2)}$, Kalai, Klivans, Mansour, & Servedio [KKMS08] presented an agnostic algorithm with complexity $d^{O(k^2/\epsilon^4)}$ in the same setting using "polynomial regression".

Blais, O'Donnell, & Wimmer [BOW10] studied how to generalize polynomial regression to arbitrary product distributions. With their method, they obtained an agnostic learning algorithm for $\mathcal{B}_k \circ \mathcal{H}$ with complexity $(dn)^{O(k^2/\epsilon^4)}$ for product distributions $X_1 \times \cdots \times X_d$ where each $|X_i| = n$, and complexity $d^{O(k^2/\epsilon^4)}$ for the "polynomially bounded" continuous distributions. This is not a complete generalization, because, for example, on the grid $[n]^d$ its complexity depends on n. This prevents a full generalization to the domain $\mathbb{R}^d$. Their algorithm also requires some prior knowledge of the support or support size. We use a different technique and fully generalize the polynomial regression algorithm to arbitrary product distributions.

Theorem 1.2.12. *There is a distribution-free, improper agnostic learning algorithm for $\mathcal{B}_k \circ \mathcal{H}$ under (continuous or finite) product distributions over $\mathbb{R}^d$, with time complexity*

$$\min\left\{\left(\frac{dk}{\epsilon}\right)^{O\left(\frac{k^2}{\epsilon^4}\right)}, O\left(\frac{1}{\epsilon^2}\left(\frac{3dk}{\epsilon}\right)^d\right)\right\}.$$

Recall that downsampling reduces learning $\mathcal{H}$ in $\mathbb{R}^d$ to learning $\mathcal{H}^{\mathsf{block}}$ over $[r]^d$, and $\mathcal{H}^{\mathsf{block}}$ is *not* the set of halfspaces over $[r]^d$. Fortunately, agnostically learning a halfspaces h is commonly done by giving a bound on the degree of a polynomial p that approximates h [KOS04, KOS08, KKMS08], and we will show that a similar idea also suffices for learning $\mathcal{H}^{\mathsf{block}}$. In the next section, we present a general algorithm based on polynomial regression, and then introduce the Fourier analysis necessary to apply the general learning algorithm to halfspaces, polynomial threshold functions, and k-alternating functions.

3.5.1 A General Learning Algorithm

The learning algorithm in this section essentially replaces step 2 of the brute force algorithm (Lemma 3.4.8) with the "polynomial regression" algorithm of Kalai *et al.* [KKMS08]. Our

general algorithm is inspired by an algorithm of Canonne *et al.* [CGG+19] for tolerantly testing k-alternating functions over the uniform distribution on $[n]^d$; we state the regression algorithm as it appears in [CGG+19]. For a set $\mathcal{F}$ of functions, $\mathsf{span}(\mathcal{F})$ is the set of all linear combinations of functions in $\mathcal{F}$:

Theorem 3.5.1 ([KKMS08, CGG+19])**.** *Let μ be a distribution over $\mathcal{X}$, let $\mathcal{H}$ be a class of functions $\mathcal{X} \to \{\pm 1\}$ and $\mathcal{F}$ a collection of functions $\mathcal{X} \to \mathbb{R}$ such that for every $h \in \mathcal{H}$, $\exists f \in \mathsf{span}(\mathcal{F})$ where $\mathop{\mathbb{E}}_{x\sim\mu}\left[(h(x) - f(x))^2\right] \leq \epsilon^2$. Then there is an algorithm that, for any distribution $\mathcal{D}$ over $\mathcal{X} \times \{\pm 1\}$ with marginal μ over $\mathcal{X}$, outputs a function $g : \mathcal{X} \to \{\pm 1\}$ such that $\mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[g(x) \neq b] \leq \inf_{h\in\mathcal{H}} \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[g(x) \neq b] + \epsilon$, with probability at least $11/12$, using at most $\mathrm{poly}(|\mathcal{F}|, 1/\epsilon)$ samples and time.*

Our general learning algorithm will apply to any hypothesis class that has small r-block boundary size, and for which there is a set of functions $\mathcal{F}$ that approximately span the class $\mathcal{H}^{\mathsf{block}}$. This algorithm is improved to work for finite (rather than only continuous) product distributions in Lemma 3.8.17.

Lemma 3.5.2. *Let $\epsilon > 0$ and let $\mathcal{H}$ be a set of measurable functions $f : \mathbb{R}^d \to \{\pm 1\}$ that satisfy:*

1. *There is some $r = r(d, \epsilon)$ such that $\mathsf{bbs}(\mathcal{H}, r) \leq \frac{\epsilon}{3} \cdot r^d$;*
2. *There is a set $\mathcal{F}$ of functions $[r]^d \to \mathbb{R}$ satisfying: $\forall f \in \mathcal{H}, \exists g \in \mathsf{span}(\mathcal{F})$ such that for $v \sim [r]^d, \mathbb{E}\left[(f^{\mathsf{block}}(v) - g(v))^2\right] \leq \epsilon^2/4$.*

Let $n = \mathrm{poly}(|\mathcal{F}|, 1/\epsilon)$ be the sample complexity of the algorithm in Theorem 3.5.1, with error parameter $\epsilon/2$. Then there is an agnostic learning algorithm for $\mathcal{H}$ on continuous product distributions over $\mathbb{R}^d$, that uses $O(\max(n^2, 1/\epsilon^2) \cdot rd^2 \log(dr))$ samples and runs in time polynomial in the sample size.

Proof. We will assume $n > 1/\epsilon$. Let μ be the marginal of $\mathcal{D}$ on $\mathbb{R}^d$. For an r-block partition, let $\mathcal{D}^{\mathsf{block}}$ be the distribution of $(\mathsf{block}(x), b)$ when $(x, b) \sim \mathcal{D}$. We may simulate samples from $\mathcal{D}^{\mathsf{block}}$ by sampling (x, b) from $\mathcal{D}$ and constructing $(\mathsf{block}(x), b)$. The algorithm is as follows:

1. Sample a grid X of length $m = O(rd^2n^2 \log(rd))$ large enough that Lemma 3.2.7 guarantees $\|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} < 1/12n$ with probability $5/6$. Construct $\mathsf{block} : \mathbb{R}^d \to [r]^d$ induced by X. We may construct the block function in time $O(dm \log m)$ by sorting, and once constructed it takes time $O(\log r)$ to compute.

2. Run the algorithm of Theorem 3.5.1 on a sample of n points from $\mathcal{D}^{\mathsf{block}}$ to learn the class $\mathcal{H}^{\mathsf{block}}$; that algorithm returns a function $g : [r]^d \to \{\pm 1\}$. Output $g \circ \mathsf{block}$.

Assume that step 1 succeeds, which occurs with probability at least 5/6. By condition 2, the algorithm in step 2 is guaranteed to work on samples $(v, b) \in [r]^d \times \{\pm 1\}$ where the marginal of v is $\mathsf{unif}([r]^d)$; let $\mathcal{D}^{\mathsf{unif}}$ be the distribution of (v, b) when $v \sim \mathsf{unif}([r]^d)$ and b is obtained by sampling $(x, b) \sim (\mathcal{D} \mid x \in \mathsf{block}^{-1}(v))$. The algorithm of step 2 will succeed on $\mathcal{D}^{\mathsf{unif}}$; we argue that it will also succeed on the actual input $\mathcal{D}^{\mathsf{block}}$ since these distributions are close. Observe that for samples $(v, b) \sim \mathcal{D}^{\mathsf{unif}}$ and $(\mathsf{block}(x), b') \sim \mathcal{D}^{\mathsf{block}}$, if $v = \mathsf{block}(x)$ then b, b' each have the distribution of b' in $(x, b') \sim (\mathcal{D} \mid \mathsf{block}(x) = v)$. Therefore

$$\begin{aligned}\|\mathcal{D}^{\mathsf{unif}} - \mathcal{D}^{\mathsf{block}}\|_{\mathsf{TV}} &= \|(v, b) - (\mathsf{block}(x), b')\|_{\mathsf{TV}} = \|v - \mathsf{block}(x)\|_{\mathsf{TV}} \\ &= \|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} < \frac{1}{12n}\,.\end{aligned}$$

It is a standard fact that for product distributions P^n, Q^n, $\|P^n - Q^n\|_{\mathsf{TV}} \le n \cdot \|P - Q\|_{\mathsf{TV}}$; using this fact,

$$\|(\mathcal{D}^{\mathsf{unif}})^n - (\mathcal{D}^{\mathsf{block}})^n\|_{\mathsf{TV}} \le n \cdot \|\mathcal{D}^{\mathsf{unif}} - \mathcal{D}^{\mathsf{block}}\|_{\mathsf{TV}} < \frac{1}{12}\,.$$

We will argue that step 2 succeeds with probability 5/6; i.e. that with probability 5/6,

$$\mathop{\mathbb{P}}_{(v,b)\sim\mathcal{D}^{\mathsf{block}}}[g(v) \neq b] < \inf_{h\in\mathcal{H}} \mathop{\mathbb{P}}_{(v,b)\sim\mathcal{D}^{\mathsf{block}}}\left[h^{\mathsf{block}}(v) \neq b\right] + \epsilon/2\,.$$

Let $E(S)$ be the event that success occurs given sample $S \in ([r]^d \times \{\pm 1\})^n$. The algorithm samples $S \sim (\mathcal{D}^{\mathsf{block}})^n$ but the success guarantee of step 2 is for $(\mathcal{D}^{\mathsf{unif}})^n$; this step will still succeed with probability 5/6:

$$\begin{aligned}\mathop{\mathbb{P}}_{S\sim(\mathcal{D}^{\mathsf{unif}})^n}[E(S)] &\ge \mathop{\mathbb{P}}_{S\sim(\mathcal{D}^{\mathsf{block}})^n}[E(S)] - \|(\mathcal{D}^{\mathsf{unif}})^n - (\mathcal{D}^{\mathsf{block}})^n\|_{\mathsf{TV}} \\ &> \mathop{\mathbb{P}}_{S\sim\mathcal{D}^n}[E(S)] - \frac{1}{12} \ge \frac{11}{12} - \frac{1}{12} = \frac{5}{6}\,.\end{aligned}$$

Assume that each step succeeds, which occurs with probability at least $1 - 2 \cdot (5/6) = 2/3$. By Proposition 3.2.5, our condition 1, and the fact that $n > 1/\epsilon$, we have for any $h \in \mathcal{H}$ that

$$\mathop{\mathbb{P}}_{x\sim\mu}[h(x) \neq h^{\mathsf{coarse}}(x)] \le r^{-d} \cdot \mathsf{bbs}(\mathcal{H}, r) + \|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} \le \epsilon/3 + \frac{1}{12n} < \epsilon/2\,.$$

The output of the algorithm is $g \circ \mathsf{block}$, which for any $h \in \mathcal{H}$ satisfies:

$$\begin{aligned}
\mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[g(\mathsf{block}(x)) \neq b] &= \mathop{\mathbb{P}}_{(v,b)\sim\mathcal{D}^{\mathsf{block}}}[g(v) \neq b] \leq \mathop{\mathbb{P}}_{(v,b)\sim\mathcal{D}^{\mathsf{block}}}\left[h^{\mathsf{block}}(v) \neq b\right] + \epsilon/2 \\
&= \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[h^{\mathsf{coarse}}(x) \neq b] + \epsilon/2 \\
&\leq \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[h(x) \neq b] + \mathop{\mathbb{P}}_{x}[h(x) \neq h^{\mathsf{coarse}}(x)] + \epsilon/2 \\
&< \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[h(x) \neq b] + \epsilon\,.
\end{aligned}$$

Then $\mathbb{P}[g(\mathsf{block}(x)) \neq b] \leq \inf_{h\in\mathcal{H}} h(x) \neq b + \epsilon$, as desired. □

3.5.2 Fourier Analysis on $[n]^d$

We will show how to construct a spanning set $\mathcal{F}$ to satisfy condition 2 of the general learning algorithm, by using noise sensitivity and the Walsh basis. For any n, let $u \sim [n]^d$ uniformly at random and draw $v \in [n]^d$ as follows: $v_i = u_i$ with probability δ, and v_i is uniform in $[n] \setminus \{u_i\}$ with probability $1-\delta$. The noise sensitivity of functions $[n]^d \to \{\pm 1\}$ is defined as:

$$\mathsf{ns}_{n,\delta}(f) := \mathop{\mathbb{P}}_{u,v}[f(u) \neq f(v)] = \frac{1}{2} - \frac{1}{2} \cdot \mathop{\mathbb{E}}_{u,v}[f(u)f(v)]\,.$$

Note that we include n in the subscript to indicate the size of the domain. We will use $\mathsf{ns}_{r,\delta}(f)$ to obtain upper bounds on the spanning set, and we will obtain bounds on $\mathsf{ns}_{r,\delta}$ by relating it to $\mathsf{ns}_{2,\delta}$, for which many bounds are known. For a function $f : [n]^d \to \{\pm 1\}$, two vectors $u, v \in [r]^d$, and $x \in \{\pm 1\}^d$, define $[u,v]^x \in [n]^d$ as the vector with $[u,v]^x_i = u_i$ if $x_i = 1$ and v_i if $x_i = -1$. Then define $f_{u,v} : \{\pm 1\}^d \to \{\pm 1\}$ as the function $f_{u,v}(x) = f([u,v]^x)$. The next lemma is essentially the same as the reduction in [BOW10].

Lemma 3.5.3. *Let $\mathcal{H}$ be a set of functions $f : \mathbb{R}^d \to \{\pm 1\}$ such that for any linear transformation $A \in \mathbb{R}^{d\times d}$, the function $f \circ A \in \mathcal{H}$, and let $\mathsf{block} : \mathbb{R}^d \to [r]^d$ be any r-block partition. Let $\mathsf{ns}_{2,\delta}(\mathcal{H}) = \sup_{f\in\mathcal{H}} \mathsf{ns}_{2,\delta}(f)$ where $\mathsf{ns}_{2,\delta}(f)$ is the δ-noise sensitivity on domain $\{\pm 1\}^d$. Then $\mathsf{ns}_{r,\delta}(f^{\mathsf{block}}) \leq \mathsf{ns}_{2,\delta}(\mathcal{H})$.*

Proof. Let $u \sim [r]^d$ and let v be uniform among the vectors $[r]^d$ where $\forall i, v_i \neq u_i$. Now let $x \sim \{\pm 1\}^d$ uniformly at random and let y be drawn such that $y_i = x_i$ with probability δ and $y_i = -x_i$ otherwise. Then $[u,v]^x$ is uniform in $[r]^d$, because $[u,v]^x_i$ is u_i or v_i with equal probability and the marginals of u_i, v_i are uniform. $[u,v]^y_i = [u,v]^x_i$ with probability $1-\delta$ and is otherwise uniform in $[r] \setminus \{[u,v]^x_i\}$. Let $f : [r]^d \to \{\pm 1\}$ and $\delta \in [0,1]$. Let (u', v')

be an independent copy of (u, v) and observe that $\mathsf{ns}_{r,\delta}(f^{\mathsf{block}}) = \mathbb{P}\left[f^{\mathsf{block}}(u') \neq f^{\mathsf{block}}(v')\right]$. Now observe that $([u,v]^x, [u,v]^y)$ has the same distribution as (u', v'), so:

$$\begin{aligned}\mathop{\mathbb{E}}_{u,v}\left[\mathsf{ns}_{2,\delta}(f_{u,v})\right] &= \mathop{\mathbb{E}}_{u,v}\left[\mathop{\mathbb{P}}_{x,y\sim_\delta x}\left[f([u,v]^x) \neq f([u,v]^y)\right]\right] \\ &= \mathop{\mathbb{E}}_{u,v,(x,y)_\delta}\left[\mathbb{1}\left[f([u,v]^x) \neq f([u,v]^y)\right]\right] \\ &= \mathop{\mathbb{E}}_{u',v'}\left[\mathbb{1}\left[f(u') \neq f(v')\right]\right] = \mathsf{ns}_{r,\delta}(f^{\mathsf{block}})\,.\end{aligned}$$

For any $u, v \in [r]^d$, define the function $\Phi_{u,v} : \{\pm 1\}^d \to [r]^d$ by $\Phi_{u,v}(x) = \mathsf{blockpoint}([u,v]^x)$. This function maps $\{\pm 1\}^d$ to a set $\{b_{1,i_1}, b_{1,j_1}\} \times \cdots \times \{b_{d,i_d}, b_{d,j_d}\}$ and can be obtained by translation and scaling, which is a linear transformation. Therefore $f_{u,v} = f \circ \Phi_{u,v}^{-1}$, so we are guaranteed that $f_{u,v} \in \mathcal{H}$. So

$$\mathsf{ns}_{r,\delta}(f) = \mathop{\mathbb{E}}_{u,v}\left[\mathsf{ns}_{2,\delta}(f_{u,v})\right] \leq \mathsf{ns}_{2,\delta}(\mathcal{H})\,. \qquad \square$$

We define the Walsh basis, an orthonormal basis of functions $[n]^d \to \mathbb{R}$; see e.g. [BRY14]. Suppose $n = 2^m$ for some positive integer m. For two functions $f, g : [n]^d \to \mathbb{R}$, define the inner product $\langle f, g\rangle = \mathbb{E}_{x\sim[n]^d}[f(x)g(x)]$. The Walsh functions $\{\psi_0, \ldots, \psi_m\}, \psi_i : [n] \to \{\pm 1\}$ can be defined by $\psi_0 \equiv 1$ and for $i \geq 1$, $\psi_i(z) := (-1)^{\mathsf{bit}_i(z-1)}$ where $\mathsf{bit}_i(z-1)$ is the i^{th} bit in the binary representation of $z-1$, where the first bit is the least significant (see e.g. [BRY14]). It is easy to verify that for all $i, j \in \{0, \ldots, m\}$, if $i \neq j$ then $\langle \psi_i, \psi_j\rangle = 0$, and $\mathbb{E}_{x\sim[n]}[\psi_i(x)] = 0$ when $i \geq 1$. For $S \subseteq [m]$ define $\psi_S = \prod_{i\in S} \psi_i$ and note that for any set $S \subseteq [m], S \neq \emptyset$,

$$\mathop{\mathbb{E}}_{x\sim[n]}\left[\psi_S(x)\right] = \mathop{\mathbb{E}}_{x\sim[n]}\left[\prod_{i\in S}\psi_i(x)\right] = \mathop{\mathbb{E}}_{x\sim[n]}\left[(-1)^{\sum_{i\in S}\mathsf{bit}_i(x-1)}\right] = 0 \tag{3.1}$$

since each bit is uniform in $\{0, 1\}$, while $\psi_\emptyset \equiv 1$. For $S, T \subseteq [m]$,

$$\langle \psi_S, \psi_T\rangle = \mathbb{E}_{x\sim[n]}[\psi_S(x)\psi_T(x)] = \mathbb{E}_x[\psi_{S\Delta T}(x)]\,,$$

where $S\Delta T$ is the symmetric difference, so this is 0 when $S\Delta T \neq \emptyset$ (i.e. $S \neq T$) and 1 otherwise; therefore $\{\psi_S : S \subseteq [m]\}$ is an orthonormal basis of functions $[n] \to \mathbb{R}$. Identify each $S \subseteq [m]$ with the number $s \in \{0, \ldots, n-1\}$ where $\mathsf{bit}_i(s) = \mathbb{1}[i \in S]$. Now for every $\alpha \in \{0, \ldots, n-1\}^d$ define $\psi_\alpha : [n]^d \to \{\pm 1\}$ as $\psi_\alpha(x) = \prod_{i=1}^d \psi_{\alpha_i}(x_i)$ where ψ_{α_i} is the Walsh function determined by the identity between subsets of $[m]$ and the integer $\alpha_i \in \{0, \ldots, n-1\}$. It is easy to verify that the set $\{\psi_\alpha : \alpha \in \{0, \ldots, n-1\}^d\}$ is an orthonormal

basis. Every function $f : [n]^d \to \mathbb{R}$ has a unique representation $f = \sum_{\alpha \in \{0,\ldots,n-1\}^d} \hat{f}(\alpha)\psi_\alpha$ where $\hat{f}(\alpha) = \langle f, \psi_\alpha \rangle$.

For each $x \in [n]^d$ and $\rho \in [0,1]$ define $N_\rho(x)$ as the distribution over $y \in [n]^d$ where for each $i \in [d]$, $y_i = x_i$ with probability ρ and y_i is uniform in $[n]$ with probability $1-\rho$. Define $T_\rho f(x) := \mathop{\mathbb{E}}_{y \sim N_\rho(x)}[f(y)]$ and $\mathsf{stab}_\rho(f) := \langle f, T_\rho f \rangle$. For any $\alpha \in \{0,\ldots,n-1\}^d$,

$$\begin{aligned} T_\rho \psi_\alpha(x) &= \mathop{\mathbb{E}}_{y \sim N_\rho(x)}[\psi_\alpha(y)] = \mathop{\mathbb{E}}_{y \sim N_\rho(x)}\left[\prod_{i=1}^d \psi_{\alpha_i}(y_i)\right] = \prod_{i=1}^d \mathop{\mathbb{E}}_{y_i \sim N_\rho(x_i)}[\psi_{\alpha_i}(y_i)] \\ &= \prod_{i=1}^d \left[\rho\psi_{\alpha_i}(x_i) + (1-\rho)\mathop{\mathbb{E}}_{z\sim[n]}[\psi_{\alpha_i}(z)]\right]. \end{aligned}$$

If $\alpha_i \geq 1$ then $\mathop{\mathbb{E}}_{z\sim[n]}[\psi_{\alpha_i}(z)] = 0$; otherwise, $\psi_1 \equiv 1$ so $\mathop{\mathbb{E}}_{y_i \sim N_\rho(x_i)}[\psi_0(y_i)] = 1$. Therefore

$$T_\rho \psi_\alpha(x) = \rho^{|\alpha|}\psi_\alpha(x)\,,$$

where $|\alpha|$ is the number of nonzero entries of α; so $\widehat{T_\rho f}(\alpha) = \langle \psi_\alpha, T_\rho f\rangle = \langle T_\rho \psi_\alpha, f\rangle = \rho^{|\alpha|}\widehat{f}(\alpha)$. Since T_ρ is a linear operator,

$$\mathsf{stab}_\rho(f) = \langle f, T_\rho f\rangle = \sum_\alpha \rho^{|\alpha|}\hat{f}(\alpha)^2\,.$$

Note that for $f : \{\pm 1\}^d \to \{\pm 1\}$, $\mathsf{stab}_\rho(f)$ is the usual notion of stability in the analysis of Boolean functions.

Proposition 3.5.4. *For any $f : [n]^d \to \{\pm 1\}$ and any $\delta, \rho \in [0,1]$, $\mathsf{ns}_{n,\delta}(f) = \frac{1}{2} - \frac{1}{2}\cdot \mathsf{stab}_{1-\frac{n}{n-1}\delta}(f)$.*

Proof. For $v \sim N_\rho(u), v_i = u_i$ with probability $\rho + \frac{1-\rho}{n}$, so in the definition of noise sensitivity, v is distributed as $N_\rho(u)$ where $(1-\delta) = \rho + \frac{1-\rho}{n}$, i.e. $\delta = 1 - \rho - \frac{1-\rho}{n} = (1-1/n) - \rho(1-1/n) = (1-\rho)(1-1/n)$; or, $\rho = 1 - \frac{n}{n-1}\delta$. By rearranging, we arrive at the conclusions. □

Proposition 3.5.5. *For any $f : [n]^d \to \mathbb{R}$ and $t = \frac{2}{\delta}$, $\sum_{\alpha:|\alpha|\geq t}\hat{f}(\alpha)^2 \leq 2.32 \cdot \mathsf{ns}_{n,\delta}(f)$.*

Proof. Following [KOS04]:

$$\begin{aligned}2\mathsf{ns}_{n,\delta}(f) &= 1 - \sum_{\alpha}\left(1 - \frac{n}{n-1}\delta\right)^{|\alpha|}\hat{f}(\alpha)^2 \geq 1 - \sum_{\alpha}(1-\delta)^{|\alpha|}\hat{f}(\alpha)^2 \\ &= \sum_{\alpha}(1-(1-\delta)^{|\alpha|})\hat{f}(\alpha)^2 \geq \sum_{\alpha:|\alpha|\geq 2/\delta}(1-(1-\delta)^{|\alpha|})\hat{f}(\alpha)^2 \\ &\geq \sum_{\alpha:|\alpha|\geq 2/\delta}(1-(1-\delta)^{2/\delta})\hat{f}(\alpha)^2 \geq (1-e^{-2})\sum_{\alpha:|\alpha|\geq 2/\delta}\hat{f}(\alpha)^2\,.\end{aligned}$$

The result now holds since $2/(1-e^{-2}) < 2.32$. □

Lemma 3.5.6. *Let $\mathcal{H}$ be a set of functions $[n]^d \to \{\pm 1\}$ where n is a power of 2, let $\epsilon, \delta > 0$ such that $\forall h \in \mathcal{H}, \mathsf{ns}_{n,\delta}(h) \leq \epsilon^2/3$, and let $t = \lceil \frac{2}{\delta} \rceil$. Then there is a set $\mathcal{F}$ of functions $[n]^d \to \mathbb{R}$ of size $|\mathcal{F}| \leq (nd)^t$, such that that for any $h \in \mathcal{H}$, there is a function $p \in \mathsf{span}(\mathcal{F})$ where $\mathbb{E}\left[(h(x)-p(x))^2\right] \leq \epsilon^2$.*

Proof. Let $p = \sum_{|\alpha|<t}\hat{f}(\alpha)\phi_\alpha$. Then by Proposition 3.5.5,

$$\begin{aligned}\mathbb{E}\left[(p(x)-f(x))^2\right] &= \mathbb{E}\left[\left(\sum_{|\alpha|\geq t}\hat{f}(\alpha)\phi_\alpha(x)\right)^2\right] = \mathbb{E}\left[\sum_{|\alpha|\geq t}\sum_{|\beta|\geq t}\hat{f}(\alpha)\hat{f}(\beta)\phi_\alpha(x)\phi_\alpha(x)\right] \\ &= \sum_{|\alpha|\geq t}\sum_{|\beta|\geq t}\hat{f}(\alpha)\hat{f}(\beta)\langle\phi_\alpha,\phi_\beta\rangle = \sum_{|\alpha|\geq t}\hat{f}(\alpha)^2 \leq \epsilon^2\,.\end{aligned}$$

Therefore p is a linear combination of functions $\phi_\alpha = \prod_{i=1}^d \phi_{\alpha_i}$ where at most t values $\alpha_i \in \{0,\ldots,n-1\}$ are not 0. There are at most $((n-1)d)^t$ such products since for each non-constant ϕ_{α_i} we choose $i \in [d]$ and $\alpha_i \in [n-1]$. We may take $\mathcal{F}$ to be the set of these products. □

3.5.3 Learning Functions of Halfspaces

To apply Lemma 3.5.2, we must give bounds on $\mathsf{bbs}(\mathcal{B}_k \circ \mathcal{H}, r)$ and the noise sensitivity:

Lemma 3.5.7. *Fix any r. Then $\mathsf{bbs}(\mathcal{B}_k \circ \mathcal{H}, r) \leq dkr^{d-1}$.*

Proof. Any halfspace $h(x) = \mathrm{sign}(\langle w, x\rangle - t)$ is unate, meaning there is a vector $\sigma \in \{\pm 1\}^d$ such that the function $h^\sigma := \mathrm{sign}(\langle w, x^\sigma\rangle - t)$, where $x^\sigma = (\sigma_1 x_1, \ldots, \sigma_d x_d)$, is monotone.

For any r-block partition $\mathsf{block} : \mathbb{R}^d \to [r]^d$ defined by values $\{a_{i,j}\}$ for $i \in [d], j \in [r-1]$, we can define $\mathsf{block}^\sigma : \mathbb{R}^d \to [r]^d$ as the block partition obtained by taking $\{\sigma_i \cdot a_{i,j}\}$. The number of non-constant blocks of h in block is the same as that of h^σ in block^σ, but h^σ is monotone. Thus the bound on bbs for monotone functions holds, so $\mathsf{bbs}(\mathcal{H}, r) \le dr^{d-1}$ by Lemma 3.7.3, and Proposition 3.4.5 gives $\mathsf{bbs}(\mathcal{B}_k \circ \mathcal{H}, r) \le dkr^{d-1}$. □

The bounds on noise sensitivity follow from known results for the hypercube.

Proposition 3.5.8. *Let $h_1, \dots, h_k : [n]^d \to \{\pm 1\}$ and let $g : \{\pm 1\}^k \to \{\pm 1\}$. Let $f := g \circ (h_1, \dots, h_k)$. Then $\mathsf{ns}_\delta(f) \le \sum_{i=1}^k \mathsf{ns}_\delta(h_i)$.*

Proof. For u, v drawn from $[n]^d$ as in the definition of noise sensitivity, the union bound gives $\mathsf{ns}_\delta(f) = \mathop{\mathbb{P}}_{u,v}[f(u) \ne f(v)] \le \mathop{\mathbb{P}}_{u,v}[\exists i : h_i(u) \ne h_i(v)] \le \sum_{i=1}^k \mathsf{ns}_\delta(h_i)$. □

Lemma 3.5.9. *Let $f = g \circ (h_1, \dots, h_k) \in \mathcal{B}_k \circ \mathcal{H}$. For any r-block partition $\mathsf{block} : \mathbb{R}^d \to [r]^d$, and any $\delta \in [0,1], \mathsf{ns}_{r,\delta}(f^{\mathsf{block}}) = O(k\sqrt{\delta})$.*

Proof. It is known that $\mathsf{ns}_{2,\delta}(\mathcal{H}) = O(\sqrt{\delta})$ (Peres' theorem [O'D14]). Let A be any full-rank linear transformation and let $h \in \mathcal{H}$, $h \circ A \in \mathcal{H}$. This holds since for some $w \in \mathbb{R}^d, t \in \mathbb{R}$, $h(Ax) = \mathrm{sign}(\langle w, Ax\rangle - t) = \mathrm{sign}(\langle Aw, x\rangle - t)$, which is a halfspace. Then Lemma 3.5.3 implies $\mathsf{ns}_{r,\delta}(h^{\mathsf{block}}) \le \mathsf{ns}_{2,\delta}(\mathcal{H}) = O(\sqrt{\delta})$ and we conclude with Proposition 3.5.8. □

Theorem 1.2.12. *There is a distribution-free, improper agnostic learning algorithm for $\mathcal{B}_k \circ \mathcal{H}$ under (continuous or finite) product distributions over $\mathbb{R}^d$, with time complexity*

$$\min\left\{\left(\frac{dk}{\epsilon}\right)^{O\left(\frac{k^2}{\epsilon^4}\right)}, O\left(\frac{1}{\epsilon^2}\left(\frac{3dk}{\epsilon}\right)^d\right)\right\}.$$

Proof. Here we prove only the continuous distribution case. The finite case is proved in Theorem 3.8.18.

For $r = \lceil dk/\epsilon \rceil$, we have $r^{-d} \cdot \mathsf{bbs}(\mathcal{B}_k \circ \mathcal{H}, r) \le \epsilon$ by Lemma 3.5.7, so condition 1 of Lemma 3.5.2 holds. Lemma 3.5.9 guarantees that for any $f \in \mathcal{B}_k \circ \mathcal{H}, \mathsf{ns}_{r,\delta}(f^{\mathsf{block}}) = O(k\sqrt{\delta})$. Setting $\delta = \Theta(\epsilon^4/k^2)$ so that $\mathsf{ns}_{r,\delta}(f^{\mathsf{block}}) \le \epsilon^2/3$, we obtain via Lemma 3.5.6 a set $\mathcal{F}$ of size $|\mathcal{F}| \le (rd)^{O(k^2/\epsilon^4)}$ satisfying condition 2 of Lemma 3.5.2. Then for $n = \mathrm{poly}(|\mathcal{F}|, 1/\epsilon)$ we apply Lemma 3.5.2 to get an algorithm with sample complexity

$$O\left(rd^2n^2\log(rd)\right) = O\left(\frac{d^3k}{\epsilon}\log(dk/\epsilon)\right) \cdot \left(\frac{dk}{\epsilon}\right)^{O\left(\frac{k^2}{\epsilon^4}\right)}.$$

The other time complexity follows from Lemma 3.4.8. □

3.6 Learning Polynomial Threshold Functions

A function $f : \mathbb{R}^d \to \{\pm 1\}$ is a degree-k PTF if there is a degree-k polynomial $p : \mathbb{R}^d \to \mathbb{R}$ such that $f(x) = \text{sign}(p(x))$. Write $\mathcal{P}_k$ for the class of degree-k PTFs. Degree-k PTFs in $\mathbb{R}^d$ can be PAC learned in time $d^{O(k)}$ using linear programming [KOS04], but agnostic learning is more challenging. Diakonikolas *et al.* [DHK+10] previously gave an agnostic learning algorithm for degree-k PTFs in the uniform distribution over $\{\pm 1\}^d$ with time complexity $d^{\psi(k,\epsilon)}$, where

$$\psi(k, \epsilon) := \min \left\{ O(\epsilon^{-2^{k+1}}), 2^{O(k^2)} \left(\log(1/\epsilon)/\epsilon^2\right)^{4k+2} \right\} .$$

The main result of that paper is an upper bound on the noise sensitivity of PTFs. Combined with the reduction of [BOW10], this implies an algorithm for the uniform distribution over $[n]^d$ with complexity $(dn)^{\psi(k,\epsilon)}$ and for the Gaussian distribution with complexity $d^{\psi(k,\epsilon)}$. Our agnostic learning algorithm for degree-k PTFs eliminates the dependence on n and works for any unknown product distribution over $\mathbb{R}^n$, while matching the complexity of [DHK+10] for the uniform distribution over the hypercube. We prove this theorem in the remainder of the section.

Theorem 1.2.13. *There is an improper agnostic learning algorithm for degree-k PTFs under (finite or continuous) product distributions over $\mathbb{R}^d$, with time complexity*

$$\min \left\{ \left(\frac{kd}{\epsilon}\right)^{\psi(k,\epsilon)} , O\left(\frac{1}{\epsilon^2}\left(\frac{9dk}{\epsilon}\right)^d\right) \right\} .$$

As for halfspaces, we will give bounds on the noise sensitivity and block boundary size and apply the general learning algorithm. The bound on noise sensitivity will follow from known results on the hypercube [DHK+10] and a trick from [BOW10], but the bound on the block boundary size is much more difficult to obtain than for halfspaces.

3.6.1 Block-Boundary Size of PTFs

A theorem of Warren [War68] gives a bound on the number of connected components of $\mathbb{R}^d$ after removing the 0-set of a degree-k polynomial. This bound (Theorem 3.6.7 below) will be our main tool.

A set $S \subseteq \mathbb{R}^d$ is *connected*[1] if for every $s, t \in S$ there is a continuous function $p : [0, 1] \to S$ such that $p(0) = s, p(1) = t$. A subset $S \subseteq X$ where $X \subseteq \mathbb{R}^d$ is a *connected*

[1]Here we are using the fact that *connected* and *path-connected* are equivalent in $\mathbb{R}^d$.

component of X if it is connected and there is no connected set $T \subseteq X$ such that $S \subseteq T$. Write $\mathsf{comp}(X)$ for the number of connected components of X.

A function $\rho : \mathbb{R}^d \to (\mathbb{R} \cup \{*\})^d$ is called a *restriction* and we will denote $|\rho| = |\{i \in [d] : \rho(i) = *\}|$. The affine subspace A_ρ induced by ρ is $A_\rho := \{x \in \mathbb{R}^d \mid x_i = \rho(i) \text{ if } \rho(i) \neq *\}$ and has affine dimension $|\rho|$.

For $n \leq d$, let $\mathcal{A}_n$ be the set of affine subspaces A_ρ obtained by choosing a restriction ρ with $\rho(i) = *$ when $i \leq n$ and $\rho(i) \neq *$ when $i > n$, so in particular $\mathcal{A}_d = \{\mathbb{R}^d\}$.

Let $f : \mathbb{R}^d \to \{\pm 1\}$ be a measurable function and define the boundary of f as:

$$\partial f := \{x \in \mathbb{R}^d \mid \forall \epsilon > 0, \exists y : \|x - y\|_2 < \epsilon, f(y) \neq f(x)\}.$$

This is equivalent to the boundary of the set of $+1$-valued points, and the boundary of any set is closed. Each measurable $f : \mathbb{R}^d \to \{\pm 1\}$ induces a partition of $\mathbb{R}^d \setminus \partial f$ into some number of connected parts. For a set $\mathcal{H}$ of functions $f : \mathbb{R}^d \to \{\pm 1\}$ and $n \leq d$, write

$$M(n) := \max_{f \in \mathcal{H}} \max_{A \in \mathcal{A}_n} \mathsf{comp}(A \setminus \partial f).$$

For each $i \in [d]$ let $\mathcal{P}_i$ be the set of hyperplanes of the form $\{x \in \mathbb{R}^d \mid x_i = a\}$ for some $a \in \mathbb{R}$. An (r, n, m)-*arrangement* for f is any set $A \setminus \left(\partial f \cup \bigcup_{i=1}^{m} \bigcup_{j=1}^{r-1} H_{i,j}\right)$ where $A \in \mathcal{A}_n$ and $H_{i,j} \in \mathcal{P}_i$ such that all $H_{i,j}$ are distinct. Write $R_f(r, n, m)$ for the set of (r, n, m)-arrangements for f. Define

$$P_r(n, m) := \max_{f \in \mathcal{H}} \max\{\mathsf{comp}(R) \mid R \in R_f(r, n, m)\}$$

and observe that $P_r(n, 0) = M(n)$.

Proposition 3.6.1. *For any set $\mathcal{H}$ of functions $f : \mathbb{R}^d \to \{\pm 1\}$ and any $r > 0$, $\mathsf{bbs}(\mathcal{H}, r) \leq P_r(d, d) - r^d$.*

Proof. Consider any r-block partition, which is obtained by choosing values $a_{i,j} \in \mathbb{R}$ for each $i \in [d], j \in [r-1]$ and defining $\mathsf{block} : \mathbb{R}^d \to [r]^d$ by assigning each $x \in \mathbb{R}^d$ the block $v \in [r]^d$ where v_i is the unique value such that $a_{i,v_i-1} < x_i \leq a_{i,v_i}$, where we define $a_{i,0} = -\infty, a_{i,r} = \infty$. Suppose v is a non-constant block, so there are $x, y \in \mathsf{block}^{-1}(v) \setminus \partial f$ such that $f(x) \neq f(y)$. Let $H_{i,j} = \{x \in \mathbb{R}^d \mid x_i = a_{i,j}\}$ and let $B = \partial f \cup \bigcup_{i,j} H_{i,j}$. Consider the set $\mathbb{R}^d \setminus B$. Since $x \notin \partial f$ there exists some small open ball R_x around x such that $\forall x' \in R_x, f(x') = f(x)$; and since $x \in \mathsf{block}^{-1}(v)$, $R_x \cap \mathsf{block}^{-1}(v)$ is a set of positive Lebesgue measure. Since B has Lebesgue measure 0, we conclude that $(R_x \cap \mathsf{block}^{-1}(v)) \setminus B$

has positive measure, so there is $x' \in \mathsf{block}^{-1}(v) \setminus B$ with $f(x') = f(x)$. Likewise, there is $y' \in \mathsf{block}^{-1}(v) \setminus B$ with $f(y') = f(y) \neq f(x')$. Therefore x', y' must belong to separate components, so $\mathsf{block}^{-1}(v) \setminus B$ is partitioned into at least 2 components. Meanwhile, each constant block is partitioned into at least 1 component. So

$$P_r(d,d) \geq 2 \cdot (\#\text{ non-constant blocks}) + (\#\text{ constant blocks}) = \mathsf{bbs}(\mathcal{H}, r) + r^d \,. \qquad \square$$

The following fact must be well-known, but not to us:

Proposition 3.6.2. *Let A be an affine subspace of $\mathbb{R}^d$, let $B \subset A$, and for $a \in \mathbb{R}$ let $H = \{x \in \mathbb{R}^d \mid x_1 = a\}$. Then*

$$\mathsf{comp}(A \setminus (H \cup B)) - \mathsf{comp}(A \setminus B) \leq \mathsf{comp}(H \setminus B) \,.$$

Proof. Let G be the graph with its vertices V being the components of $A \setminus (H \cup B)$ and the edges E being the pairs (S,T) where S, T are components of $A \setminus (H \cup B)$ such that $\forall s \in S, s_1 < a$, $\forall t \in T, t_1 > a$, and there exists a component U of $A \setminus B$ such that $S, T \subset U$. Clearly $\mathsf{comp}(A \setminus (H \cup B)) = |V|$; we will show that $\mathsf{comp}(A \setminus B)$ is the number of connected components of G and that $|E| \leq \mathsf{comp}(H \setminus B)$. This suffices to prove the statement. We will use the following claim:

Claim 3.6.3. *Let U be a connected component of $A \setminus B$. If $S, T \in V$ and there is a path $p : [0,1] \to U$ such that $p(0) \in S, p(1) \in T$ and either $\forall \lambda, p(\lambda)_1 \leq a$ or $\forall \lambda, p(\lambda)_1 \geq a$, then $S = T$.*

Proof of claim. Assume without loss of generality that $p(\lambda)_1 \leq a$ for all λ. Let $P = \{p(\lambda) \mid \lambda \in [0,1]\}$. Since U is open we can define for each λ a ball $B(\lambda) \ni p(\lambda)$ such that $B(\lambda) \subset U$. Consider the sets $B_a(\lambda) := \{x \in B(\lambda) \mid x_1 < a\}$, which are open, and note that for all $\alpha, \beta \in [0,1]$, if $p(\alpha) \in B(\beta)$ then $B_a(\alpha) \cap B_a(\beta) \neq \emptyset$ since $p(\alpha)_1, p(\beta)_1 \leq a$.

Assume for contradiction that there is λ such that $B_a(\lambda)$ is not connected to S or T; then let λ' be the infimum of all such λ, which must satisfy $\lambda' > 0$ since $p(0) \in S$. For any α, if $p(\alpha) \in B(\lambda')$ and $B_a(\alpha)$ is connected to S or T then since $B_a(\lambda') \cap B_a(\alpha)$ it must be that $B_a(\lambda)$ is connected as well; therefore $B_a(\alpha)$ is not connected to either S or T. But since p is continuous, there is $\alpha < \lambda'$ such that $p(\alpha) \in B(\lambda')$, so λ' cannot be the infimum, which is a contradiction. Therefore every λ has $B_a(\lambda)$ connected to either S or T. If $S \neq T$, this is a contradiction since there must then be α, β such that $p(\alpha) \in B(\beta)$ but $B_a(\alpha), B_a(\beta)$ are connected to S, T respectively. Therefore $S = T$. $\square$

We first show that $\mathsf{comp}(A \setminus B)$ is the number of graph-connected components of G. Suppose that vertices (S, T) are connected, so there is a path $S = S_0, \ldots, S_n = T$ in G. Then there are connected components U_i of $A \setminus B$ such that $S_{i-1}, S_i \subset U_i$; so $S_i \subset U_i \cap U_{i+1}$, which implies that $\bigcup_i U_i \subset A \setminus B$ is connected. Therefore we may define Φ as mapping each connected component $\{S_i\}$ of G to the unique component U of $A \setminus B$ with $\bigcup_i S_i \subset U$. Φ is surjective since for each component U of $A \setminus B$ there is some vertex S (a component of $A \setminus (H \cup B)$) such that $S \subseteq U$: this is U itself if $U \cap H = \emptyset$. For some connected component U of $A \setminus B$, let $S, T \subseteq U$ be vertices of G, and let $s \in S, t \in T$; since U is connected, there is a path $p : [0,1] \to U$ such that $s = p(0), t = p(1)$. Let $S = S_0, \ldots, S_n = T$ be the multiset of vertices such that $\forall \lambda \in [0,1], \exists i : p(\lambda) \in S_i$; let $\psi(\lambda) \in \{0, \ldots, n\}$ be the index such that $p(\lambda) \in S_{\psi(\lambda)}$, and order the sequence such that if $\alpha < \beta$ then $\psi(\alpha) \leq \psi(\beta)$ (note that we may have $S_i = S_j$ for some $i < j$ if $p(\lambda)$ visits the same set more than once). Then for any i, $S_i, S_{i+1} \subseteq U$ since the path visits both and is contained in U. If S_i, S_{i+1} are on opposite sides of H, then there is an edge (S_i, S_{i+1}) in G; otherwise, the above claim implies $S_i = S_{i+1}$. Thus there is a path S to T in G; this proves that Φ is injective, so $\mathsf{comp}(A \setminus B)$ is indeed the number of graph-connected components of G.

Now let $(S, T) \in E$, so there is a component U of $A \setminus B$ such that $S, T \subset U$. For any $s \in S, t \in T$ there is a continuous path $p_{s,t} : [0,1] \to U$ where $p_{s,t}(0) = s, p_{s,t}(1) = t$. There must be some $z \in [0,1]$ such that $p_{s,t}(z) \in H$, otherwise the path is a path in $\mathbb{R}^d \setminus B$ and $S = T$. Since $p_{s,t}(z) \in H \cap U$, so $p_{s,t}(z) \notin B$, there is some component $Z \in \mathcal{C}_H$ containing $p_{s,t}(z)$. We will map the edge (S, T) to an arbitrary such Z, for any choice s, t, z, and show that it is injective. Suppose that $(S, T), (S', T')$ map to the same $Z \in \mathcal{C}_H$. Without loss of generality we may assume that S, S' lie on the same side of H and that $\forall x \in S \cup S', x_1 < a$. Then there are $s \in S, s' \in S', t \in T, t' \in T'$, and $z, z' \in [0,1]$ such that $p_{s,t}(z), p_{s',t'}(z') \in Z$. Then since Z is a connected component, we may take z, z' to be the least such values that $p_{s,t}(z), p_{s',t'}(z) \in Z$, and connected $p_{s,t}(z), p_{s',t'}(z)$ by a path in Z to obtain a path $q : [0,1] \to U$ such that $q(0) = s, q(1) = s'$, and $\forall \lambda, q(\lambda)_1 \leq a$. Then by the above claim, $S = S'$; the same holds for T, T', so the mapping is injective. This completes the proof of the proposition. □

Proposition 3.6.4. *For any set $\mathcal{H}$ of measurable functions $f : \mathbb{R}^d \to \{\pm 1\}$ and any $r > 1$,*

$$P_r(n, m) \leq P_r(n, m-1) + (r-1) \cdot P_r(n-1, m-1)\,.$$

Proof. Let $A \in \mathcal{A}_n$ and $a_{i,j} \in \mathbb{R}$, $i \in [m], j \in [r-1]$ such that the number of connected components in $A \setminus B$, where $B = \partial f \cup \bigcup_{i,j} H_{i,j}$ and $H_{i,j} = \{x \in A \mid x_i = a_{i,j}\}$, is $P_r(n, m)$.

For $0 \le k \le r-1$ let

$$B_k := \partial f \cup \left(\bigcup_{i=1}^{m-1} \bigcup_{j=1}^{r-1} H_{i,j} \right) \cup \left(\bigcup_{j=1}^{k} H_{m,j} \right) ,$$

so that $B = B_{r-1}$ and $B_k = B_{k-1} \cup H_{m,k}$. Since B_0 is an $(r, n, m-1)$-arrangement, $\mathsf{comp}(A \setminus B_0) \le P_r(n, m-1)$. For $k > 0$, since B_k is obtained from B_{k-1} by adding a hyperplane $H_{m,k}$, Proposition 3.6.2 implies

$$\mathsf{comp}(A \setminus B_k) \le \mathsf{comp}(A \setminus B_{k-1}) + \mathsf{comp}(H \setminus B_{k-1}) \le \mathsf{comp}(A \setminus B_{k-1}) + P_r(n-1, m-1) ,$$

because $H \setminus B_{k-1}$ is an $(r, n-1, m-1)$-arrangement. Iterating $r-1$ times, once for each added hyperplane, we arrive at

$$\begin{aligned} P_r(n,m) &= \mathsf{comp}(A \setminus B) \\ &= \mathsf{comp}(A \setminus B_0) + \sum_{k=1}^{r-1} (\mathsf{comp}(A \setminus B_k) - \mathsf{comp}(A \setminus B_{k-1})) \\ &\le P_r(n, m-1) + (r-1) P_r(n-1, m-1) . \end{aligned}$$

□

Lemma 3.6.5. *For any set $\mathcal{H}$ of measurable functions $\mathbb{R}^d \to \{\pm 1\}$ and any r,*

$$P_r(d,d) \le (r-1)^d + \sum_{i=0}^{d-1} \binom{d}{i} \cdot M(d-i) \cdot (r-1)^i .$$

Proof. Write $s = r-1$ for convenience. We will show by induction the more general statement that for any $m \le n \le d$,

$$P_r(n,m) \le \sum_{i=0}^{m} \binom{m}{i} \cdot M(n-i) \cdot s^i$$

where we define $M(0) := 1$. In the base case, note that $P_r(n,0) = M(n)$. Assume the

statement holds for $P_r(n', m')$ when $n' \le n, m' < m$. Then by Proposition 3.6.4,

$$\begin{aligned}
P_r(n,m) &\le P_r(n,m-1) + s \cdot P_r(n-1,m-1) \\
&\le \sum_{i=0}^{m-1} \binom{m-1}{i} \cdot M(n-i) \cdot s^i + \sum_{i=0}^{m-1} \binom{m-1}{i} \cdot M(n-1-i) \cdot s^{i+1} \\
&\le \sum_{i=0}^{m-1} \binom{m-1}{i} \cdot M(n-i) \cdot s^i + \sum_{i=1}^{m} \binom{m-1}{i-1} \cdot M(n-i) \cdot s^i \\
&= M(n) + M(n-m) \cdot s^m + \sum_{i=1}^{m-1} \left(\binom{m-1}{i} + \binom{m-1}{i-1} \right) \cdot M(n-i) \cdot s^i \\
&= \sum_{i=0}^{m} \binom{m}{i} \cdot M(n-i) \cdot s^i \,. \qquad \square
\end{aligned}$$

Lemma 3.6.6. *Let $\mathcal{H}$ be a set of functions $f : \mathbb{R}^d \to \{\pm 1\}$ such that for some $k \ge 1$, $M(n) \le k^n$. Then for any $\epsilon > 0$ and $r \ge 3dk/\epsilon$, $\mathsf{bbs}(\mathcal{H}, r) < \epsilon \cdot r^d$.*

Proof. Write $s = r - 1$. By Proposition 3.6.1 and Lemma 3.6.5, the probability that v is a non-constant block is

$$\begin{aligned}
\frac{\mathsf{bbs}(r)}{r^d} &\le r^{-d} \left(P_r(d,d) - r^d \right) \le r^{-d} \left(\sum_{i=0}^{d-1} \left[\binom{d}{i} \cdot M(d-i) \cdot s^i \right] + s^d - r^d \right) \\
&\le r^{-d} \sum_{i=0}^{d-1} \binom{d}{i} \cdot M(d-i) \cdot s^i \,.
\end{aligned}$$

Split the sum into two parts:

$$\begin{aligned}
&\sum_{i=0}^{\lfloor d/2 \rfloor} \binom{d}{i} \cdot \frac{M(d-i) \cdot s^i}{r^d} + \sum_{i=1}^{\lceil d/2 \rceil - 1} \binom{d}{i} \cdot \frac{M(i) \cdot s^{d-i}}{r^d} \\
&\quad \le \sum_{i=0}^{\lfloor d/2 \rfloor} \binom{d}{i} \cdot \frac{k^{d-i} \cdot r^i}{r^d} + \sum_{i=1}^{\lceil d/2 \rceil - 1} \binom{d}{i} \cdot \frac{k^i \cdot r^{d-i}}{r^d} \\
&\quad \le \sum_{i=0}^{\lfloor d/2 \rfloor} \frac{d^i k^{d-i} \cdot r^i}{r^d} + \sum_{i=1}^{\lceil d/2 \rceil - 1} \frac{d^i k^i \cdot r^{d-i}}{r^d} \qquad \le \sum_{i=0}^{\lfloor d/2 \rfloor} \frac{\epsilon^{d-i}}{3^{d-i} d^{d-i}} + \sum_{i=1}^{\lceil d/2 \rceil - 1} \frac{\epsilon^i}{3^i} \\
&\quad \le \frac{\epsilon}{3} + \sum_{i=2}^{\lceil d/2 \rceil - 1} \frac{\epsilon^i}{3^i} + \lfloor d/2 \rfloor \cdot \frac{\epsilon^{\lceil d/2 \rceil}}{3^{\lceil d/2 \rceil} d^{\lceil d/2 \rceil}} \qquad \le \frac{\epsilon}{3} + \frac{\epsilon}{3} \sum_{i=1}^{\infty} \frac{\epsilon^i}{3^i} + \frac{\epsilon^{\lceil d/2 \rceil}}{3^{\lceil d/2 \rceil}} \le \epsilon \,. \qquad \square
\end{aligned}$$

It is a standard fact that for degree-k polynomials, $M(1) \leq k$, and a special case of a theorem of Warren bounds gives a bound for larger dimensions:

Theorem 3.6.7 ([War68])**.** *Polynomial threshold functions $p : \mathbb{R}^d \to \{\pm 1\}$ of degree k have $M(n) \leq 6(2k)^n$.*

Since $M(1) \leq \sqrt{24}k$ and $6(2k)^n \leq (\sqrt{24}k)^n$, for $n > 1$, Lemma 3.6.6 gives us:

Corollary 3.6.8. *For $r \geq 3\sqrt{24}dk/\epsilon$, $r^{-d} \cdot \mathsf{bbs}(\mathcal{P}_k, r) < \epsilon$.*

3.6.2 Application

As was the case for halfspaces, our reduction of noise sensitivity on $[r]^d$ to $\{\pm 1\}^d$ requires that the class $\mathcal{P}_k$ is invariant under linear transformations:

Proposition 3.6.9. *For any $f \in \mathcal{P}_k$ and full-rank linear transformation $A \in \mathbb{R}^{d \times d}$, $f \circ A \in \mathcal{P}_k$.*

Proof. Let $f(x) = \operatorname{sign}(p(x))$ where p is a degree-k polynomial and let $c_q \prod_{i=1}^d x_i^{q_i}$ be a term of p, where $c \in \mathbb{R}$ and $q \in \mathbb{Z}^d_{\geq 0}$ such that $\sum_i q_i \leq k$. Let $A_i \in \mathbb{R}^d$ be the i^{th} row of A. Then

$$\prod_{i=1}^d (Ax)_i^{q_i} = \prod_{i=1}^d \left(\sum_{j=1}^d A_{i,j} x_j \right)^{q_i} = p_q(x)$$

where $p_q(x)$ is some polynomial of degree at most $\sum_{i=1}^d q_i \leq k$. Then $p \circ A = \sum_q c_q p_q$ where q ranges over $\mathbb{Z}_{\geq 0}$ with $\sum_i q_i \leq k$, and each p_q has degree at most k, so $p \circ A$ is a degree-k polynomial. □

The last ingredient we need is the following bound of Diakonikolas *et al.* on the noise sensitivity:

Theorem 3.6.10 ([DHK+10])**.** *Let $f : \{\pm 1\}^d \to \{\pm 1\}$ be a degree-k PTF. Then for any $\delta \in [0, 1]$,*

$$\mathsf{ns}_{2,\delta}(f) \leq O(\delta^{1/2^k})$$
$$\mathsf{ns}_{2,\delta}(f) \leq 2^{O(k)} \cdot \delta^{1/(4k+2)} \log(1/\delta)\,.$$

Putting everything together, we obtain a bound that is polynomial in d for any fixed k, ϵ, and which matches the result of Diakonikolas *et al.* [DHK+10] for the uniform distribution over $\{\pm 1\}^d$.

Theorem 1.2.13. *There is an improper agnostic learning algorithm for degree-k PTFs under (finite or continuous) product distributions over $\mathbb{R}^d$, with time complexity*

$$\min\left\{\left(\frac{kd}{\epsilon}\right)^{\psi(k,\epsilon)}, O\left(\frac{1}{\epsilon^2}\left(\frac{9dk}{\epsilon}\right)^d\right)\right\}.$$

Proof. We prove the continuous case here; the finite case is proved in Theorem 3.8.18.

Let $r = \lceil 9dk/\epsilon \rceil$, so that by Corollary 3.6.8, condition 1 of Lemma 3.5.2 is satisfied. Due to Proposition 3.6.9, we may apply Theorem 3.6.10 and Lemma 3.5.3 to conclude that for all $f \in \mathcal{P}_k$

$$\begin{aligned}\mathsf{ns}_{r,\delta}(f^{\mathsf{block}}) &\leq O(\delta^{1/2^k})\\ \mathsf{ns}_{r,\delta}(f^{\mathsf{block}}) &\leq 2^{O(k)} \cdot \delta^{1/(4k+2)} \log(1/\delta).\end{aligned}$$

In the first case, setting $\delta = O(\epsilon^{2^{k+1}})$ we get $\mathsf{ns}_{r,\delta}(f^{\mathsf{block}}) < \epsilon^2/3$, so by Lemma 3.5.6 we get a set $\mathcal{F}$ of functions $[r]^d \to \mathbb{R}$ of size $|\mathcal{F}| \leq (rd)^{O\left(\frac{1}{\epsilon^{2^{k+1}}}\right)}$ satisfying condition 2 of Lemma 3.5.2. For $n = \mathrm{poly}(|\mathcal{F}|, 1/\epsilon)$, Lemma 3.5.2 implies an algorithm with sample size

$$O(rd^2n^2\log(rd)) = O\left(\frac{d^3k}{\epsilon}\log(dk/\epsilon)\right) \cdot \left(\frac{kd}{\epsilon}\right)^{O\left(\frac{1}{\epsilon^{2^{k+1}}}\right)}.$$

In the second case, setting $\delta = O\left(\left(\frac{2^{O(k)}\log(2^k/\epsilon)}{\epsilon^2}\right)^{4k+2}\right)$, we again obtain $\mathsf{ns}(f^{\mathsf{block}})_{r,\delta} \leq \epsilon^2/3$ and get an algorithm with sample size

$$\left(\frac{kd}{\epsilon}\right)^{2^{O(k^2)}\left(\frac{\log(1/\epsilon)}{\epsilon^2}\right)^{4k+2}}.$$

The final result is obtained by applying Lemma 3.4.8. □

3.7 Learning & Testing k-Alternating Functions

A function $f : \mathcal{X} \to \{\pm 1\}$ on a partial order $\mathcal{X}$ is *k-alternating* if for every chain $x_1 < \ldots < x_{k+2}$ there is $i \in [k+1]$ such that $f(x_i) = f(x_{i+1})$. In other words, on any chain, the function changes value at most k times. Monotone functions are examples of 1-alternating functions. We consider k-alternating functions on $\mathbb{R}^d$ with the usual partial order: for $x, y \in \mathbb{R}^d$ we say $x < y$ when $x_i \leq y_i$ for each $i \in [d]$ and $x \neq y$.

Learning k-alternating functions on domain $\{\pm 1\}^d$ was studied by Blais *et al.* [BCO+15], motivated by the fact that these functions are computed by circuits with few negation gates. They show that $2^{\Theta(k\sqrt{d}/\epsilon)}$ samples are necessary and sufficient in this setting. Canonne *et al.* [CGG+19] later obtained an algorithm for (ϵ_1, ϵ_2)-tolerant testing k-alternating functions, when $\epsilon_2 > 3\epsilon_1$, in the uniform distribution over $[n]^d$, with query complexity $(kd/\tau)^{O(k\sqrt{d}/\tau^2)}$, where $\tau = \epsilon_2 - 3\epsilon_1$.

We obtain an agnostic learning algorithm for k-alternating functions that matches the query complexity of the tester in [CGG+19], and nearly matches the complexity of the (non-agnostic) learning algorithm of [BCO+15] for the uniform distribution over the hypercube.

Theorem 3.7.1. *There is an agnostic learning algorithm for k-alternating functions under (finite or continuous) product distributions over $\mathbb{R}^d$ that runs in time at most*

$$\min\left\{\left(\frac{dk}{\epsilon}\right)^{O\left(\frac{k\sqrt{d}}{\epsilon^2}\right)}, O\left(\frac{1}{\epsilon^2}\left(\frac{3kd}{\epsilon}\right)^d\right)\right\}.$$

We also generalize the tolerant tester of [CGG+19] to be distribution-free under product distributions, and eliminate the condition $\epsilon_2 > 3\epsilon_1$.

Theorem 3.7.2. *For any $\epsilon_2 > \epsilon_1 > 0$, let $\tau = (\epsilon_2 - \epsilon_1)/2$, there is a sample-based (ϵ_1, ϵ_2)-tolerant tester for k-alternating functions using $\left(\frac{dk}{\tau}\right)^{O\left(\frac{k\sqrt{d}}{\tau^2}\right)}$ samples, which is distribution-free under (finite or continuous) product distributions over $\mathbb{R}^d$.*

To prove these theorems, we require a bound the block boundary size, which has been done already by Canonne *et al.*. We include the proof because their work does not share our definition of block boundary size, and because we have used it in our short proof of the monotonicity tester in Section 3.3.1.

Lemma 3.7.3 ([CGG+19])**.** *The r-block boundary size of k-alternating functions is at most kdr^{d-1}.*

Proof. Let f be k-alternating, let $\mathsf{block} : \mathbb{R}^d \to [r]^d$ be any block partition and let $v_1, \ldots, v_m$ be any chain in $[r]^d$. Suppose that there are $k+1$ indices $i_1, \ldots, i_{k+1}$ such that f is not constant on $\mathsf{block}^{-1}(v_{i_j})$. Then there is a set of points $x_1, \ldots, x_{k+1}$ such that $x_j \in \mathsf{block}^{-1}(v_{i_j})$ and $x_j \neq x_{j+1}$ for each $j \in [k]$. But since $v_{i_1} < \cdots < v_{i_{k+1}}$, $x_1 < \cdots < x_{k+1}$ also, which contradicts the fact that f is k-alternating. Then every chain in $[r]^d$ has at most k non-constant blocks, and we may partition $[r]^d$ into at most dr^{d-1} chains by taking the diagonals $v + \lambda\vec{1}$ where v is any vector satisfying $\exists i : v_i = 1$ and λ ranges over all integers. □

Canonne *et al.* also use noise sensitivity bound to obtain a spanning set $\mathcal{F}$; we quote their result.

Lemma 3.7.4 ([CGG+19])**.** *There is a set $\mathcal{F}$ of functions $[r]^d \to \mathbb{R}$, with size*

$$|\mathcal{F}| \leq \exp\, O\left(\frac{k\sqrt{d}}{\epsilon^2}\log(rd/\epsilon)\right),$$

such that for any k-alternating function $h : [r]^d \to \{\pm 1\}$, there is $g : [r]^d \to \mathbb{R}$ that is a linear combination of functions in $\mathcal{F}$ and $\mathop{\mathbb{E}}_{x\sim[r]^d}\left[(h(x)-g(x))^2\right] \leq \epsilon^2$.

Finally, we prove Theorem 3.7.1.

Proof of Theorem 3.7.1. We prove the continuous case; see Theorem 3.8.18 for the finite case.

Let $r = \lceil dk/\epsilon \rceil$ and let $\mathsf{block} : \mathbb{R}^d \to [r]^d$ be any r-block partition. By Lemma 3.7.3, the first condition of Lemma 3.5.2 is satisfied. Now let $f \in \mathcal{H}$ and consider f^{block}. For any chain $v_1 < v_2 < \cdots < v_m$ in $[r]^d$, it must be $\mathsf{blockpoint}(v_1) < \mathsf{blockpoint}(v_2) < \cdots < \mathsf{blockpoint}(v_m)$ since every $x \in \mathsf{block}^{-1}(v_i), y \in \mathsf{block}^{-1}(v_j)$ satisfy $x < y$ when $v_i < v_j$; then f alternates at most k times on the chain $\mathsf{blockpoint}(v_1) < \cdots < \mathsf{blockpoint}(v_m)$ and, since $f^{\mathsf{block}}(v_i) = f(\mathsf{blockpoint}(v_i))$, f^{block} is also k-alternating. Therefore the set $\mathcal{F}$ of functions given by Lemma 3.7.4 satisfies condition 2 of Lemma 3.7.3, and we have $n = \mathrm{poly}(|\mathcal{F}|, 1/\epsilon) = \exp\, O\left(\frac{k\sqrt{d}}{\epsilon^2}\log(rd/\epsilon)\right)$. Applying Lemma 3.5.2 gives an algorithm with sample complexity

$$O\left(rd^2n^2\log(rd)\right) = O\left(\frac{d^3k}{\epsilon}\log(dk/\epsilon)\cdot\left(\frac{dk}{\epsilon}\right)^{O\left(\frac{k\sqrt{d}}{\epsilon^2}\right)}\right) = \left(\frac{dk}{\epsilon}\right)^{O\left(\frac{k\sqrt{d}}{\epsilon^2}\right)}.$$

The other sample complexity follows from Lemma 3.4.8. □

Next we prove Theorem 3.7.2.

Proof of Theorem 3.7.2. The following argument is for the continuous case, but generalizes to the finite case using the definitions in Section 3.8.

Let $\mathcal{H}$ be the class of k-alternating functions. Suppose there is a set $\mathcal{K} \subset \mathcal{H}$, known to the algorithm, that is a $(\tau/2)$-cover. Then, taking a set Q of $q = O(\frac{1}{\epsilon^2} \log |\mathcal{K}|)$ independent random samples from μ and using Hoeffding's inequality,

$$\begin{aligned}\mathop{\mathbb{P}}_{Q}\left[\exists h \in \mathcal{K} : |\mathsf{dist}_Q(f,h) - \mathsf{dist}_\mu(f,h)| > \frac{\tau}{2}\right] &\le |\mathcal{K}| \cdot \max_{h\in\mathcal{K}} \mathbb{P}\left[|\mathsf{dist}_Q(f,h) - \mathsf{dist}_\mu(f,h)| > \frac{\tau}{2}\right] \\ &\le |\mathcal{K}| \cdot 2\exp\left(-\frac{q\tau^2}{2}\right) < 1/6\,.\end{aligned}$$

Then the tester accepts if $\mathsf{dist}_Q(f,\mathcal{K}) < \epsilon_1 + \tau$ and rejects otherwise; we now prove that this is correct with high probability. Assume that the above estimation is accurate, which occurs with probability at least $5/6$. If $\mathsf{dist}_\mu(f,\mathcal{H}) \le \epsilon_1$ then $\mathsf{dist}_\mu(f,\mathcal{K}) \le \mathsf{dist}_\mu(f,h) + \mathsf{dist}_\mu(h,\mathcal{K}) \le \epsilon_1 + \tau/2$. Then for $g \in \mathcal{K}$ minimizing $\mathsf{dist}_\mu(f,g)$,

$$\mathsf{dist}_Q(f,\mathcal{K}) \le \mathsf{dist}_Q(f,g) < \mathsf{dist}_\mu(f,g) + \frac{\tau}{2} \le \epsilon_1 + \tau\,,$$

so f is accepted. Now suppose that f is accepted, so $\mathsf{dist}_Q(f,\mathcal{K}) < \epsilon_1 + \tau$. Then

$$\mathsf{dist}_\mu(f,\mathcal{H}) \le \mathsf{dist}_\mu(f,g) \le \mathsf{dist}_Q(f,g) + \frac{\tau}{2} < \epsilon_1 + \frac{3}{2}\tau = \epsilon_1 + \frac{3}{4}(\epsilon_2 - \epsilon_1) \le \epsilon_2\,.$$

What remains is to show how the tester constructs such a cover $\mathcal{K}$.

Consider the learning algorithm of Theorem 3.7.1 with error parameter $\tau/12$, so $r = \lceil 12dk/\tau \rceil$. Let X be the grid constructed by that algorithm and let $\mathsf{block} : \mathbb{R}^d \to [r]^d$ be the induced r-block partition. We may assume that with probability at least $5/6$, $\|\mathsf{block}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} < \tau/12$; suppose that this event occurs. The learner then takes $m = \left(\frac{dk}{\tau}\right)^{O\left(\frac{k\sqrt{d}}{\tau^2}\right)}$ additional samples to learn the class $\mathcal{H}^{\mathsf{block}}$ with domain $[r]^d$. For every $f \in \mathcal{H}$ the learner has positive probability of outputting a function $h : [r]^d \to \{0,1\}$ with $\mathop{\mathbb{P}}_{v}\left[h(v) \ne f^{\mathsf{block}}(v)\right] < \tau/12$ (where v is chosen from $\mathsf{block}(\mu)$). Let $\mathcal{K}'$ be the set of possible outputs of the learner; then $\mathcal{K}'$ is a $(\tau/12)$-cover for $\mathcal{H}^{\mathsf{block}}$. Construct a set $\mathcal{K}^{\mathsf{block}}$ by choosing, for each $h \in \mathcal{K}'$, the nearest function $g \in \mathcal{K}$ with respect to the distribution $\mathsf{block}(\mu)$. Then $\mathcal{K}^{\mathsf{block}}$ is a $(\tau/6)$-cover, since for any function $f^{\mathsf{block}} \in \mathcal{H}^{\mathsf{block}}$, if $h \in \mathcal{K}'$ is the nearest output of the learner and $g \in \mathcal{K}^{\mathsf{block}}$ is nearest h, then by the triangle inequality f^{block} has distance at most $\tau/6$ to g with respect to $\mathsf{block}(\mu)$. Finally, construct a set

$\mathcal{K} \subset \mathcal{H}$ by taking each function $h \in \mathcal{H}$ such that $h^{\mathsf{coarse}} = h$ and $h^{\mathsf{block}} \in \mathcal{K}^{\mathsf{block}}$ (note that there exists $h \in \mathcal{H}$ such that $h^{\mathsf{coarse}} = h$ since h^{coarse} is k-alternating when h^{block} is k-alternating). Then $\mathcal{K}$ is a $(\tau/2)$-cover since for any $f \in \mathcal{H}$, when $h \in \mathcal{K}$ minimizes $\mathop{\mathbb{P}}_{v \sim \mathsf{block}(\mu)} \left[f^{\mathsf{block}}(v) \neq h^{\mathsf{block}} \right]$,

$$\begin{aligned}
&\mathsf{dist}_\mu(f, \mathcal{K}) \\
&\quad \leq \mathsf{dist}_\mu(f, f^{\mathsf{coarse}}) + \mathsf{dist}_\mu(f^{\mathsf{coarse}}, \mathcal{K}) \\
&\quad \leq r^{-d} \cdot \mathsf{bbs}(\mathcal{H}, r) + \mathop{\mathbb{P}}_{v \sim \mathsf{block}(\mu)} \left[f^{\mathsf{block}}(v) \neq h^{\mathsf{block}}(v) \right] + 2 \| \mathsf{block}(\mu) - \mathsf{unif}([r]^d) \|_{\mathsf{TV}} \\
&\quad < \tau/6 + \tau/6 + 2\tau/12 \leq \tau/2 \,.
\end{aligned}$$

Now we bound the size of $\mathcal{K}^{\mathsf{block}}$. Since there are m samples and each sample $v \sim \mathsf{block}(\mu)$ is in $[r]^d$, labelled by $\{0, 1\}$, there are at most $(r^d)^m 2^m$ possible sample sequences, so at most $(2r^d)^m$ outputs of the learner (after constructing X), so $|\mathcal{K}^{\mathsf{block}}| \leq (2r^d)^m$. Therefore, after constructing X, the tester may construct $\mathcal{K}^{\mathsf{block}}$ and run the above estimation procedure, with $q = O\left(\frac{1}{\tau^2} dm \log r\right) = \left(\frac{dk}{\tau}\right)^{O\left(\frac{k\sqrt{d}}{\tau^2}\right)}$. □

3.8 Discrete Distributions

We will say that a distribution μ_i over $\mathbb{R}$ is *finite* if it is a distribution over a finite set $X \subset \mathbb{R}$. In this section, we extend downsampling to work for finite product distributions: distributions $\mu = \mu_1 \times \cdots \times \mu_d$ such that all μ_i are finite. As mentioned in the introduction, our algorithms have the advantage that they do not need to know in advance whether the distribution is continuous or finite, and if they are finite they do not need to know the support. This is in contrast to the algorithms of Blais *et al.* [BOW10], which work for arbitrary finite product distributions but must know the support (since it learns a function under the "one-out-of-k encoding"). Our algorithms have superior time complexity for large supports.

We begin with an example of a pathological set of functions that illustrates some of the difficulties in the generalization.

Example 3.8.1. The *Dirichlet function* $f : \mathbb{R} \to \{\pm 1\}$ is the function that takes value 1 on all rational numbers and value -1 on all irrational numbers. We will define the *Dirichlet class* of functions as the set of all functions $f : \mathbb{R}^d \to \{\pm 1\}$ such that $f(x) = -1$ on all x with at least 1 irrational coordinate x_i, and $f(x)$ is arbitrary for any x with all rational

coordinates. Since the Lebesgue measure of the set of rational numbers is 0, in any continuous product distribution, any function f in the Dirichlet class satisfies $\mathbb{P}[f(x) \neq -1] = 0$; therefore learning this class is trivial in any continuous product distribution since we may output the constant -1 function. And $\mathsf{bbs}(f, r) = 0$ for this class since no block contains a set S of positive measure containing 1-valued points. On the other hand, if μ is a finitely supported product distribution, then it may be the case that it is supported *only* on points with all rational coordinates. In that case, the Dirichlet class of functions is the set of all functions on the support, which is impossible to learn when the size of the support is unknown (since the number of samples will depend on the support size). It is apparent that our former definition of bbs no longer suffices to bound the complexity of algorithms when we allow finitely supported distributions.

Another difficulty arises for finitely supported distributions with small support: for example, the hypercube $\{\pm 1\}^d$. Consider what happens when we attempt to sample a uniform grid, as in the first step of the algorithms above. We will sample many points x such that $x_1 = 1$ and many points such that $x_1 = -1$. Essentially, the algorithm takes a small domain $\{\pm 1\}^d$ and constructs the larger domain $[r]^d$, which is antithetical to the downsampling method. A similar situation would occur in large domains $[n]^d$ where some coordinates have exceptionally large probability densities and are sampled many times. Our algorithm must be able to handle such cases, so we must redefine the grid sampling step and block partitions to handle this situation. To do so, we introduce *augmented samples*: for every sample point $x \sim \mu$ we will append a uniformly random value in $[0, 1]^d$.

3.8.1 Augmented Samples & Constructing Uniform Partitions

For *augmented points* $\overline{a}, \overline{b}, \in \mathbb{R} \times [0, 1]$, where $\overline{a} = (a, a'), \overline{b} = (b, b')$, we will define a total order by saying $\overline{a} < \overline{b}$ if $a < b$, or $a = b$ and $a' < b'$. Define interval $(\overline{a}, \overline{b}] := \{\overline{c} \mid \overline{a} < \overline{c} \leq \overline{b}\}$. For convenience, when $\overline{a} \in \mathbb{R} \times [0, 1]$ and $\overline{a} = (a, a')$ we will write $\xi(\overline{a}) = a$. If $\overline{x} \in \mathbb{R}^d \times [0, 1]^d$ is an augmented vector (i.e. each coordinate $\overline{x}_i$ is an augmented point), we will write $\xi(\overline{x}) = (\xi(x_1), \ldots, \xi(x_d))$; and when $S \subseteq \mathbb{R}^d \times [0, 1]^d$ is a set of augmented points, we will write $\xi(S) = \{\xi(\overline{x}) \mid \overline{x} \in S\}$.

Definition 3.8.2 (Augmented Block Partition)**.** An *augmented r-block partition* of $\mathbb{R}^d$ is a pair of functions $\overline{\mathsf{block}} : \mathbb{R}^d \times [0, 1]^d \to [r]^d$ and $\mathsf{blockpoint} : [r]^d \to \mathbb{R}^d$ obtained as follows. For each $i \in [d], j \in [r - 1]$ let $\overline{a}_{i,j} \in \mathbb{R} \times [0, 1]$ such that $\overline{a}_{i,j} < \overline{a}_{i,j+1}$ and define $\overline{a}_{i,0} = (-\infty, 0), \overline{a}_{i,r} = (\infty, 1)$. For each $i \in [d], j \in [r]$ define the interval $B_{i,j} = (\overline{a}_{i,j-1}, \overline{a}_{i,j}]$ and a point $\overline{b}_{i,j} \in \mathbb{R} \times [0, 1]$ such that $\overline{a}_{i,j} \leq \overline{b}_{i,j} \leq \overline{a}_{i,j+1}$. The function $\overline{\mathsf{block}} : \mathbb{R}^d \times [0, 1]^d \to [r]^d$

is defined by setting $\overline{\mathsf{block}}(\overline{x})$ to be the unique vector $v \in [r]^d$ such that $\overline{x}_i \in B_{i,v_i}$ for each $i \in [d]$. Observe that $\overline{\mathsf{block}}^{-1}(v) := \{\overline{x} : \overline{\mathsf{block}}(x) = v\}$ is a set of augmented points in $\mathbb{R}^d \times [0,1]$ and that it is possible for two augmented points $\overline{x}, \overline{y}$ to satisfy $\xi(\overline{x}) = \xi(\overline{y})$ while $\overline{\mathsf{block}}(\overline{x}) \neq \overline{\mathsf{block}}(\overline{y})$. The function $\mathsf{blockpoint} : [r]^d \to \mathbb{R}^d$ is defined by setting $\mathsf{blockpoint}(v) = (\xi(\overline{b}_{1,v_1}), \ldots, \xi(\overline{b}_{d,v_d}))$; note that this is a non-augmented point.

Definition 3.8.3 (Block Functions and Coarse Functions)**.** For a function $f : \mathbb{R}^d \to \{\pm 1\}$ we will define the functions $f^{\mathsf{block}} : [r]^d \to \{\pm 1\}$ as $f^{\mathsf{block}} := f \circ \mathsf{blockpoint}$ and for each $z \in [0,1]^d$ we will define $f_z^{\mathsf{coarse}} : \mathbb{R}^d \to \{\pm 1\}$ as $f_z^{\mathsf{coarse}}(x) := f^{\mathsf{block}}(\overline{\mathsf{block}}(x, z))$. Unlike in the continuous setting, f_z^{coarse} depends on an additional variable $z \in [0,1]^d$, which is necessary because a single point $x \in \mathbb{R}^d$ may be augmented differently to get different $\overline{\mathsf{block}}$ values. For a distribution μ over $\mathbb{R}^d$ define the augmented distribution $\overline{\mu}$ over $\mathbb{R}^d \times [0,1]^d$ as the distribution of (x, z) when $x \sim \mu$ and z is uniform in $[0,1]^d$. For an augmented r-block partition $\overline{\mathsf{block}} : \mathbb{R}^d \times [0,1]^d \to [r]^d$ we define the distribution $\overline{\mathsf{block}}(\mu)$ over $[r]^d$ as the distribution of $\overline{\mathsf{block}}(\overline{x})$ for $\overline{x} \sim \overline{\mu}$.

Definition 3.8.4 (Augmented Random Grid)**.** An *augmented random grid* $\overline{X}$ of length m is obtained by sampling m augmented points $\overline{x}_1, \ldots, \overline{x}_m \sim \overline{\mu}$ and for each $i \in [d], j \in [m]$ defining $\overline{X}_{i,j}$ to the be j^{th} smallest coordinate in dimension i by the augmented partial order. For any r that divides m we define an augmented r-block partition depending on $\overline{X}$ by defining for each $i \in [d], j \in [r-1]$ the points $\overline{a}_{i,j} = \overline{X}_{i,mj/r}$, (and $\overline{a}_{i,0} = (-\infty, 0), \overline{a}_{i,r} = (\infty, 1)$), so that the intervals are $B_{i,j} = (\overline{X}_{i,m(j-1)/r}, \overline{X}_{i,mj/r}]$ for $j \in \{2, \ldots, r-1\}$ and $B_{i,1} = ((-\infty, 0), \overline{X}_{i,m/r}], B_{i,r} = (\overline{X}_{i,m(r-1)/r}, (\infty, 1)]$. We set the points $\overline{b}_{i,j}$ defining $\mathsf{blockpoint} : [r]^d \to \mathbb{R}^d$ to be $\overline{b}_{i,j} = \overline{X}_{i,k}$ for some $\overline{X}_{i,k} \in B_{i,j}$. This is the augmented r-block partition *induced* by $\overline{X}$.

Definition 3.8.5 (Augmented Block Boundary)**.** For an augmented block partition $\overline{\mathsf{block}} : \mathbb{R}^d \times [0,1]^d \to [r]^d$, a distribution μ over $\mathbb{R}^d$, and a function $f : \mathbb{R}^d \to \{\pm 1\}$, we say f is *non-constant* on an augmented block $\overline{\mathsf{block}}^{-1}(v)$ if there are sets $\overline{S}, \overline{T} \subset \overline{\mathsf{block}}^{-1}(v)$ such that $\mu(\xi(\overline{S})), \mu(\xi(\overline{T})) > 0$ and for all $s \in S, t \in T : f(s) = 1, f(t) = -1$. For a function $f : \mathbb{R}^d \to \{\pm 1\}$ and a number r, we define the augmented r-block boundary size $\overline{\mathsf{bbs}}(f, r)$ as the maximum number of blocks on which f is non-constant with respect to a distribution μ, where the maximum is taken over all augmented r-block partitions.

The augmented block partitions satisfy analogous properties to the previously-defined block partitions:

Lemma 3.8.6. *Let $\overline{X}$ be an augmented random grid with length m sampled from a finite product distribution μ, and let $\overline{\mathsf{block}} : \mathbb{R}^d \times [0,1]^d \to [r]^d$ be the augmented r-block partition*

induced by $\overline{X}$. *Then*

$$\mathop{\mathbb{P}}_{\overline{X}}\left[\|\overline{\mathsf{block}}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} > \epsilon\right] \le 4rd \cdot \exp\left(-\frac{m\epsilon^2}{18rd^2}\right) .$$

Proof. Let μ_i be a finitely supported distribution with support $S \subset \mathbb{R}$, and let $\eta = \frac{1}{2}\min_{a,b\in S}|a-b|$. Let μ_i' be the distribution of $x_i + \eta z_i$ where $x_i \sim \mu_i$ and $z_i \sim [0,1]$ uniformly at random; note that μ_i' is a continuous distribution over $\mathbb{R}$. For $\overline{x} = (x, x'), \overline{y} = (y, y') \in \mathbb{R} \times [0,1]$, observe that $\overline{x} < \overline{y}$ iff $x + \eta x' < y + \eta y'$. Therefore,

$$\mathop{\mathbb{P}}_{\overline{x},\overline{y}\sim\overline{\mu}_i}[\overline{x} < \overline{y}] = \mathop{\mathbb{P}}_{x,y\sim\mu_i'}[x < y] .$$

By replacing each finitely supported μ_i with μ_i' we obtain a continuous product distribution μ' such that $\overline{\mathsf{block}}(\mu)$ is the same distribution as $\mathsf{block}(\mu')$, so by Lemma 3.2.7 the conclusion holds. □

Proposition 3.8.7. *For any continuous or finite product distribution* μ *over* $\mathbb{R}^d$, *any augmented* r*-block partition* $\overline{\mathsf{block}} : \mathbb{R}^d \times [0,1]^d \to [r]^d$ *constructed from a random grid* $\overline{X}$, *and any* $f : \mathbb{R}^d \to \{\pm 1\}$,

$$\mathop{\mathbb{P}}_{x\sim\mu, z\sim[0,1]^d}[f(x) \ne f_z^{\mathsf{coarse}}(x)] \le r^{-d} \cdot \overline{\mathsf{bbs}}(f, r) + \|\overline{\mathsf{block}}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} .$$

Proof. The result for continuous product distributions holds by Proposition 3.2.5 and the fact that $\mathsf{bbs}(f,r) \le \overline{\mathsf{bbs}}(f,r)$, so assume μ is a finite product distribution, and let $S = \mathrm{supp}(\mu)$.

Suppose that for (x,z) sampled from $\overline{\mu}$, $f(x) \ne f_z^{\mathsf{coarse}}(x)$, and let $v = \overline{\mathsf{block}}(x,z)$. Then for $y = \mathsf{blockpoint}(v)$, $f(x) \ne f(y)$ and $x, y \in \xi(\overline{\mathsf{block}}^{-1}(v))$. The points x, y clearly have positive measure because μ is finite, so v a non-constant block. Then

$$\begin{aligned}\mathop{\mathbb{P}}_{x\sim\mu, z\sim[0,1]^d}[f(x) \ne f_z^{\mathsf{coarse}}(x)] &\le \mathop{\mathbb{P}}_{x,z}\left[\overline{\mathsf{block}}(x,z) \text{ is non-constant}\right] \\ &\le \mathop{\mathbb{P}}_{v\sim[r]^d}[v \text{ is non-constant}] + \|\overline{\mathsf{block}}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} . \ \square\end{aligned}$$

3.8.2 Augmented Block-Boundary Size and Noise Sensitivity

To obtain learning algorithms for k-alternating functions, functions of k convex sets, functions of k halfspaces, and degree-k PTFs, we must provide a bound on $\overline{\mathsf{bbs}}$.

For a finite set $X \subset \mathbb{R}^d$ and a function $f : \mathbb{R}^d \to \{\pm 1\}$, we will call a function $f' : \mathbb{R}^d \to \{\pm 1\}$ a *blowup* of f (with respect to X) if $\forall x \in X$ there exists an open ball $B_x \ni x$ where $\forall y \in B_x, f'(y) = f(x)$. We will call a set $\mathcal{H}$ of functions $f : \mathbb{R}^d \to \{\pm 1\}$ *inflatable* if for every finite product set $X = X_1 \times \cdots \times X_d$ and $f \in \mathcal{H}$, there exists $f' \in \mathcal{H}$ that is a blowup of f with respect to X.

Proposition 3.8.8. *Let $\mathcal{H}$ be a inflatable set of functions. Then $\overline{\mathsf{bbs}}(\mathcal{H}, r) \leq \mathsf{bbs}(\mathcal{H}, r)$.*

Proof. Let $\overline{\mathsf{block}} : \mathbb{R}^d \times [0,1]^d \to [r]^d$ be an augmented r-block partition defined by parameters $\overline{b}_{i,j} \in \mathbb{R} \times [0,1]$ for $i \in [d], j \in [r-1]$, and write $\overline{b}_{i,j} = (b_{i,j}, b'_{i,j})$. Let $X = X_1 \times \cdots \times X_d$ be any finite product set, and let $f \in \mathcal{H}$; we will bound the number of non-constant blocks We construct a (non-augmented) r-block partition as follows. Let $\eta > 0$ be sufficiently small that:

- $\forall x \in X$, the rectangle $R_x := [x_1, x_1 + \eta] \times \cdots \times [x_d, x_d + \eta]$ is contained within B_x,
- $\forall i \in [d], [x_i, x_i + \eta] \cap X_i = \{x_i\}$; and
- $\forall i \in [d], b_{i,j} + \eta < b_{i,j+1}$ unless $b_{i,j} = b_{i,j+1}$.

Such an η exists since the number of constraints is finite. Then define $\mathsf{block} : \mathbb{R}^d \to [r]^d$ by the parameters $c_{i,j} = b_{i,j} + \eta \cdot b'_{i,j}$. Note that $c_{i,j} = b_{i,j} + \eta \cdot b'_{i,j} \leq b_{i,j} + \eta < b_{i,j+1} \leq c_{i,j+1}$. Let $v \in [r]^d$ and suppose that f is non-constant on $\overline{\mathsf{block}}^{-1}(v)$, so there are $\overline{x}, \overline{y} \in \overline{\mathsf{block}}^{-1}(v) \cap (X \times [0,1]^d)$ such that $f(x) \neq f(y)$, where $\overline{x} = (x, x'), \overline{y} = (y, y')$, and $\forall i \in [d], x_i, y_i \in (b_{i,v_i-1}, b_{i,v_i}]$ where we define $(b, b] = \{b\}$. Consider $\mathsf{block}^{-1}(v) = (c_{1,v_1-1}, c_{1,v_1}] \times \cdots \times (c_{d,v_d-1}, c_{d,v_d}]$.

Since $\overline{x}_i \in (\overline{b}_{i,v_i-1}, \overline{b}_{i,v_i}]$, $x_i \in (b_{i,v_i-1}, b_{i,v_i}]$ (where we define $(b, b] = \{b\}$) and $x'_i \in (b'_{i,v_i-1}, b'_{i,v_i}]$. Therefore $x_i + \eta \cdot x'_i \leq b_{i,v_i} + \eta \cdot b'_{i,v_i} = c_{i,v_i}$ and $x_i + \eta \cdot x'_i > b_{i,v_i-1} + \eta \cdot b'_{i,v_i-1} = c_{i,v_i-1}$ so $x + \eta \cdot x' \in \mathsf{block}^{-1}(v)$. Also, $x + \eta \cdot x'$ is in the rectangle $R_x \subset B_x$ so there is a ball around $x + \eta \cdot x'$, containing only points with value $f'(x) = f(x)$. Likewise, there is a ball around $y + \eta \cdot y'$ inside $\mathsf{block}^{-1}(v)$ containing only points with value $f'(y) = f(y) \neq f'(x)$. Since these balls must intersect $\mathsf{block}^{-1}(v)$ on sets with positive measure (in the product of Lebesgue measures), f' is non-constant on $\mathsf{block}^{-1}(v)$, which proves the statement. □

Lemma 3.8.9. *The set $\mathcal{A}_k$ of k-alternating functions is inflatable.*

Proof. Let $f \in \mathcal{A}_k$ and let $X = X_1 \times \cdots \times X_d$ be a finite set. We use the standard ordering on $\mathbb{R}^d$. Let $u \in \mathbb{R}^d$. We claim that the set $\{x \in X : x \leq u\}$ has a unique

maximum. Suppose otherwise, so there are $x, y \leq u$ that are each maximal. Let $x \wedge y = (\max(x_1, y_1), \ldots, \max(x_d, y_d))$. Then $x \vee y \in X$ and $x \wedge y > x, y$ but $u \geq x \wedge y$, a contradiction. For every $u \in \mathbb{R}^d$, write $u^{\downarrow}$ for this unique maximum. Let $\eta > 0$ be small enough that $\forall x \in X, (x + \eta \cdot \vec{1})^{\downarrow} = x$; such a value exists since X is finite. Define the map $\phi(u) = (u + (\eta/2) \cdot \vec{1})^{\downarrow}$ and $\forall u \in \mathbb{R}^d$, we define $f'(u) := f(\phi(u))$, and argue that this satisfies the required properties. It is clear by our choice of η that $f'(x) = f((x + (\eta/2) \cdot \vec{1})^{\downarrow}) = f(x)$. Since ϕ is order-preserving (i.e. if $u < v$ then $\phi(u) \leq \phi(v)$), f' is k-alternating. Now consider the ball $B(x) := \{y \in \mathbb{R}^d : \|y - x\|_2 < \eta/2\}$. Since $|y_i - x_i| < \eta/2$ for all $y \in B(x)$, we have $\phi(y) = (y_1 + \eta/2, \ldots, y_d + \eta/2)^{\downarrow} \leq (x_1 + \eta, \ldots, x_d + \eta)^{\downarrow} = x$, and $\phi(y) \geq (x_1, \ldots, x_d)^{\downarrow} = x$ so $f'(y) = f(\phi(y)) = f(x)$. □

Lemma 3.8.10. *The set $\mathcal{C}$ of indicator functions of convex sets is inflatable.*

Proof. Let $f : \mathbb{R}^d \to \{\pm 1\}$ indicate a closed convex set, let $S = f^{-1}(1)$ be this set, and write $\delta(x) := \min\{\|x - y\|_2 : y \in S\}$ (this minimum exists since S is closed). Let X be any finite set and let $\delta = \min\{\delta(x) : x \in X \setminus S\}$. Consider $S' = \{x : \delta(x) \leq \delta/2\}$, and let f' be the indicator function for this set. Then $f'(x) = f(x)$ for all $x \in X$. Finally, S' is closed, and it is convex since for any two points x, y, it is well-known that the function $\lambda \mapsto \delta(\lambda x + (1 - \lambda)y)$ is convex for $\lambda \in [0, 1]$. □

Lemma 3.8.11. *The set $\mathcal{H}$ of halfspaces is inflatable.*

Proof. It suffices to show that for any finite set X (not necessarily a product set) and any halfspace $f(x) = \text{sign}(\langle w, x \rangle - t)$, there is a halfspace $f'(x) = \text{sign}(\langle w, x \rangle - t')$ such that $f(x) = f'(x)$ for all $x \in X$ but $\langle w', x \rangle - t \neq 0$ for all $x \in X$; this is a commonly-used fact. Let $\delta = \min\{-(\langle w, x \rangle - t) : \langle w, x \rangle - t < 0\}$. It must be the case that $\delta > 0$. Then $f'(x) = \text{sign}(\langle w, x \rangle - t + \delta/2)$ satisfies the condition. □

Lemma 3.8.12. *The set $\mathcal{P}_k$ of degree-k PTFs is inflatable.*

Proof. This follows from the above proof for halfspaces, since for any finite X we may map $x \in X$ to its vector $(x_1^k, x_2^k, \ldots)$ of monomials, so that any polynomial $p(x)$ is a linear threshold function in the space of monomials. □

For a set $\mathcal{H}$ of functions $f : \mathbb{R}^d \to \{\pm 1\}$ and an augmented r-block partition $\overline{\mathsf{block}} : \mathbb{R}^d \to [r]^d$, we will write $\overline{\mathcal{H}}^{\mathsf{block}} := \{f^{\mathsf{block}} : f \in \mathcal{H}\}$ for the set of block functions $f^{\mathsf{block}} : [r]^d \to \{\pm 1\}$; note that this is not necessarily the same set of functions as $\mathcal{H}^{\mathsf{block}}$ defined for continuous distributions. We must show that the same learning algorithms used above for learning $\mathcal{H}^{\mathsf{block}}$ will work also for $\overline{\mathcal{H}}^{\mathsf{block}}$. For the brute-force learning algorithm of

Lemma 3.4.8, this is trivial, but for the regression algorithm in Lemma 3.5.2 we must show that there exists a set $\mathcal{F}$ such that each $f^{\mathsf{block}} \in \overline{\mathcal{H}}^{\mathsf{block}}$ is close to a function $g \in \mathsf{span}(\mathcal{F})$. For functions of halfspaces and PTFs, we used the bound on noise sensitivity, Lemma 3.5.3, to construct a set $\mathcal{F}$ of functions suitable for the regression algorithm. The proof for that lemma works without modification for augmented block partitions, so we have the following:

Lemma 3.8.13. *Let $\mathcal{H}$ be any family of functions $f : \mathbb{R}^d \to \{\pm 1\}$ such that, for any linear transformation $A : \mathbb{R}^d \to \mathbb{R}^d$, if $f \in \mathcal{H}$ then $f \circ A \in \mathcal{H}$. Let $\overline{\mathsf{block}} : \mathbb{R}^d \times [0,1]^d \to [r]^d$ be any augmented r-block partition. Let $\mathrm{ns}_{2,\delta}(\mathcal{H}) := \sup_{f \in \mathcal{H}} \mathsf{ns}_{2,\delta}(f)$. Then $\mathrm{ns}_{r,\delta}(f^{\mathsf{block}}) \leq \mathsf{ns}_{2,\delta}(\mathcal{H})$.*

3.8.3 Rounding the Output

After learning a function $g : [r]^d \to \{\pm 1\}$, we must output a function $g' : \mathbb{R}^d \to \{\pm 1\}$. In the continuous setting, we simply output $g \circ \mathsf{block}$. In the finite setting, we cannot simply output $g \circ \overline{\mathsf{block}}$ since $\overline{\mathsf{block}} : \mathbb{R}^d \times [0,1]^d \to [r]^d$ requires an additional argument $z \in [0,1]^d$. For example, if the distribution μ is a finitely supported distribution on $\{\pm 1\}^d$, then for each point $x \in \{\pm 1\}^d$ there may be roughly $(r/2)^d$ points $v \in [r]^d$ for which $(x, z) \in \overline{\mathsf{block}}^{-1}(v)$ for an appropriate choice of $z \in [0,1]^d$, and these points v may have different values in g. The algorithm must choose a single value to output for each x. We do so by approximating the function $x \mapsto \mathop{\mathbb{E}}_{z}[g_z(x)]$ and then rounding it via the next lemma.

Lemma 3.8.14. *Fix a domain $\mathcal{X}$, let $\gamma : \mathcal{X} \to [-1,1]$, and let $\epsilon > 0$. There is an algorithm such that, given query access to γ and sample access to any distribution $\mathcal{D}$ over $\mathcal{X} \times \{\pm 1\}$, uses at most $O\left(\log(1/\delta)/\epsilon^2\right)$ samples and queries and with probability at least $1-\delta$ produces a value t such that*

$$\mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[\mathrm{sign}(f(x) - t) \neq b] \leq \frac{1}{2} \mathop{\mathbb{E}}_{(x,b)\sim\mathcal{D}}[|f(x) - b|] + \epsilon\,.$$

Proof. Let $\mathcal{T}$ be the set of functions $x \mapsto \mathrm{sign}(\gamma(x) - t)$ for any choice of $t \in [-1,1]$. We will show that the VC dimension of $\mathcal{T}$ is 1. Suppose for contradiction that two points $x, y \in \mathcal{X}$ are shattered by $\mathcal{T}$, so in particular there are $s, t \in \mathbb{R}$ such that $\mathrm{sign}(f(x) - s) = 1$ and $\mathrm{sign}(f(y) - s) = -1$, while $\mathrm{sign}(f(x) - t) = -1$ and $\mathrm{sign}(f(y) - t) = 1$. Without loss of generality, suppose $s < t$. But then $\mathrm{sign}(f(y) - s) \geq \mathrm{sign}(f(y) - t)$, which is a contradiction. Therefore, by standard VC dimension arguments ([SB14], Theorem 6.8), using $O(\log(1/\delta)/\epsilon^2)$ samples and choosing t to minimize the error on the samples, with probability at least $1 - \delta$ we will obtain a value t such that

$$\mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[\mathrm{sign}(\gamma(x) - t) \neq b] \leq \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[\mathrm{sign}(\gamma(x) - t^*) \neq b] + \epsilon$$

where t^* minimizes the latter quantity among all values $[-1,1]$. Since the algorithm can restrict itself to those values $t \in [-1,1]$ for which $\gamma(x) = t$ for some x in the sample, the value minimizing the error on the sample can be computed time polynomial in the number of samples. Next, we show that the minimizer t^* satisfies the desired properties. Suppose that $t \sim [-1,1]$ uniformly at random. For any $y \in [-1,1], b \in \{\pm 1\}$,

$$\mathop{\mathbb{P}}_{t}[\mathrm{sign}(y-t) \neq b] = \begin{cases} \mathop{\mathbb{P}}_{t}[t > y] = \frac{1}{2}|b-y| & \text{if } b = 1 \\ \mathop{\mathbb{P}}_{t}[t \leq y] = \frac{1}{2}|y-b| & \text{if } b = -1\,. \end{cases}$$

Therefore

$$\mathop{\mathbb{E}}_{t\sim[-1,1]}\left[\mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[\mathrm{sign}(\gamma(x)-t) \neq b]\right] = \mathop{\mathbb{E}}_{(x,b)\sim\mathcal{D}}\left[\mathop{\mathbb{P}}_{t}[\mathrm{sign}(f(x)-t) \neq b]\right] = \frac{1}{2}\mathop{\mathbb{E}}_{(x,b)\sim\mathcal{D}}[|\gamma(x)-b|]\,,$$

so we can conclude the lemma with

$$\mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[\mathrm{sign}(\gamma(x)-t^*) \neq b] \leq \frac{1}{2}\mathop{\mathbb{E}}_{(x,b)\sim\mathcal{D}}[|\gamma(x)-b|]\,. \qquad \square$$

Lemma 3.8.15. *Let $\overline{\mathsf{block}} : \mathbb{R}^d \times [0,1]^d \to [r]^d$ be an augmented r-block partition. There is an algorithm which, given $\epsilon, \delta > 0$, query access to a function $g : [r]^d \to \{\pm 1\}$ and sample access to a distribution $\mathcal{D}$ over $\mathbb{R}^d \times \{\pm 1\}$, outputs a function $g' : \mathbb{R}^d \to \{\pm 1\}$ such that, with probability $1-\delta$,*

$$\mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[g'(x) \neq b] \leq \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}, z\sim[0,1]^d}\left[g(\overline{\mathsf{block}}(x,z)) \neq b\right] + \epsilon\,,$$

uses at most $O\left(\frac{d\log(r)}{\epsilon^2}\log\frac{1}{\delta}\right)$ samples and queries, and runs in time polynomial in the number of samples.

Proof. For $z \in [0,1]^d$, write $g_z(x) = g(\overline{\mathsf{block}}(x,z))$. For any (x,b),

$$|\mathop{\mathbb{E}}_{z}[g_z(x)] - b| = |b\mathop{\mathbb{P}}_{z}[g_z(x) = b] - b\mathop{\mathbb{P}}_{z}[g_z(x) \neq b] - b| = |-2b\mathop{\mathbb{P}}_{z}[g_z(x) \neq b]\,| = 2\mathop{\mathbb{P}}_{z}[g_z(x) \neq b]\,,$$

so

$$\frac{1}{2}\mathop{\mathbb{E}}_{(x,b)\sim\mathcal{D}}\left[|\mathop{\mathbb{E}}_{z}[g_z(x)] - b|\right] = \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D},z}[g_z(x) \neq b]\,.$$

The algorithm will construct a function $\gamma(x) \approx \mathop{\mathbb{E}}_{z}[g_z(x)]$ and then learn a suitable parameter t for rounding $\gamma(x)$ to $\mathrm{sign}(\gamma(x)-t)$.

First the algorithm samples a set $Z \subset [0,1]^d$ of size $m = \frac{2d\ln(r)\ln(1/\delta)}{\epsilon^2}$ and construct the function $\gamma(x) = \frac{1}{m}\sum_{z\in Z} g(\overline{\mathsf{block}}(x,z))$. Fix $Z \subset [0,1]^d$ and suppose $x \in \mathbb{R}^d$ satisfies $\gamma(x) \neq \mathop{\mathbb{E}}_z[g_z(x)]$. Then there must be $w, z \in [0,1]^d$ such that $\overline{\mathsf{block}}(x,z) \neq \overline{\mathsf{block}}(x,w)$, otherwise $g_z(x) = g_w(x)$ for all z, w so for all $w, \gamma(x) = g_w(x) = \mathop{\mathbb{E}}_z[g_z(x)]$. There can be at most r^d values of $x \in \mathbb{R}^d$ for which $\exists z, w \in [0,1]^d : \overline{\mathsf{block}}(x,z) \neq \overline{\mathsf{block}}(x,w)$, so by the union bound and the Hoeffding bound,

$$\begin{aligned}\mathop{\mathbb{P}}_Z\left[\exists x \in \mathbb{R}^d : |\gamma(x) - \mathop{\mathbb{E}}_z[g_z(x)]\,| > \epsilon\right] &\leq r^d \max_{x\in X} \mathop{\mathbb{P}}_Z\left[|\gamma(x) - \mathop{\mathbb{E}}_z[g_z(x)]\,| > \epsilon\right] \\ &\leq 2r^d \exp\left(-\frac{m\epsilon^2}{2}\right) < \delta\,.\end{aligned}$$

Therefore with probability at least $1 - \delta/2$, γ satisfies $|\gamma(x) - \mathop{\mathbb{E}}_z[g_z(x)]\,| \leq \epsilon$ for all x. Suppose this occurs. Then

$$\begin{aligned}\frac{1}{2}\mathop{\mathbb{E}}_{(x,b)\sim\mathcal{D}}[|\gamma(x) - b|] &\leq \frac{1}{2}\mathop{\mathbb{E}}_{(x,b)\sim\mathcal{D}}\left[|\mathop{\mathbb{E}}_z[g_z(x)] - b| + |\gamma(x) - \mathop{\mathbb{E}}_z[g_z(x)]\,|\right] \\ &\leq \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D},z}[g_z(x) \neq b] + \frac{\epsilon}{2}\,.\end{aligned}$$

Now we apply Lemma 3.8.14 with error $\epsilon/2$, using $O(\log(1/\delta)/\epsilon^2)$ samples and polynomial time, to output a value t such that with probability $1 - \delta/2$,

$$\mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[\mathrm{sign}(\gamma(x) - t) \neq b] \leq \frac{1}{2}\mathop{\mathbb{E}}_{(x,b)}[|\gamma(x) - b|] + \frac{\epsilon}{2} \leq \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D},z}[g_z(x) \neq b] + \epsilon\,. \qquad \square$$

3.8.4 Algorithms for Finite Distributions

We now state improved versions of our monotonicity tester and two general learning algorithms: the "brute force" learning algorithm (Lemma 3.4.8) and the "polynomial regression" algorithm (Lemma 3.5.2). Using these algorithms we obtain the same complexity bounds as for continuous product distributions, but the algorithms can now handle finite product distributions as well.

Theorem 1.2.11. *There is a one-sided non-adaptive tester for monotonicity under product distributions over $\mathbb{R}^d$, with query complexity $\widetilde{O}(d^{5/6}/\epsilon^{4/3})$ and sample complexity $\widetilde{O}((d/\epsilon)^3)$.*

Proof. The proof of Theorem 1.2.11 goes through as before, with block replaced by $\overline{\mathsf{block}}$, $\mathsf{block}^{-\downarrow}(v)$ replaced with $\overline{\mathsf{block}}^{-\downarrow}(v)$ defined as the infimal element $\overline{x}$ such that $\overline{\mathsf{block}}(\overline{x}) = v$,

and $\mathsf{block}^{-\uparrow}(v)$ defined as the supremal element $\overline{x}$ such that $\overline{\mathsf{block}}(\overline{x}) = v$, and g redefined appropriately. □

Next, we move on to the learning algorithms:

Lemma 3.8.16. *Let $\mathcal{H}$ be any set of functions $\mathbb{R}^d \to \{\pm 1\}$, let $\epsilon > 0$, and suppose r satisfies $r^{-d} \cdot \overline{\mathsf{bbs}}(\mathcal{H}, r) \leq \epsilon/3$. Then there is an agnostic learning algorithm for $\mathcal{H}$ that uses $O\left(\frac{r^d + rd^2 \log(rd/\epsilon)}{\epsilon^2}\right)$ samples and time and works for any distribution $\mathcal{D}$ over $\mathbb{R}^d \times \{\pm 1\}$ whose marginal on $\mathbb{R}^d$ is a finite or continuous product distribution.*

Proof. On input distribution $\mathcal{D}$:

1. Sample a grid $\overline{X}$ of size $m = O(\frac{rd^2}{\epsilon^2} \log(rd/\epsilon))$ large enough that Lemma 3.8.6 guarantees $\|\overline{\mathsf{block}}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} < \epsilon/3$ with probability $5/6$, where $\overline{\mathsf{block}} : \mathbb{R}^d \times [0,1]^d \to [r]^d$ is the induced augmented r-block partition.
2. Agnostically learn a function $g : [r]^d \to \{\pm 1\}$ with error $\epsilon/3$ and success probability $5/6$ using $O(r^d/\epsilon^2)$ samples from $\mathcal{D}^{\mathsf{block}}$.
3. Run the algorithm of Lemma 3.8.14 using $O\left(\frac{d \log r}{\epsilon^2}\right)$ samples to obtain g' and output g'.

The proof proceeds as in the case for continuous distributions (Lemma 3.4.8). Assume all steps succeed, which occurs with probability at least $2/3$. After step 3 we obtain $g : [r]^d \to \{\pm 1\}$ such that, for any $h \in \mathcal{H}$,

$$\mathop{\mathbb{P}}_{(v,b)\sim\mathcal{D}^{\mathsf{block}}}[g(v) \neq b] \leq \mathop{\mathbb{P}}_{(v,b)\sim\mathcal{D}^{\mathsf{block}}}\left[h^{\mathsf{block}}(v) \neq b\right] + \epsilon/3\,.$$

By Lemma 3.8.14 and Proposition 3.8.7, the output satisfies,

$$\begin{aligned}
\mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[g'(x) \neq b] &\leq \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D},z}\left[g(\overline{\mathsf{block}}(x,z)) \neq b\right] + \epsilon/3 \\
&\leq \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D},z}\left[h^{\mathsf{block}}(\overline{\mathsf{block}}(x,z)) \neq b\right] + 2\epsilon/3 \\
&\leq \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[h(x) \neq b] + \mathop{\mathbb{P}}_{x,z}[h(x) \neq h_z^{\mathsf{coarse}}(x)] + 2\epsilon/3 \\
&\leq \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[h(x) \neq b] + \epsilon\,.
\end{aligned}$$

□

We now state the general learning algorithm from Lemma 3.5.2, improved to allow finite product distributions.

Lemma 3.8.17. *Let $\epsilon > 0$ and let $\mathcal{H}$ be a set of measurable functions $f : \mathbb{R}^d \to \{\pm 1\}$ that satisfy:*

1. *There is some $r = r(d, \epsilon)$ such that $\overline{\mathsf{bbs}}(\mathcal{H}, r) \leq \epsilon \cdot r^d$;*
2. *There is a set $\mathcal{F}$ of functions $[r]^d \to \mathbb{R}$ satisfying: $\forall f \in \mathcal{H}, \exists g \in \mathsf{span}(\mathcal{F})$ such that for $v \sim [r]^d, \mathbb{E}\left[(f^{\mathsf{block}}(v) - g(v))^2\right] \leq \epsilon^2$.*

Let $n = \mathrm{poly}(|\mathcal{F}|, 1/\epsilon)$ be the sample complexity of the algorithm in Theorem 3.5.1. Then there is an agnostic learning algorithm for $\mathcal{H}$ on finite and continuous product distributions over $\mathbb{R}^d$, that uses $O(\max(n^2, 1/\epsilon^2) \cdot rd^2 \log(dr))$ samples and runs in time polynomial in the sample size.

Proof. Let $\overline{\mathcal{D}}$ be the augmented distribution, where $\overline{x} \sim \overline{\mathcal{D}}$ is obtained by drawing $x \sim \mathcal{D}$ and augmenting it with a uniformly random $z \in [0, 1]^d$. We will assume $n > 1/\epsilon$. Let μ be the marginal of $\mathcal{D}$ on $\mathbb{R}^d$. For an augmented r-block partition, let $\mathcal{D}^{\mathsf{block}}$ be the distribution of $(\overline{\mathsf{block}}(\overline{x}), b)$ when $(\overline{x}, b) \sim \overline{\mathcal{D}}$. We may simulate samples from $\mathcal{D}^{\mathsf{block}}$ by sampling (x, b) from $\mathcal{D}$ and constructing $(\overline{\mathsf{block}}(\overline{x}), b)$. The algorithm is as follows:

1. Sample a grid X of length $m = O(rd^2n^2 \log(rd))$; by Lemma 3.8.6, this ensures that $\|\overline{\mathsf{block}}(\mu) - \mathsf{unif}([r]^d)\|_{\mathsf{TV}} < 1/12n$ with probability $5/6$. Construct $\overline{\mathsf{block}} : \mathbb{R}^d \to [r]^d$ induced by X.
2. Run the algorithm of Theorem 3.5.1 on a sample of n points from $\mathcal{D}^{\mathsf{block}}$; that algorithm returns a function $g : [r]^d \to \{\pm 1\}$.
3. Run the algorithm of Lemma 3.8.14 using $O\left(\frac{d \log r}{\epsilon^2}\right)$ samples to obtain g' and output g'.

The proof proceeds as in the case for continuous distributions (Lemma 3.5.2). Assume all steps succeed, which occurs with probability at least $2/3$. After step 3 we obtain $g : [r]^d \to \{\pm 1\}$ such that, for any $h \in \mathcal{H}$,

$$\mathop{\mathbb{P}}_{(v,b) \sim \mathcal{D}^{\mathsf{block}}}[g(v) \neq b] \leq \mathop{\mathbb{P}}_{(v,b) \sim \mathcal{D}^{\mathsf{block}}}\left[h^{\mathsf{block}}(v) \neq b\right] + \epsilon/3\,.$$

By Lemma 3.8.14 and Proposition 3.8.7, the output satisfies,

$$\begin{aligned}
\mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[g'(x) \neq b] &\leq \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D},z}\left[g(\overline{\mathsf{block}}(x,z)) \neq b\right] + \epsilon/3 \\
&\leq \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D},z}\left[h^{\mathsf{block}}(\overline{\mathsf{block}}(x,z)) \neq b\right] + 2\epsilon/3 \\
&\leq \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[h(x) \neq b] + \mathop{\mathbb{P}}_{x,z}[h(x) \neq h_z^{\mathsf{coarse}}(x)] + 2\epsilon/3 \\
&\leq \mathop{\mathbb{P}}_{(x,b)\sim\mathcal{D}}[h(x) \neq b] + \epsilon\,. \qquad \square
\end{aligned}$$

Theorem 3.8.18. *There are agnostic learning algorithms for functions of convex sets, functions of halfspaces, degree-k PTFs, and k-alternating functions achieving the sample and time complexity bounds in Theorems 1.2.12, 1.2.13, 3.4.4 and 3.7.1, that work for any finite or continuous product distribution over $\mathbb{R}^d$.*

Proof. This follows from the same arguments as for each of those theorems, except with the bounds from Proposition 3.8.8 and Lemmas 3.8.9 to 3.8.12 to bound $\overline{\mathsf{bbs}}$; Lemma 3.8.13 to bound the noise sensitivity; and the improved general algorithms of Lemmas 3.8.16 and 3.8.17. $\square$

Chapter 4

Sketching Adjacency and Distances in Graphs

The people wander across the plane,
searching for each other in vain.
Praying to their god divine,
"Am I near the friends of mine?"
they discover to their great surprise,
their prayers are of constant size!

Here begins Part II, where we introduce adjacency and distance sketching in hereditary classes of graphs. More generally, we introduce f-sketching where f is a partial function, parameterized by a graph G, that takes two vertices as arguments. The main examples are adjacency, exact distance thresholds, approximate distance thresholds, and certain first-order formulas, introduced in Section 4.2.1. We begin to develop a theory of the graph classes which admit *constant-size* f-sketches; we call these classes *f-sketchable*. This chapter introduces the basic facts about sketching, which are taken from [Har20, HWZ22]. The main three important facts about these sketches are:

1. Constant-cost randomized communication problems are equivalent to hereditary graph classes that are adjacency sketchable. Therefore we may attempt to develop a theory of constant-cost communication from a structural graph theory perspective by studying adjacency sketching. See Section 4.2.2.
2. Any adjacency sketchable class admits a constant-size *probabilistic universal graph (PUG)*, which is a probabilistic variant of an *induced-universal graph*, as introduced

by Rado [Rad64]. This is in parallel with the equivalence between adjacency labeling and induced-universal graphs observed by Kannan, Naor, & Rudich. See Section 4.2.3.

3. Any f-sketchable class admits a deterministic *f-labeling scheme* with labels of size $O(\log n)$. In particular, adjacency sketchable classes admit *adjacency labeling schemes* of size $O(\log n)$, the subject of the main open problem in the area. See Sections 4.2.4 and 4.3.

There is a particular category of f-sketch that we call special attention to: the *equality-based* sketch, which we define in Section 4.2.5, and which has a special relationship to items (1) and (3). A constant-size equality-based f sketch is one that corresponds to a randomized constant-cost communication protocol that can be simulated by a constant-cost *deterministic* protocol with access to a unit-cost EQUALITY oracle. Equality-based sketches also capture a common type of labeling scheme, including the original adjacency labeling schemes for bounded arboricity graphs in [KNR92]. We will use these types of sketches extensively, and we show some of their limitations in Chapter 5.

As a consequence of item (1), constant-cost communication problems correspond to a certain subset of hereditary graph classes. Hereditary graph classes form a *lattice*, whose structure has been explored thoroughly in the literature. In Section 4.3 we give an overview of this literature and give some simple results that place constant-cost communication and the adjacency sketchable classes in this context.

As a warm-up, we begin with an extremely simple proof of an adjacency sketch for the induced subgraphs of the hypercube and some of its consequences: the first adjacency labeling scheme for the induced subgraphs of the hypercube, and the first non-trivial upper bounds on the number of unique subgraphs and induced subgraphs of the hypercube. We will extend these ideas in Chapter 5.

4.1 Warm-Up: The Hypercube

Recall from Chapter 1 that for any set of graphs $\mathcal{F}$ we define $\mathsf{mon}(\mathcal{F})$ as the *monotone closure* of $\mathcal{F}$, which is the set of subgraphs of graphs $G \in \mathcal{F}$, and $\mathsf{her}(\mathcal{F})$ as the *hereditary closure*, which is the set of induced subgraphs of graphs $G \in \mathcal{F}$. We have also defined

$$\mathcal{F}^{\square} = \{G_1 \square \cdots \square G_d : d \in \mathbb{N}, G_i \in \mathcal{F}\}.$$

The hypercube is the d-wise Cartesian product K_2^d of the single edge K_2, and we write $\mathcal{H} = \mathsf{her}(\{K_2\}^{\square})$ for the class of induced subgraphs of hypercubes.

For a class of graphs $\mathcal{F}$, a one-sided error adjacency sketch with size s is a function $D : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$ such that, for every graph $G \in \mathcal{F}$, there is a random function $\mathsf{sk} : V(G) \to \{0,1\}^s$ satisfying

$$\begin{aligned} \forall x, y \in V(G): \qquad xy \in E(G) &\implies \mathop{\mathbb{P}}_{\mathsf{sk}}[D(\mathsf{sk}(x), \mathsf{sk}(y)) = 1] = 1 \\ xy \notin E(G) &\implies \mathop{\mathbb{P}}_{\mathsf{sk}}[D(\mathsf{sk}(x), \mathsf{sk}(y)) = 0] \geq 1/4\,. \end{aligned}$$

The value $1/4$ is chosen for convenience in the following proof.

Theorem 4.1.1. *$\mathcal{H}$ has a one-sided error adjacency sketch of size 4.*

Proof. For any $G \in \mathsf{her}(\{K_2\}^{\square})$, we may identify the vertices V of G with a subset binary strings in $\{0,1\}^d$, such that two vertices $x, y \in V \subseteq \{0,1\}^d$ are adjacent if and only if they differ on one coordinate.

Define $D : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$ such that $D(u,v) = 1$ if and only if the vector $w = u \oplus v$ has Hamming weight 1 (where $u \oplus v$ is the coordinate-wise XOR). For a graph $G \in \mathcal{H}$ on vertex set $V \subseteq \{0,1\}^d$, choose a random function $\mathsf{sk} : \{0,1\}^d \to \{0,1\}^4$ by choosing a uniformly random map $p : [d] \to [4]$. For each $i \in [4]$, define

$$\mathsf{sk}(x)_i := \bigoplus_{j \in p^{-1}(i)} x_j\,.$$

Suppose that $xy \in E(G)$ so that they differ on exactly one coordinate. For any choice of $p : [d] \to [4]$, we may partition the coordinates into four sets $A_i = p^{-1}(i)$. One of these sets A_i contains the differing coordinate and will have $w_i = 1$, while the other three sets A_j will have $w_j = 0$, so

$$\mathop{\mathbb{P}}_{\mathsf{sk}}[D(\mathsf{sk}(x), \mathsf{sk}(y)) = 1] = 1\,.$$

Now suppose $xy \notin E(G)$. If $x = y$ then we have $\mathsf{sk}(x) = \mathsf{sk}(y)$ so $D(\mathsf{sk}(x), \mathsf{sk}(y)) = 0$ with probability 1. Now suppose that x, y differ on $t \geq 2$ coordinates. We will show that w has Hamming weight 1 with probability at most $3/4$. Note that w is obtained by the random process where $\vec{0} = w^{(0)}, w = w^{(t)}$, and $w^{(i)}$ is obtained from $w^{(i-1)}$ by flipping a uniformly random coordinate.

Observe that, for $i \geq 1$, $\mathbb{P}\left[w^{(i)} = \vec{0}\right] \leq 1/4$. This is because $w^{(i)} = \vec{0}$ can occur only if $w^{(i-1)}$ has Hamming weight 1, so the probability of flipping the 1-valued coordinate is $1/4$.

Then, for $t \geq 2$, we have

$$\begin{aligned}\mathbb{P}\left[|w^{(t)}| = 1\right] &= \mathbb{E}_{w^{(t-1)}}\left[\mathbb{P}\left[|w^{(t)}| = 1 \;\middle|\; w^{(t-1)}\right]\right] \\ &= \mathbb{P}\left[w^{(t-1)} = \vec{0}\right] + \mathbb{P}\left[w^{(t-1)} \neq \vec{0}\right] \cdot \mathbb{P}\left[|w^{(t)}| = 1 \;\middle|\; w^{(t-1)} \neq \vec{0}\right] \\ &\leq \frac{1}{4} + \frac{1}{2} = \frac{3}{4}.\end{aligned}$$

□

It is interesting to note that the decoder of this sketch simply checks adjacency in K_2^4. So, all induced subgraphs of hypercubes can be "probabilistically embedded" in K_2^4. This is defined formally in Definition 4.2.5.

Some consequences of this theorem are as follows.

Theorem 4.1.2. *There exists an adjacency labeling scheme for $\mathcal{H}$ of size $O(\log n)$.*

Proof. By the probabilistic method. For a graph $G \in \mathcal{H}$ on n vertices, choose $k = 5\lceil \log n \rceil$, and let $\mathsf{sk}^{(1)}, \ldots, \mathsf{sk}^{(k)} : V(G) \to \{0,1\}^4$, be independently randomly chosen as above. To each $x \in V(G)$, assign a random label with size $4k$ as follows:

$$\mathsf{sk}(x) = \left(\mathsf{sk}^{(1)}(x), \ldots, \mathsf{sk}^{(k)}(x)\right),$$

Let D' be the decoder defined for the adjacency sketch above. On a pair of inputs $\mathsf{sk}(x), \mathsf{sk}(y)$, the decoder will output $D(\mathsf{sk}(x), \mathsf{sk}(y)) = 1$ if $D'(\mathsf{sk}^{(i)}(x), \mathsf{sk}^{(i)}(y)) = 1$ for all $i \in [k]$. If x, y are adjacent, we have from above that $\mathbb{P}\left[D(\mathsf{sk}(x), \mathsf{sk}(y)) = 1\right] = 1$. Otherwise, we have

$$\mathbb{P}\left[D(\mathsf{sk}(x), \mathsf{sk}(y)) = 1\right] = \mathbb{P}\left[\forall i \in [k] : D'(\mathsf{sk}^{(i)}(x), \mathsf{sk}^{(i)}(y)) = 1\right] \leq (3/4)^k$$

By the union bound, the probability that there exist two non-adjacent vertices $x, y \in V(G)$ such that $D(\mathsf{sk}(x), \mathsf{sk}(y)) = 1$ is at most

$$\frac{n(n-1)}{2} \cdot \mathbb{P}\left[D(\mathsf{sk}(x), \mathsf{sk}(y)) = 1\right] < \frac{n(n-1)}{2}\left(\frac{3}{4}\right)^k < 1$$

Therefore, for every G on n vertices, there exists a deterministic function $\mathsf{sk} : V(G) \to \{0,1\}^{4k}$ such that $D(\mathsf{sk}(x), \mathsf{sk}(y))$ is correct for all pairs $x, y \in V(G)$. □

An immediate consequence is an upper bound on the number of unique induced subgraphs of the hypercube.

Corollary 4.1.3. *The number of n-vertex induced subgraphs of a hypercube is at most $2^{O(n \log n)}$.*

As discussed in Chapter 1, the above results resolve two open questions of Alecu, Atminas, & Lozin [AAL21]. A further consequence is a bound on the number of subgraphs. The best known bound was previously $2^{O(n \log^2 n)}$ (see e.g. [CLR20]).

Corollary 4.1.4. *The number of n-vertex graphs subgraphs of a hypercube is at most $2^{O(n \log n)}$.*

Proof. The number of edges of an n-vertex subgraph of a hypercube is at most $n \log n$ [Gra70] (or see below), so every n-vertex subgraph of a hypercube has at most $2^{n \log n}$ spanning subgraphs. Every subgraph of the hypercube is a spanning subgraph of an induced subgraph, so the number of subgraphs is at most $2^{n \log n} \cdot 2^{O(n \log n)}$. □

Here is a simple proof of the upper bound on the number of edges in a subgraph of the hypercube, which we present for the sake of completeness of this warm-up.

Proposition 4.1.5. *Any n-vertex subgraph of a hypercube has at most $n \log n$ edges.*

Proof. By induction on n. The claim holds trivially for $n = 1$. For $n > 1$, assume that G is a subgraph of Q^d with vertex set $V(G) \subseteq \{0,1\}^d$. Let G_0 be the induced subgraph on vertices $x \in V(G)$ with $x_1 = 0$, and G_1 be the induced subgraph on vertices $x \in V(G)$ with $x_1 = 1$. We may assume that G_0, G_1 are non-empty (otherwise G is a subgraph of Q^{d-1}). Write $n = n_0 + n_1$ where $n_0 = |V(G_0)|$, $n_1 = |V(G_1)|$. Assume without loss of generality that $n_0 \leq n_1$, and observe that the number of edges in G between $V(G_0)$ and $V(G_1)$ is at most $\min(n_0, n_1) = n_0$, since these edges form a matching between $V(G_0)$ and $V(G_1)$. Then, by induction, we have

$$\begin{aligned} |E(G)| &\leq |E(G_0)| + |E(G_1)| + n_0 \leq n_0 \log(n_0) + n_1 \log(n_1) + n_0 \\ &= n_0 \log(2n_0) + n_1 \log(n_1) \leq n_0 \log(n_0 + n_1) + n_1 \log(n_0 + n_1) \\ &= (n_0 + n_1) \log(n_0 + n_1) = n \log n \,. \end{aligned}$$

□

4.1.1 Notation and Terminology

All graphs in this work are simple, i.e. undirected, without loops and multiple edges. A *class* of graphs $\mathcal{F}$ is a set of graphs closed under isomorphism, where we assume that an

n-vertex graph $G \in \mathcal{F}$ has vertex set $[n]$. We write $\mathcal{F}_n$ for the set of graphs $G \in \mathcal{F}_n$ on vertex set $[n]$. The *speed* of a class $\mathcal{F}$ is the function $n \mapsto |\mathcal{F}_n|$.

Let G be a graph and let v be a vertex in G. A vertex that is adjacent to v is called a *neighbour* of v. The set of all neighbours of v is called the *neighbourhood* of v and it is denoted as $N(v)$. The *degree* of v is the number of neighbours of v and it is denoted as $\deg(v)$. A bipartite graph is a graph whose vertex set can be partitioned into two independent sets. A *colored* bipartite graph is a bipartite graph with a given bipartition of its vertex set. We denote a colored bipartite graph by a triple (X, Y, E), where X, Y is the partition of its vertex set into two parts, and the function $E : X \times Y \to \{0, 1\}$ defines the edge relation. If a bipartite graph G is connected, it has a unique partition of its vertices into two parts and therefore there is only one colored bipartite graph corresponding to G; (note that (X, Y, E) and (Y, X, E) are considered the same colored bipartite graph). If G is disconnected, however, there is more than one corresponding colored bipartite graph.

For colored bipartite graphs $G = (X, Y, E)$ and $H = (X', Y', E')$, we say that H is an induced subgraph of G, and write $H \sqsubset G$, when there is an injective map $\phi : X' \cup Y' \to X \cup Y$ that preserves adjacency and preserves parts. The latter means that the images $\phi(X')$ and $\phi(Y')$ satisfy either $\phi(X') \subseteq X, \phi(Y') \subseteq Y$ or $\phi(X') \subseteq Y, \phi(Y') \subseteq X$. A colored bipartite graph $G = (X, Y, E)$ is called *biclique* if every vertex in X is adjacent to every vertex in Y, and G is called *co-biclique* if $E = \emptyset$.

For any graph $G = (V, E)$ and subset $W \subseteq V$, we write $G[W]$ for the subgraph of G induced by W. For disjoint sets $X, Y \subseteq V$, we write $G[X, Y]$ for the colored bipartite graph (X, Y, E') where for $(x, y) \in X \times Y$, $(x, y) \in E'$ if and only if $(x, y) \in E$.

We also write $\overline{G}$ for the graph complement of G, i.e. the graph $(V, \overline{E})$ where $(x, y) \in \overline{E}$ if and only if $(x, y) \notin E$. The *bipartite* complement, $\overline{\overline{G}}$, of a colored bipartite graph $G = (X, Y, E)$ is the graph $\overline{\overline{G}} = (X, Y, \overline{\overline{E}})$ with $(x, y) \in \overline{\overline{E}}$ if and only if $(x, y) \notin E$ for $x \in X$, $y \in Y$.

The *disjoint union* of two graphs $G = (V, E)$ and $H = (V', E')$ is the graph $G + H = (V \cup V', E \cup E')$.

4.2 Graph Sketching: The Basics

We will define a general notion of sketching. For a class $\mathcal{F}$ of graphs, we will write f to refer to a set $\{f_G\}_{G \in \mathcal{F}}$ of partial functions $f_G : V(G) \times V(G) \to \{0, 1, *\}$ indexed by the graphs $G \in \mathcal{F}$. For example, f could be the set of adjacency functions where $f_G(x, y) = 1$

if and only if x, y are adjacent in G, and $f_G(x, y) = 0$ otherwise. Or f could be a "gap" distance function where $f_G(x, y) = 1$ if $\mathsf{dist}_G(x, y) \leq k$, $f_G(x, y) = 0$ if $\mathsf{dist}_G(x, y) > 2k$, and $f_G(x, y) = *$ otherwise.

For a graph class $\mathcal{F}$ and $\delta > 0$, we define an *f-sketch* with error δ for $\mathcal{F}$ as a decoder $D : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$, such that for every $G \in \mathcal{F}$ the following holds. There is a probability distribution over functions $\mathsf{sk} : V(G) \to \{0,1\}^*$, such that for all $x, y \in V(G)$,

$$f_G(x, y) \neq * \implies \mathbb{P}\left[D(\mathsf{sk}(x), \mathsf{sk}(y)) = f_G(x, y)\right] \geq 1 - \delta$$

We define the *size* of the sketch as

$$\max_{G \in \mathcal{F}_n} \sup_{\mathsf{sk}} \max_{x \in V(G)} |\mathsf{sk}(x)|\,,$$

where the supremum is over the set of functions $\mathsf{sk} : V(G) \to \{0,1\}^*$ in the support of the distribution defined for G, and $|\mathsf{sk}(x)|$ is the number of bits of $\mathsf{sk}(x)$. We will refer to f-sketches with error $1/3$ as *f-sketches*, and will say that a class $\mathcal{F}$ is *f-sketchable* if there exists an f-sketch for $\mathcal{F}$ with size that does not depend on the number of vertices n.

For a graph class $\mathcal{F}$, we also define an *f-labeling scheme* for $\mathcal{F}$ similar to above, except that for every $G \in \mathcal{F}$ there is a *deterministic* function $\ell : V(G) \to \{0,1\}^*$ such that for all $x, y \in V(G)$,

$$f_G(x, y) \neq * \implies D(\ell(x), \ell(y)) = f_G(x, y)\,.$$

4.2.1 Types of Sketches: Adjacency, Distance, and First-Order

Given a graph G, the length of a path P in G is the number of edges of P. Given two vertices $x, y \in V(G)$, we define $\mathsf{dist}_G(x, y)$ to be the infimum of the length of a path between x and y in G; we define $\mathsf{dist}_G(x, y) = \infty$ if there exists no path between x and y. Notice that $(V(G), \mathsf{dist}_G)$ is a metric space (with possibly infinite distances between pairs of vertices if G is disconnected).

We now define certain important types of f-sketches. Let $\mathcal{F}$ be a class of graphs. For any $r_1 \leq r_2$, a *distance-(r_1, r_2) sketch* for $\mathcal{F}$ is an f-sketch, as defined above, when for any graph G we define the function

$$f_G(x, y) = \begin{cases} 1 & \text{if } \mathsf{dist}_G(x, y) \leq r_1 \\ 0 & \text{if } \mathsf{dist}_G(x, y) > r_2 \\ * & \text{otherwise.} \end{cases}$$

The size of such a sketch may depend on r_1, r_2, the number of vertices n, or other graph parameters. In Chapter 1, we defined *adjacency*, *small-distance*, and *approximate distance threshold (ADT) sketching*. We redefine these sketching problems here, using the above formulation:

1. A class $\mathcal{F}$ is *adjacency sketchable* if it is distance-$(1,1)$ sketchable;
2. A class $\mathcal{F}$ is *small-distance sketchable* if for every $r \geq 1$ it is distance-(r,r) sketchable. For simplicity, we will write *distance-r sketch* instead of distance-(r,r) sketch.
3. A class $\mathcal{F}$ is *α-ADT sketchable* if for every $r \geq 1$ it is distance-$(r,\alpha r)$ sketchable, and furthermore the size of the sketch does not depend on r.

We will also define *first-order (FO) sketchable* classes, for which we require some terminology. A *relational vocabulary* Σ is a set of relation symbols, with each $R \in \Sigma$ having an *arity* $\text{arity}(R) \in \mathbb{N} \setminus \{0\}$. A Σ-structure $\mathcal{A}$ consists of a *domain* A, and for each relation symbol $R \in \Sigma$ an *interpretation* $R^{\mathcal{A}} \subseteq A^{\text{arity}(R)}$, which is a relation. Fix a countably infinite set X of *variables*. *Atomic formulas of vocabulary* Σ are of the form

- $x = y$ for $x, y \in X$; or,
- $R(x_1, \ldots, x_r)$ for $x_1, \ldots, x_r \in X$, $R \in \Sigma$ and $r = \text{arity}(R)$, which evaluates to true when $(x_1, \ldots, x_r) \in R$.

First-order (FO) formulas of vocabulary Σ are inductively defined as either atomic formulas, or a formula of the form $\neg\phi, \phi \wedge \psi, \phi \vee \psi$, or $\exists x.\phi$ or $\forall x.\psi$, where ϕ and ψ are each FO formulas. A *free variable* of a formula ϕ is one which is not bound by a quantifier. We will write $\phi(x_1, x_2, \ldots, x_k)$ to show that the free variables of ϕ are $x_1, \ldots, x_k \in X$. For a value $u \in A$, we write $\phi[u/x]$ for the formula obtained by substituting the constant u for the free variable x.

Let $\phi(x, y)$ be any formula with two free variables and relational vocabulary $\Sigma = \{E', R_1, \ldots, R_k\}$ where E' is symmetric of arity 2 and each R_i has arity 1. We will say that a graph class $\mathcal{F}$ is *ϕ-sketchable* if it is f-sketchable for any f chosen as follows. For any graph $G = (V, E)$, we choose any Σ-structure with domain V where E is the interpretation of the symbol E'. Then set $f_G(u, v) = 1$ if and only if $\phi(u/x, v/y)$ evaluates to true.

We remark that for any graph G, there are many ways to choose a Σ-structure with domain V with E being the interpretation of E'. To be first-order sketchable, a class $\mathcal{F}$

must be f-sketchable for *every* such choice of functions f_G. A concrete example is that, for any $r \in \mathbb{N}$, we can choose the formula

$$\phi(x,y) = \exists u_1, u_2, \ldots, u_{r-1} : \\ (E'(x,u_1) \vee x = u_1) \wedge (E'(u_1,u_2) \vee u_1 = u_2) \wedge \cdots \wedge (E'(u_r,y) \vee u_r = y)\,,$$

which evaluates to true if and only if $\mathsf{dist}_G(x,y) \leq r$. So any class that is first-order sketchable must be small-distance sketchable.

4.2.2 Communication Complexity

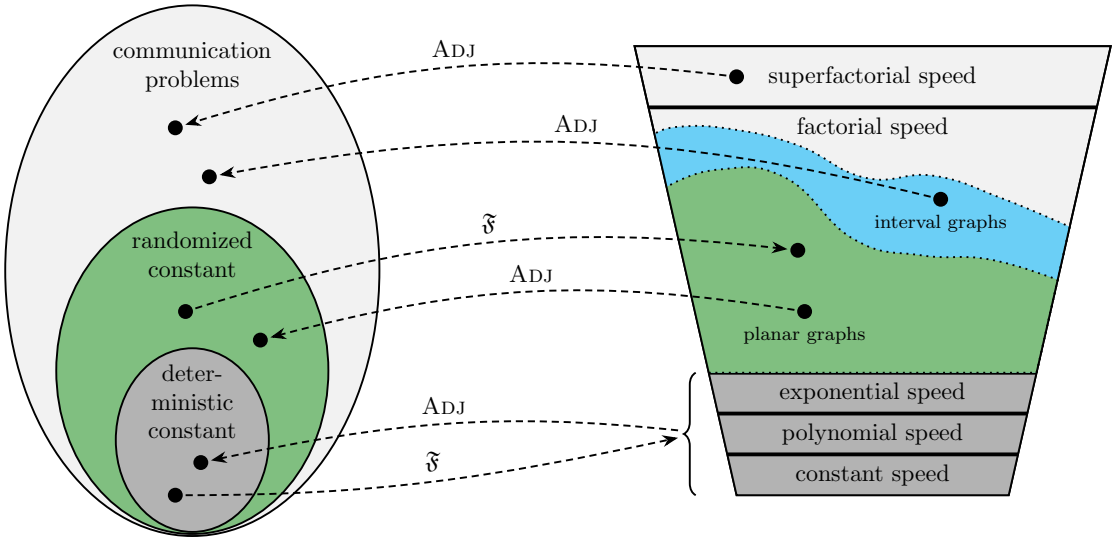


Figure 4.1: The communication-to-graph correspondence. Section 4.3 describes the lattice on the right. Communication problems with constant-cost randomized protocols (green) are mapped to the set of hereditary graph classes with constant-size adjacency sketches (green) by $\mathfrak{F}$. This is a subset of the classes that admit $O(\log n)$-size adjacency labels (blue). Adjacency sketchable classes are mapped to constant-cost communication problems by Adj.

We now establish the *communication-to-graph* correspondence mentioned in Chapter 1, Section 1.3. We refer the reader to [NK96, RY20] for an introduction to communication complexity. A *communication problem* is a sequence $f = (f_n)_{n\in\mathbb{N}}$ of functions[1] $f_n : [n] \times$

[1]In the literature, the domain is usually $\{0,1\}^n \times \{0,1\}^n$. We use $[n] \times [n]$ to highlight the graph interpretation.

$[n] \to \{0,1\}$. For any communication problem $f = (f_n)_{n\in\mathbb{N}}$, write $\mathsf{CC}(f_n)$ for the cost of the optimal two-way, randomized protocol (Definition 4.2.1) computing f_n, and write $\mathsf{CC}(f)$ for the function $n \mapsto \mathsf{CC}(f_n)$. We may represent f_n as a bipartite graph $F_n = ([n], [n], f_n)$ where f_n defines the edge relation.

For a function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$, we will write $\mathsf{CC}(f)$ for the optimal two-way, public-coin randomized communication cost of f. Informally, in this model, the players Alice and Bob share a source of randomness. Alice receives input x, Bob receives input y, and they communicate by sending messages back and forth using their shared randomness. After communication, Bob must output a (random) value $b \in \{0,1\}$ such that $b = f_n(x,y)$ with probability at least $2/3$. The cost of such a protocol is the maximum over all inputs of the number of bits communicated between the players. Formally, the definition is as follows.

Definition 4.2.1. A two-way public-coin communication protocol is a probability distribution over *communication trees*. For input size n, a communication tree T_n is a binary tree with each inner node being a tuple (p, m) where $p \in \{A, B\}$ and $m : [n] \to \{0,1\}$, and each edge of T_n is labeled either 0 or 1. Each leaf node is labeled either 0 or 1. For any fixed tree T_n and inputs $x, y \in [n]$, communication proceeds by setting the current node c to the root node. At each step of the protocol, if c is an inner node (A, m) then Alice sends $m(x)$ to Bob and both players set c to the child along the edge labeled $m(x)$. If c is an inner node (B, m) then Bob sends $m(y)$ to Alice, and both players set c to the child along the edge labeled $m(y)$. The protocol terminates when c becomes a leaf node, and the output is the value of the leaf node; we write $T(x,y)$ for the output of communication tree T on inputs x, y.

For communication problem $f = (f_n)_{n\in\mathbb{N}}$, a randomized protocol must satisfy $T_n(x,y) = f_n(x,y)$ with probability at least $2/3$, where the probability is over the choice of T_n. The cost of a protocol for f_n is the minimum value d such that all trees T_n in the support of the distribution have depth at most d.

We now describe the central communication-to-graph correspondence, which is illustrated in Figure 4.1. For any undirected graph $G = (V, E)$, we identify E with the function $E : V \times V \to \{0,1\}$ where $E(x,y)$ holds true ($E(x,y) = 1$) if and only if (x,y) is an edge of G.

We will define the hereditary class $\mathfrak{F}(f)$ associated with f as the smallest hereditary class that contains each F_n, as follows. For graphs G, H, we write $H \sqsubset G$ if H is an induced subgraph of G. Recall that for any set of graphs $\mathcal{G}$, we define the *hereditary closure* $\mathsf{her}(\mathcal{G}) := \{H : \exists G \in \mathcal{G}, H \sqsubset G\}$. By definition, $\mathsf{her}(\mathcal{G})$ is a hereditary graph class that includes $\mathcal{G}$. We then define

$$\mathfrak{F}(f) := \mathsf{her}\left(\{F_1, F_2, \dots\}\right) .$$

In the other direction, for any set of graphs $\mathcal{F}$, we define the natural Adjacency communication problem, which captures the complexity of the two-player game of deciding whether the players are adjacent in a given graph $G \in \mathcal{F}$. A communication problem contains only one function f_n for each input size n, whereas $\mathcal{F}$ contains many graphs of size n. The Adjacency problem should capture the maximum complexity of computing adjacency in any graph $G \in \mathcal{F}$, so for each input size $n \in \mathbb{N}$, we choose from $\mathcal{F}_n$ the graph where adjacency is hardest to compute. Let $\prec$ be a total order on functions $[n] \times [n] \to \{0,1\}$ that extends the partial order $\prec'$ defined by $f_n \prec' g_n \iff \mathsf{CC}(f_n) < \mathsf{CC}(g_n)$. We define the Adjacency problem as $\textsc{Adj}_{\mathcal{F}} = (f_n)_{n\in\mathbb{N}}$, where

$$f_n = \max\{g_n : ([n], g_n) \in \mathcal{F}_n\}$$

and the maximum is with respect to $\prec$. Here we have written $g_n : [n] \times [n] \to \{0,1\}$ for the edge relation in the graph $([n], g_n)$. It follows that for any communication problem f, we have $f = \textsc{Adj}_{\mathcal{F}}$ where $\mathcal{F} = \{F_1, F_2, \dots\}$, since $\mathcal{F}_n$ is a singleton; but it is *not* true that $f = \textsc{Adj}_{\mathfrak{F}(f)}$, since for each $n \in \mathbb{N}$, $\textsc{Adj}_{\mathfrak{F}(f)}$ effectively chooses the *hardest* subproblem of size n of any f_m for $m \geq n$.

Proposition 4.2.2. *For any communication problem $f = (f_n)_{n\in\mathbb{N}}$ and hereditary graph class $\mathcal{F}$:*

1. $\mathsf{CC}(f) = O(1)$ *if and only if* $\mathfrak{F}(f)$ *is adjacency sketchable;*
2. $\mathcal{F}$ *is adjacency sketchable if and only if* $\mathsf{CC}(\textsc{Adj}_{\mathcal{F}}) = O(1)$.

Proof. First suppose that $\mathcal{F}$ is adjacency sketchable, so that there is an adjacency sketch of constant size s, and write $\textsc{Adj} = (\textsc{Adj}_n)_{n\in\mathbb{N}}$ for the communication problem $\textsc{Adj}_{\mathcal{F}}$. Let D be the decoder of the adjacency sketch. We obtain a constant-cost communication protocol for Adj as follows. For each $n \in \mathbb{N}$, let $G_n \in \mathcal{F}_n$ be the graph such that $\textsc{Adj}_n$ is the edge relation of G_n. On inputs $x, y \in [n]$, Alice and Bob sample the random function $\mathsf{sk} : V(G_n) \to \{0,1\}^s$ and Alice sends $\mathsf{sk}(x)$ to Bob, which requires at most s bits of communication. Then Bob simulates the decoder on $D(\mathsf{sk}(x), \mathsf{sk}(y))$. By definition

$$\mathop{\mathbb{P}}_{\mathsf{sk}}\left[D(\mathsf{sk}(x), \mathsf{sk}(y)) = \textsc{Adj}_n(x, y)\right] \geq 2/3\,.$$

Now suppose that $\mathsf{CC}(\textsc{Adj}) = k$ for some constant k. For any $G \in \mathcal{F}_n$ it holds that the edge relation $g_n : [n] \times [n] \to \{0,1\}$ for G satisfies $\mathsf{CC}(g_n) \leq \mathsf{CC}(\textsc{Adj}_n) \leq k$, by definition. For each $G \in \mathcal{F}$, let $\mathcal{P}(G)$ be the probability distribution over communication trees defined by an optimal communication protocol for the edge relation of G. Then it holds that every

communication tree in the support of $\mathcal{P}(G)$ has depth at most $\mathsf{CC}(\textsc{Adj})$. So there is some d, such that, for every $G \in \mathcal{F}$, all communication trees T in the support of $\mathcal{P}(G)$ have depth at most d. We define the adjacency sketch for $\mathcal{F}$ as follows. For every $G = (V, E) \in \mathcal{F}$, construct the random sketch sk by sampling $T \sim \mathcal{P}(G)$, and then for every $v \in V$:

> For every node c of T, append to the label $\mathsf{sk}(v)$ the following:
>
> 1. If c is an inner node (p, m) (with $p \in \{A, B\}$ and $m : [n] \to \{0, 1\}$), append the symbol p and the value $m(v)$.
> 2. If c is a leaf with value b, append the symbol L and the value b.

We define the decoder D as follows. On input $(\mathsf{sk}(u), \mathsf{sk}(v))$, the decoder simulates the communication tree T on (u, v) using the values $m(u), m(v)$ for each inner node. We therefore obtain

$$\mathop{\mathbb{P}}_{\mathsf{sk}}[D(\mathsf{sk}(u), \mathsf{sk}(v)) = E(u, v)] = \mathop{\mathbb{P}}_{T \sim \mathcal{P}(G)}[T(u, v) = E(u, v)] \geq 2/3\,.$$

Consider $\mathfrak{F}(f) = \mathsf{her}(\{F_1, F_2, \dots\})$ for the graphs F_n on vertex set $[n]$ with edge relation f_n. Then it holds for any $G \in \mathfrak{F}(f)$ that there exists $n \in \mathbb{N}$ such that $G \sqsubset F_n$. But then the edge relation g of G satisfies $\mathsf{CC}(g) \leq \mathsf{CC}(f_n) \leq k$, since the communication problem g is a subproblem of f_n. We may then construct adjacency sketches by the scheme above. □

Standard distance sketching (as distinct from the graph sketching problems that we discuss in this thesis) is usually compared to communication in the *simultaneous message passing (SMP)* model (e. g. [AK08]). We will briefly discuss the relationship between our notion of f-sketching and SMP communication. This relationship was the subject of [Har20].

In the SMP model, Alice and Bob are given (private) inputs $x, y \in [n]$ to problem $f_n : [n] \times [n] \to \{0, 1\}$. They use shared randomness to send random messages $A(x)$, $B(y)$ to a third-party referee, who must output $f_n(x, y)$ with probability at least $2/3$ over the choice of messages. The complexity of the protocol is $\max_{x,y} \max(|A(x)|, |B(y)|)$.

The difference between the SMP model and the sketching model, as we have defined it, is that all parties in the SMP model know the function f_n in advance. In our sketching model, the function f_n depends on the graph G given to Alice and Bob (but not the referee), so the referee does not know it in advance. One important consequence is that communication in the SMP model always has an upper bound of $\lceil \log n \rceil$, whereas this is not true in the sketching model.

4.2.3 Probabilistic Universal Graphs

Universal graphs were introduced in [Rad64]. Kannan, Naor, & Rudich [KNR92] observed that adjacency labeling schemes of size $O(\log n)$ are equivalent to *polynomial-sized* universal graphs.

Definition 4.2.3. Let $\mathcal{F}$ be a class of graphs. A sequence $U = (U_1, U_2, \dots)$ of graphs is called a *universal graph* for $\mathcal{F}$ if for every $n \in \mathbb{N}$ and every graph $G \in \mathcal{F}_n$, it holds that $G \sqsubset U_n$. The *size* of the universal graph is the function $n \mapsto |V(U_n)|$.

Proposition 4.2.4 ([KNR92])**.** *A hereditary graph class $\mathcal{F}$ admits a universal graph of size* $\mathrm{poly}(n)$ *if and only if it admits an adjacency labeling scheme of size $O(\log n)$.*

We will call the analogous object for adjacency sketching a *probabilistic universal graph (PUG)* and define it as follows.

Definition 4.2.5. Let $\mathcal{F}$ be a class of graphs. A sequence $U = (U_1, U_2, \dots)$ of graphs is called a *probabilistic universal graph* for $\mathcal{F}$ if for every $n \in \mathbb{N}$ and every $G \in \mathcal{F}_n$ there is a probability distribution over functions $\phi_G : V(G) \to V(U_n)$ such that

$$\begin{aligned} \forall x, y \in V(G): \qquad xy \in E(G) &\implies \mathop{\mathbb{P}}_{\phi_G}[\phi_G(x)\phi_G(y) \in E(U_n)] \geq 2/3 \\ xy \notin E(G) &\implies \mathop{\mathbb{P}}_{\phi_G}[\phi_G(x)\phi_G(y) \notin E(U_n)] \geq 2/3\,. \end{aligned}$$

The *size* of the probabilistic universal graph is the function $n \mapsto |V(U_n)|$. If there is a constant k such that $\mathcal{F}$ admits a probabilistic universal graph of size at most k, then $\mathcal{F}$ admits a *constant-size* probabilistic universal graph. We remark that in this case we must have a finite number of unique graphs in the sequence U and therefore (by taking the graph union of these) we may instead assume that $U_1 = U_2 = \cdots$.

Following the same proof as [KNR92], we have:

Proposition 4.2.6. *Suppose a hereditary graph class $\mathcal{F}$ admits an adjacency sketch of size $s(n)$. Then $\mathcal{F}$ has a PUG of size $2^{s(n)}$.*

Proof. Let D be the decoder for the adjacency sketch for $\mathcal{F}$. For every $n \in \mathbb{N}$, we take U_n to be the graph on vertices $\{0,1\}^{s(n)}$ and edge set $xy \in E(U_n) \iff D(x,y) = 1$. For any $G \in \mathcal{F}_n$, choose a random $\mathsf{sk} : V(G) \to \{0,1\}^{s(n)}$ defined by the adjacency sketch. Observe that for all $u, v \in V(G)$,

$$\mathop{\mathbb{P}}_{\mathsf{sk}}[\mathsf{sk}(u)\mathsf{sk}(v) \in E(U_n)] = \mathop{\mathbb{P}}_{\mathsf{sk}}[D(\mathsf{sk}(u), \mathsf{sk}(v)) = 1]\,,$$

from which the conclusion follows by definition. □

An interesting example is that, by examining the adjacency sketch for the hypercube in Section 4.1, we see that a constant-dimensional hypercube K_2^d is a PUG for the class of induced subgraphs of hypercubes (although we note that the proof in Section 4.1 does not establish that K_2^4 is a PUG; this is because we used error 3/4 instead of 1/3 in that proof, which was sufficient due to its one-sided error).

4.2.4 Boosting and Derandomization

Like most randomized algorithms, we can boost sketches to arbitrarily high accuracy.

Proposition 4.2.7. *For any graph class $\mathcal{F}$, any partial function f parameterized by the graphs in $\mathcal{F}$, and any $0 < \delta_1 < \delta_2 < 1/2$ (which may depend on n), if $\mathcal{F}$ admits an f-sketch of error $\delta_2(n)$ of size $s(n)$, then it admits an f-sketch of error $\delta_1(n)$ with size $O\left(s(n) \cdot \frac{1}{(1/2-\delta_2(n))^2} \log \frac{1}{\delta_1(n)}\right)$.*

Proof. Let D be the decoder for the f-sketch with error δ_2. For any $n \in \mathbb{N}$ and any graph $G \in \mathcal{F}_n$, write $\delta_1 = \delta(n)$ and $\delta_2 = \delta_2(n)$. Let $\mathsf{sk}^{(1)}, \dots, \mathsf{sk}^{(k)} : V(G) \to \{0,1\}^{s(n)}$ be independently random, drawn from the distribution defined by the f-sketch of error δ_2. For any $x \in V(G)$ we then define the random function $\mathsf{sk}(x) = (\mathsf{sk}^{(1)}(x), \mathsf{sk}^{(2)}(x), \dots, \mathsf{sk}^{(k)}(x))$. We define the decoder D' such that on inputs $\mathsf{sk}(x), \mathsf{sk}(y)$, it outputs

$$D'(\mathsf{sk}(x), \mathsf{sk}(y)) = \mathsf{majority}\left(D(\mathsf{sk}^{(1)}(x), \mathsf{sk}^{(1)}(y)), \dots, D(\mathsf{sk}^{(k)}(x), \mathsf{sk}^{(k)}(y))\right).$$

Assume $f_G(x,y) \neq *$. Write $X_i = \mathbb{1}\left[f_G(x,y) = D(\mathsf{sk}^{(i)}(x), \mathsf{sk}^{(i)}(y))\right]$, which indicates whether the decoder was correct on the i^{th} instance, so that the X_i are independently and identically distributed Bernoulli random variables with some parameter $p \geq \delta_2$. Write $X = \sum_{i=1}^k X_i$. Then by standard concentration inequalities (see [BLM13], Exercise 2.10),

$$\begin{aligned}\mathop{\mathbb{P}}_{\mathsf{sk}}[D'(\mathsf{sk}(x), \mathsf{sk}(y)) = 0] &= \mathop{\mathbb{P}}_{\mathsf{sk}}[X < k/2] = \mathop{\mathbb{P}}_{\mathsf{sk}}[X - pk < k(1/2 - p)] \\ &\leq e^{-k(1/2-p)^2/2} \leq e^{-k(1/2-\delta_2)^2/2}.\end{aligned}$$

This is at most δ_1 when $k \geq \frac{2}{(1/2-\delta_2)^2} \ln(1/\delta_1)$. □

Lemma 4.2.8. *Suppose a class $\mathcal{F}$ admits an f-sketch of size $s(n)$. Then $\mathcal{F}$ admits an f-labeling of size $O(s(n) \log n)$.*

Proof. By Proposition 4.2.7, we obtain an f-sketch with error $\frac{1}{3n^2}$ and size $t(n) = O(s(n)\log n)$; let D be the decoder for this sketch. Then for any graph $G \in \mathcal{F}$ there is a random $\mathsf{sk} : V(G) \to \{0,1\}^{t(n)}$ such that, by the union bound:

$$\begin{aligned}&\mathop{\mathbb{P}}_{\mathsf{sk}}\left[\exists x, y \in V(G) : f(x,y) \neq *, D(\mathsf{sk}(x), \mathsf{sk}(y)) \neq f(x,y)\right] \\ &\qquad \leq n^2 \max_{x,y \in V(G)} \mathop{\mathbb{P}}_{sk}\left[f(x,y) \neq *, D(\mathsf{sk}(x), \mathsf{sk}(y)) \neq f(x,y)\right] \leq n^2 \cdot \frac{1}{3n^2} = \frac{1}{3}\,.\end{aligned}$$

Therefore there exists a fixed, deterministic function $\mathsf{sk} : V(G) \to \{0,1\}^{t(n)}$ that satisfies the requirements of an f-labeling. □

Newman's Theorem is a classic result that bounds the number of uniform random bits required to choose a randomized public-coin communication protocol. We give a version of this theorem for sketching. This gives an alternative proof for Lemma 4.2.8, since we may concatenate all $O(\log n)$ size $s(n)$ sketches obtained from each setting of the random bits.

Theorem 4.2.9 (Newman's Theorem for Sketching)**.** *Let $\epsilon, \delta > 0$ and suppose there is an ϵ-error f-sketch for the class $\mathcal{F}$ with size $s(n)$. Then there is a $(\epsilon+\delta)$-error f-sketch of size $s(n)$ for the class $\mathcal{F}$ that uses at most $\log\log\left(n^{O(1/\delta^2)}\right)$ bits of randomness to generate the sketch.*

Proof. Let D be the decoder of the ϵ-error f-sketch, and let $G \in \mathcal{F}_n$. We suppose that the random sketch $\mathsf{sk} : V(G) \to \{0,1\}^{s(n)}$ is obtained from a choice of *random seed* $r \sim \rho$ sampled from distribution ρ, so that for each $r \in \mathrm{supp}(\rho)$ we write sk_r for the sketch obtained deterministically from r.

For any $x, y \in V(G)$, we will say that a random seed $r \in \mathrm{supp}(\rho)$ is *bad* for x, y if $f_G(x,y) \neq *$ and $D(\mathsf{sk}_r(x), \mathsf{sk}_r(y)) \neq f_G(x,y)$. We will write $\mathsf{bad}(x,y,r)$ for the event that r is bad for x, y. We say that a fixed set $r_1, r_2, \ldots, r_m \in \mathrm{supp}(\rho)$ of seeds *fails* for x, y if $f_G(x,y) \neq *$ and

$$\mathop{\mathbb{P}}_{i \sim [m]}\left[\mathsf{bad}(x,y,r_i)\right] > \epsilon + \delta\,.$$

We write $\mathsf{fail}(x,y,r_1,\ldots,r_m)$ if the set $r_1,\ldots,r_m$ fails for x, y. Note that $\mathsf{fail}(x,y,r_1,\ldots,r_m)$ occurs if and only if

$$\sum_{i=1}^{m} \mathbb{1}\left[\mathsf{bad}(x,y,r_i)\right] > m(\epsilon+\delta)\,.$$

Let $r_1, \ldots, r_m$ be independent seeds drawn from ρ. The expected number of pairs $x, y \in V(G)$ such that $r_1, \ldots, r_m$ fails for x, y is

$$\begin{aligned}\mathop{\mathbb{E}}_{r_1,\ldots,r_m}\left[\sum_{x,y} \mathbb{1}\left[\mathsf{fail}(x,y,r_1,\ldots,r_m)\right]\right] &\leq n^2 \max_{x,y} \mathop{\mathbb{E}}_{r_1,\ldots,r_m}\left[\mathbb{1}\left[\mathsf{fail}(x,y,r_1,\ldots,r_m)\right]\right] \\ &= n^2 \max_{x,y} \mathop{\mathbb{P}}_{r_1,\ldots,r_m}\left[\mathsf{fail}(x,y,r_1,\ldots,r_m)\right] \\ &= n^2 \max_{x,y} \mathop{\mathbb{P}}_{r_1,\ldots,r_m}\left[\sum_{i=1}^{m} \mathbb{1}\left[\mathsf{bad}(x,y,r_i)\right] > m(\epsilon+\delta)\right] .\end{aligned}$$

For a fixed x, y, write $X_i = \mathbb{1}\left[\mathsf{bad}(x,y,r_i)\right]$. We have that $X_1, \ldots, X_m$ are independent Bernoulli random variables with some parameter $p < \epsilon$. Then (by [BLM13], Exercise 2.10) we have

$$\mathop{\mathbb{P}}_{r_1,\ldots,r_m}\left[\sum_{i=1}^{m} X_i > m(\epsilon+\delta)\right] = \mathop{\mathbb{P}}_{r_1,\ldots,r_m}\left[\sum_{i=1}^{m} X_i - pm > m(\epsilon+\delta-p)\right] \leq e^{-m\delta^2/3} .$$

Choosing any $m > 3\delta^2 \ln(n^2)$, we see that this is less than 1. Therefore there exist fixed seeds $r_1, \ldots, r_m$ such that $\mathbb{1}\left[\mathsf{fail}(x,y,r_1,\ldots,r_m)\right] = 0$ for all pairs x, y. We then obtain a new sketch with error $(\epsilon+\delta)$ by choosing $i \sim [m]$ uniformly at random using at most $\lceil \log m \rceil = \log \ln \left(n^{O(1/\delta^2)}\right)$ bits, and assigning the sketch $\mathsf{sk}_{r_i}(x)$ to each vertex x. □

4.2.5 Equality-Based Labelings

An equality-based labeling scheme is one which assigns to each vertex a deterministic label, comprising a data structure of size s that holds k "equality codes", which can be used only for checking equality. These labeling schemes: 1) capture the constant-cost randomized communication protocols that can be simulated by a constant-cost *deterministic* communication protocol with access to an Equality oracle (as studied in e.g. [CLV19, BBM+20, HHH21b]); and 2) capture a common type of adjacency labels, including those of [KNR92] for bounded arboricity graphs; see also [Cha18, CLR20] for other recent examples.

One might formalize these schemes in a few ways. We choose a definition intended to simplify notation, rather than optimize label size, since we care mainly about constant vs. non-constant. For the sake of readability, we write

$$\textsc{Eq}(a,b) = \mathbb{1}\left[a = b\right] .$$

Definition 4.2.10 (Equality-Based Labeling Scheme)**.** Let $\mathcal{F}$ be a class of graphs and let $f : \mathbb{N} \times \mathbb{N} \times \mathcal{F} \to \{0, 1, *\}$ be a partial function. An (s, t, k)-equality-based f-labeling scheme for $\mathcal{F}$ is an algorithm D, called a *decoder*, which satisfies the following. For every $G \in \mathcal{F}$ with vertex set $[n]$ and every $x \in [n]$, there is a sequence of the form

$$\ell_G(x) = [(p_1(x) \mid \vec{q}_1(x)), (p_2(x) \mid \vec{q}_2(x)), \ldots, (p_t(x) \mid \vec{q}_t(x))] \,,$$

where the vectors $p_i(x) \in \{0, 1\}^*$ are called the *prefixes*, and the entries of the vectors $\vec{q}_i(x) \in \mathbb{N}^*$ are called *equality codes* (which we may assume are positive integers). We must have $\sum_{i=1}^{t} |p_i(x)| \leq s$ and $\sum_{i=1}^{t} |\vec{q}_i(x)| \leq k$ (recall that given a vector v of binary numbers or integers, $|v|$ denotes the number of entries of v). We insist on the fact that k bounds the total number of equality codes associated with any vertex x, but not necessarily the total number of bits needed to store these codes (see Example 4.2.11 below, where $k = 2$ but storing the codes would require $2 \log n$ bits per vertex). On inputs $\ell_G(x), \ell_G(y)$, the algorithm D chooses a function $D_{p(x),p(y)}$, where $p(x) = (p_1(x), \ldots, p_t(x))$, and outputs

$$D_{p(x),p(y)}(Q_{x,y}) \,,$$

where

$$Q_{x,y}(i_1, i_2, j_1, j_2) = \textsc{Eq}\left((\vec{q}_{i_1}(x))_{j_1}, (\vec{q}_{i_2}(x))_{j_2}\right) \tag{4.1}$$

is the set of equality values for every pair of equality codes. It is required that, for every $G \in \mathcal{F}$ and $x, y \in V(G)$,

$$f_G(x, y) \neq * \implies D_{p(x),p(y)}(Q_{x,y}) = f_G(x, y) \,.$$

We will say that a class $\mathcal{F}$ admits a *constant-size* equality-based f-labeling scheme if there exist constant s, t, k such that $\mathcal{F}$ admits an (s, t, k)-equality-based f-labeling scheme. We make the further distinction of calling a labeling scheme (s, t, k)-*disjunctive* if it is an (s, t, k)-equality-based labeling scheme, where each function $D_{p(x),p(y)}$ is simply a disjunction over a subset of values $Q_{x,y}(i_1, i_2, j_1, j_2)$.

When an element $(p_i(x) \mid \vec{q}_i(x))$ in an equality-based label has $p_i(x)$ of size 0, we will write $(- \mid \vec{q}_i(x))$; similarly, we write $(p_1(x) \mid -)$ when $\vec{q}_i(x)$ is empty.

Example 4.2.11. The adjacency labeling scheme of [KNR92] for forests can be written as an equality-based labeling scheme. For each x in an n-vertex forest G with arbitrarily rooted trees, which we assume has vertex set $[n]$, we assign the label $\ell_G(x) = [(- \mid (x, p(x)))]$ where $p(x)$ is the parent of x if it has one, or 0 otherwise. Here $\vec{q}_1(x) = (x, p(x)) \in \mathbb{N}^2$. The decoder simply outputs the disjunction of $p(x) = y$ or $p(y) = x$, so in fact this is a $(0, 1, 2)$-disjunctive labeling scheme.

An equality-based labeling scheme is easily transformed into a standard deterministic labeling scheme or a sketch. We sketch the proof for the sake of clarity.

Proposition 4.2.12. *Let $\mathcal{F}$ be a class of graphs and $f : \mathbb{N} \times \mathbb{N} \times \mathcal{F} \to \{0, 1, *\}$ be a partial function. If there is an (s, t, k)-equality-based f-labeling scheme for $\mathcal{F}$ then there is an f-sketch for $\mathcal{F}$ of size at most $O(s + t + k \log k)$. If the scheme is disjunctive, the sketch has* one-sided error*: when $f(x, y, G) = 1$, the sketch will produce the wrong output with probability 0.*

Proof sketch. Choose a random function $\xi : \mathbb{N} \to [w]$ for $w = 3k^2$. For any vertex x of a graph G, replace each vector $\vec{q}_i(x) = (q_{i,1}(x), \ldots, q_{i,m}(x))$ with $(\xi(q_{i,1}(x)), \ldots, \xi(q_{i,m}(x)))$. We have replaced each of the (at most) k equality codes $(\vec{q}_i(x))_j$ with $\xi((q_i(x))_j)$, using $k \log w = O(k \log k)$ bits in total. The sketch has size $O(s + t + k \log k)$ since we must include each $p_i(x)$ (using s bits in total), the $O(k \log k)$ bits for the equality codes, and $O(t)$ bits to encode the symbols $(\ |\ )$.

For two vertices x, y, write $Q^{\xi}_{x,y}(i_1, i_2, j_1, j_2) = \mathbb{1}\left[\xi((\vec{q}_{i_1}(x))_{i_2}) = \xi((\vec{q}_{j_1}(y))_{j_2})\right]$. Since there are at most k equality codes in each label, there are at most k^2 equality comparisons. By the union bound, the probability that any of these comparisons have

$$\mathbb{1}\left[\xi((\vec{q}_{i_1}(x))_{i_2}) = \xi((\vec{q}_{j_1}(y))_{j_2})\right] \neq \mathbb{1}\left[(\vec{q}_{i_1}(x))_{i_2} = (\vec{q}_{j_1}(y))_{j_2}\right]$$

is at most $k^2 \cdot (1/w) = 1/3$, so with probability at least $2/3$ all of the comparisons made by the decoder have the correct value, so the decoder will be correct. Note that when $(\vec{q}_{i_1}(x))_{i_2} = (\vec{q}_{j_1}(y))_{j_2}$, the random values under ξ will be equal with certainty. We conclude from this that disjunctive schemes will produce sketches with one-sided error. □

Here we give some simple adjacency sketches for equivalence graphs and bounded-arboricity graphs that we will require for our later results.

Definition 4.2.13. A graph G is an *equivalence graph* if it is a disjoint union of cliques. A colored bipartite graph $G = (X, Y, E)$ is a *bipartite equivalence graph* if it is a colored disjoint union of bicliques, i.e. if there are partitions $X = X_1 \cup \cdots \cup X_m$, $Y = Y_1 \cup \cdots \cup Y_m$ such that each $G[X_i, Y_i]$ is a biclique and each $G[X_i, Y_j]$ is a co-biclique when $i \neq j$.

The equivalence graphs are exactly the P_3-free graphs and the bipartite equivalence graphs are exactly the P_4-free bipartite graphs. The following fact is an easy exercise.

Fact 4.2.14. *The equivalence graphs and the bipartite equivalence graphs admit constant-size equality-based labeling schemes.*

Definition 4.2.15. A graph $G = (V, E)$ has *arboricity* α if its edges can be partitioned into at most α forests.

In the next lemma, we interpret the classic labeling scheme of [KNR92] as an equality-based labeling scheme, and we obtain adjacency sketches for bounded-arboricity graphs that improves slightly upon the naïve bound in Proposition 4.2.12 and in [Har20].

Lemma 4.2.16. *For any $\alpha \in \mathbb{N}$, let $\mathcal{A}$ be the family of graphs with arboricity at most α. Then $\mathcal{A}$ admits a constant-size equality-based adjacency labeling scheme. $\mathcal{A}$ also admits an adjacency sketch of size $O(\alpha)$.*

Proof. For any graph $G \in \mathcal{A}_n$ with vertex set $[n]$, partition the edges of G into forests $F_1, \ldots, F_\alpha$ and to each tree in each forest, identify some arbitrary vertex as the root. For every vertex x, assign equality codes $q_1(x) = x$ and for $i \in [\alpha]$ set $q_{i+1}(x)$ to be the parent of x in forest F_i; if x is the root assign $q_{i+1}(x) = 0$. For vertices x, y, the decoder outputs

$$\left(\bigvee_{j=2}^{\alpha} \textsc{Eq}(q_1(x), q_j(y))\right) \vee \left(\bigvee_{j=2}^{\alpha} \textsc{Eq}(q_1(y), q_j(x))\right).$$

This is 1 if and only if y is the parent of x or x is the parent of y in some forest F_i.

One can apply Proposition 4.2.12 to obtain an $O(\alpha \log \alpha)$ adjacency sketch. We can improve this using a Bloom filter, since the output is simply a disjunction of equality checks. To each $i \in [n]$, assign a uniformly random number $r(i) \sim [6\alpha]$, and to each vertex x assign the sketch $(r(x), b(x))$ where $b(x) \in \{0,1\}^{6\alpha}$ satisfies $b(x)_i = 1$ if and only if $r(q_j(x)) = i$ for some $j \in \{2, \ldots, \alpha + 1\}$. On input $(r(x), b(x))$ and $(r(y), b(y))$, the decoder outputs 1 if and only if $b(x)_{r(y)} = 1$ or $b(y)_{r(x)} = 1$. If y is a parent of x in any of the α forests, then $y = q_j(x)$ for some j, so $b(x)_{r(y)} = b(x)_{r(q_j(x))} = 1$ and the decoder will output 1 with probability 1. Similarly, if x is a parent of y in any of the α forests, the decoder will output 1 with probability 1. The decoder fails only when x, y are not adjacent and $r(x) = r(q_j(y))$ or $r(y) = r(q_j(x))$ for some j. By the union bound, this occurs with probability at most $2\alpha \cdot \frac{1}{6\alpha} = 1/3$, as desired. The size of the sketches is $O(\log(\alpha) + \alpha) = O(\alpha)$. □

Remark 4.2.17. Disjunctive labeling schemes with $s = 0$ (i.e. the p values are empty) can be transformed into *locality-sensitive hashes (LSH)* [IM98]. A $(r_1, r_2, \gamma_1, \gamma_2)$-LSH must map any two points x, y with $\mathsf{dist}(x, y) \leq r_1$ to the same hash value with probability at least γ_1, and map any two points x, y with $\mathsf{dist}(x, y) > r_2$ to the same hash value with probability at most γ_2, where $r_1 < r_2$ and $\gamma_1 > \gamma_2$. By boosting the success probability of each EQUALITY check in the disjunction, and then sampling a uniformly random term

from the disjunction, one obtains an LSH with distance parameters that depend on the original sketch. All of the equality-based sketches presented in this chapter, except the first-order sketches, are of this form.

An *equality-based communication protocol* is a deterministic protocol that has access to an EQUALITY oracle. Specifically, at each step of the protocol, the players can decide to (exactly) compute an instance of EQUALITY at unit cost. Informally, this means that the nodes of the communication tree are either standard nodes, as in deterministic two-way communication, or EQUALITY nodes. Formally, we define these protocols as follows.

Definition 4.2.18 (Equality-based Communication Protocol)**.** An *equality-based communication protocol* for a communication problem $f = (f_n)_{n\in\mathbb{N}}$, $f_n : [n] \times [n] \to \{0,1\}$ is a deterministic protocol of the following form. For each n there is a binary communication tree T_n whose inner nodes are either:

1. *Communication nodes* of the form (p, m), where p is a symbol in $\{A, B\}$ and $m : [n] \to \{0,1\}$; or,
2. *Equality nodes* of the form (a, b), where $a, b : [n] \to \mathbb{N}$,

and edges are labeled in $\{0,1\}$. Leaf nodes of the T are labeled with values in $\{0,1\}$. On input $x, y \in [n]$, the players Alice and Bob perform the following. Each player keeps track of the current node c, which begins at the root of T. The protocol proceeds as follows:

1. If c is a leaf node, the protocol outputs the label of that node.
2. If $c = (p, m)$ is a communication node and $p = A$, then Alice computes $m(x)$ and sends the result to Bob, and both players reset c to be the child labeled with edge value $m(x)$. If $p = B$ then Bob computes $m(y)$ and sends the result to Alice and c becomes the child labeled with edge value $m(y)$.
3. If $c = (a, b)$ is an equality node, then c moves to the child labeled with edge value $\mathrm{EQ}[a(x), b(y)]$.

For a communication tree T and inputs x, y, we will write $T(x, y)$ for the output of the protocol T. We will write $\mathsf{CC}^{\mathrm{EQ}}(f_n)$ for the minimum depth of such a tree that computes f_n, and $\mathsf{CC}^{\mathrm{EQ}}(f)$ for the function $n \mapsto \mathsf{CC}^{\mathrm{EQ}}(f_n)$. The equality-based communication protocol is *constant-cost* if $\mathsf{CC}^{\mathrm{EQ}}(f) = O(1)$.

It was observed in [CLV19] that EQUALITY nodes can simulate standard communication nodes. We include a proof for the sake of clarity.

Proposition 4.2.19. *For any equality-based communication tree T, there is an equality-based communication tree T' with the same depth as T, with all nodes being equality nodes, and such that $T'(x, y) = T(x, y)$ on all inputs x, y.*

Proof. Consider any node (p, m) in T with $p \in \{A, B\}$. If $p = A$, replace this node with an equality node (a, b) where $a(x) = m(x)$ and $b(y) = 1$. If $p = B$, replace this node with an equality node (a, b) where $a(x) = 1$ and $b(y) = m(y)$. We now observe that for any x, y, if $p = A$ then the output of the node is 1 if and only if $1 = m(x)$, and if $p = B$ then the output of the node is 1 if and only if $1 = m(y)$. So the output of this node is the same as the original node (p, m). We may replace each node in the tree in such a way to produce T'. □

Proposition 4.2.20. *Let $\mathcal{F}$ be any hereditary graph class. If $\mathcal{F}$ admits a constant-size equality-based labeling scheme, then there is a constant-size equality-based protocol for $\text{ADJ}_{\mathcal{F}}$.*

Proof. Suppose there is a constant-size equality-based labeling scheme for $\mathcal{F}$. For any $G \in \mathcal{F}_n$, we can compute the edge relation $g : [n] \times [n] \to \{0, 1\}$ with a protocol as follows. On input x and y, Alice and Bob compute $p(x), p(y)$ and $q_i(x), q_i(y)$ for each $i \in [k]$. Alice sends $p(x)$ to Bob using s bits of communication, and then using k^2 calls to the EQUALITY oracle, they compute each pair $\text{EQ}(q_i(x), q_j(y))$ and construct $Q_{x,y}$. Then Bob outputs $D_{p(x),p(y)}(Q_{x,y})$. □

4.2.6 Lower Bound for the Class of All Graphs

Here we prove a general lower bound for size of an adjacency sketch for the class of all graphs. Below, we will write $\mathsf{adj}_G : V(G) \times V(G) \to \{0, 1\}$ for the function which satisfies $\mathsf{adj}_G(x, y) = 1$ if and only if xy is an edge of G.

Theorem 4.2.21. *Let $\mathfrak{G}$ be the class of all graphs. Any adjacency sketch for $\mathfrak{G}$ has size $s(n) = \Omega(n)$.*

Proof. Consider any sketch for $\mathfrak{G}$ with decoder D and fix any n. For any graph G on n vertices, let sk denote the random sketch function. For any $x, y \in V(G)$, write $F(x, y) = \mathbb{1}[D(\mathsf{sk}(x), \mathsf{sk}(y)) \neq \mathsf{adj}_G(x, y)]$, so that

$$\mathbb{E}[F(x, y)] = \mathbb{P}[D(\mathsf{sk}(x), \mathsf{sk}(y)) \neq \mathsf{adj}_G(x, y)] \leq 1/3\,.$$

Then for a uniformly randomly chosen pair of distinct vertices (x, y), we have

$$\mathop{\mathbb{E}}_{\mathsf{sk}}\left[\mathop{\mathbb{E}}_{x,y}\left[F(x,y)\right]\right] = \mathop{\mathbb{E}}_{(x,y)}\left[\mathop{\mathbb{E}}_{\mathsf{sk}}\left[F(x,y)\right]\right] \leq 1/3\,,$$

so there exists a fixed function $f_G : V(G) \to \{0,1\}^{s(n)}$ such that

$$\mathop{\mathbb{E}}_{x,y}\left[D(f_G(x), f_G(y)) \neq \mathsf{adj}_G(x,y)\right] \leq 1/3\,.$$

Suppose that G, G' have $f_G = f_{G'}$, and define $\mathsf{dist}(G,G') = \mathop{\mathbb{P}}_{x,y}[\mathsf{adj}_G(x,y) \neq \mathsf{adj}_G G'(x,y)]$, where x, y is a uniformly random pair of distinct vertices. Write $N = \binom{n}{2}$. The number of graphs G' with $\mathsf{dist}(G,G') \leq 1/3$ is at most

$$\sum_{k=0}^{N/3}\binom{N}{k} \leq \left(\frac{eN}{N/3}\right)^{N/3} = 2^{\frac{N}{3}(\log(e)+\log N-\log(N/3))} = 2^{\frac{N}{3}\log(3e)}$$

There are at most $2^{s(n)\cdot n}$ functions $f : [n] \to \{0,1\}^{s(n)}$. For each function that arises as f_G for some graph G, there are most $2^{\frac{N}{3}\log(3e)}$ graphs G with $f_G = f$. Since we must cover the set of all 2^N graphs, we must have

$$2^N \leq 2^{\frac{N}{3}\log(2e)} \cdot 2^{s(n)\cdot n} \leq 2^{\frac{n(n-1)}{6}\log(2e)+s(n)\cdot n}\,.$$

Therefore

$$\begin{aligned} s(n)\cdot n &\geq \frac{n(n-1)}{2} - \frac{n(n-1)}{6}\log(2e) \\ s(n) &\geq \frac{n-1}{2} - \frac{n-1}{6}\log(2e) \\ &\geq \frac{n-1}{2} - \frac{n-1}{6}(2.45) \geq \frac{3-2.45}{6}(n-1)\,. \end{aligned}$$

This concludes the proof. □

4.2.7 Stability and Forbidden Induced Subgraphs

An important notion in the structure of hereditary graph classes, with respect to adjacency sketching, is *stability*. We take this notion and terminology from the literature on first-order model checking; see e.g. [CS18, NMP+21, GPT21]. Recall the *chain number* from Chapter 1, Section 1.3.5.

Definition 4.2.22 (Chain Number)**.** For a graph G, the *chain number* $\mathsf{ch}(G)$ is the maximum number k for which there exist disjoint sets of vertices $\{a_1, \ldots, a_k\}, \{b_1, \ldots, b_k\} \subseteq V(G)$ such that $(a_i, b_j) \in E(G)$ if and only if $i \leq j$. For a graph class $\mathcal{F}$, we write $\mathsf{ch}(\mathcal{F}) = \max_{G \in \mathcal{F}} \mathsf{ch}(G)$. If $\mathsf{ch}(\mathcal{F}) = \infty$, then $\mathcal{F}$ has *unbounded chain number*, otherwise it has *bounded chain number*.

As in [CS18, NMP+21] we call graph classes of bounded chain number *stable* (they are called *graph-theoretically stable* in [GPT21]). These classes have many interesting properties, including stronger versions of Szemerédi's Regularity Lemma [MS14] and the Erdős-Hajnal property [CS18], and they play a central role in algorithmic graph theory [GPT21]. Stability is also important for understanding sketching and communication.

This is clearly illustrated by the Greater-Than problem, which is defined as $\mathrm{GT}_n : [n] \times [n] \to \{0, 1\}$, where $\mathrm{GT}_n(x, y) = 1$ if and only if $x \leq y$. It is straightforward to check that, if a hereditary class $\mathcal{F}$ has unbounded chain number, then the Greater-Than problem on domain $[n]$ can be reduced to the problem of computing adjacency in some graph $G \in \mathcal{F}_{2n}$.

Recall the SMP model of communication from Section 4.2.2. It is known that on domain $[n]$, the SMP complexity of Greater-Than is $\Theta(\log n)$ (see the bibliographic remark in Appendix B).

Lemma 4.2.23. *If a hereditary graph class $\mathcal{F}$ is not stable, then any adjacency sketch for $\mathcal{F}$ has size at least $\Omega(\log n)$.*

Proof. This follows from the fact that an adjacency sketch for $\mathcal{F}$ can be used to construct a communication protocol for Greater-Than in the SMP model of communication. The construction is as follows. Let D be the decoder for an adjacency sketch for $\mathcal{F}$, and let $s(n)$ be the size of the adjacency sketch. Given inputs $x, y \in [n]$, Alice and Bob can compute $\mathrm{GT}_n(x, y)$ in the SMP model by choosing a graph $G \in \mathcal{F}$ with $\mathsf{ch}(G) = n$, so there exist disjoint sets of vertices $\{a_1, \ldots, a_n\}, \{b_1, \ldots, b_n\}$ such that (a_i, b_j) are adjacent if and only if $i \leq j$. Since $\mathcal{F}$ is hereditary, the induced subgraph $H \sqsubset G$ on vertices $\{a_1, \ldots, a_n, b_1, \ldots, b_n\}$ is in $\mathcal{F}$. Alice and Bob draw random sketches $\mathsf{sk}(a_x), \mathsf{sk}(b_y)$ of size $s(2n)$ according to the adjacency sketch for H, and send them to the referee, who outputs $D(\mathsf{sk}(a_x), \mathsf{sk}(b_y))$. This communication protocol has complexity at most $s(2n)$, so by the lower bound on the SMP complexity of Greater-Than, we must have $s(2n) = \Omega(\log n)$ for any n. □

Therefore, any adjacency sketchable graph class must have bounded chain number; i.e. it is stable. In other words, if f is any communication problem with $\mathsf{CC}(f) = O(1)$ then

$\mathfrak{F}(f)$ is stable. In the next subsection we will see that stability is closely related to the structure of the lattice of hereditary graph classes, and in particular the set of minimal factorial classes.

We conclude this section with a useful characterization of stable graph families via forbidden induced subgraphs. It is a well-known fact that any hereditary graph family can be defined by its set of *minimal forbidden induced subgraphs*. That is, for any hereditary family $\mathcal{F}$, there is a *unique minimal* set of graphs $\mathcal{H}$ such that $\mathcal{F}$ is the family $\mathcal{H}$*-free* graphs, i.e. $\mathcal{F} = \mathsf{Free}(\mathcal{H})$, where

$$\mathsf{Free}(\mathcal{H}) := \{G : \forall H \in \mathcal{H}, H \not\sqsubseteq G\}.$$

One can show (Proposition 4.3.4) that a graph family $\mathcal{F}$ has a bounded chain number (i.e. $\mathcal{F}$ is stable) if and only if

$$\mathcal{F} \subseteq \mathsf{Free}(H_p^{\circ\circ}, H_q^{\bullet\circ}, H_r^{\bullet\bullet}), \text{ for some choice of } p, q, r,$$

where $H_p^{\circ\circ}$ is a *half-graph*, $H_q^{\bullet\circ}$ is a *co-half-graph*, and $H_r^{\bullet\bullet}$ is a *threshold graph* (Definition 4.3.2, depicted in Figure 4.2). For any $\mathcal{F}$, we denote by $\mathsf{stable}(\mathcal{F})$ the set of all stable subfamilies of $\mathcal{F}$.

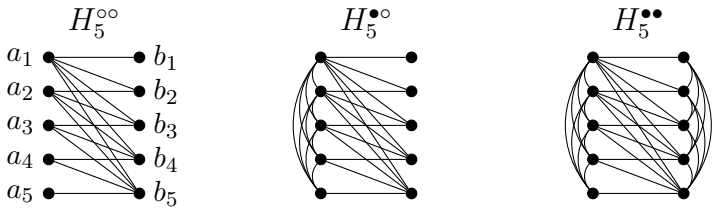


Figure 4.2: Examples of the half-graph, co-half-graph, and threshold graphs.

4.2.8 Example: k-Hamming Distance

Here we give an example to help to understand the communication-to-graph correspondence and some of its nuances.

The k-Hamming Distance problem HD^k requires Alice and Bob to decide whether the Hamming distance between their inputs $x, y \in \{0,1\}^d$ is at most $k(d)$. It has complexity $\Theta(k(d)\log k(d))$ when $k(d) = o(\sqrt{d})$ [HSZZ06, Sağ18]. Setting $k(d)$ to be non-constant, we will see that the non-constant bound on the communication complexity is "caused" by

GREATER-THAN subproblems. We will show that every $t \in \mathbb{N}$ we can choose d sufficiently large to construct disjoint sets $\{a^{(1)}, \ldots, a^{(t)}\}, \{b^{(1)}, \ldots, b^{(t)}\} \subset \{0,1\}^d$ so that $\mathsf{dist}(a_i, b_j) \leq k(d)$ if and only if $i \leq j$. We choose d such that $k = k(d) \geq t$ and define $a_r^{(i)} = 1$ if and only if $r = i$ and $b_r^{(j)} = 1$ if and only if $r > d - k + j$ or $r \leq j$.

For $i \leq j$ it holds that $a_i^{(i)} = b_i^{(j)} = 1$ while $b^{(j)}$ takes value 1 on exactly $k-1$ other coordinates, so $\mathsf{dist}(a^{(i)}, b^{(j)}) \leq k$. On the other hand, if $i > j$ then $a_i^{(i)} = 1, b_i^{(j)} = 0$ and $b^{(j)}$ takes value 1 on exactly k other coordinates, so $\mathsf{dist}(a^{(i)}, b^{(j)}) = k + 1$.

This illustrates some subtleties of the correspondence. Write $n = 2^d$ and think of domain $\{0,1\}^d$ as $[n] = [2^d]$. Let $k(d) = \omega(1)$. Then $\mathsf{CC}(\mathrm{HD}_n^k) = \Theta(k(d) \log k(d))$. The corresponding hereditary graph family $\mathfrak{F}(\mathrm{HD}^k)$ has unbounded chain number, so for every m there is $G \in \mathfrak{F}(\mathrm{HD}^k)$ with chain number m. So $\mathsf{CC}(\textsc{Adj}_{\mathfrak{F}(\mathrm{HD}^k)}) = \Omega(\log d) = \Omega(\log \log n)$. But for $k(d) = \log \log \log d$, say, this is a doubly-exponential increase in complexity. This shows how the hereditary closure within the map $\mathfrak{F}$ "blows up" any non-constant subproblem.

4.3 Adjacency Sketching and the Lattice of Hereditary Graph Classes

The hereditary graph classes form a lattice, since for any two hereditary classes $\mathcal{F}$ and $\mathcal{H}$, it holds that $\mathcal{F} \cap \mathcal{H}$ and $\mathcal{F} \cup \mathcal{H}$ are also hereditary classes. In this section we review the structure of this lattice, and give some basic results that place the set of constant-PUG classes within this lattice.

4.3.1 The Speed of Hereditary Graph Classes

The speed $|\mathcal{F}_n|$ of a hereditary graph class cannot be arbitrary. Classic results of Alekseev [Ale92, Ale97], Bollobás & Thomason [BT95], and Scheinerman & Zito [SZ94] have classified some of the possible speeds of hereditary graph classes. Scheinerman & Zito [SZ94] and Alekseev [Ale97] showed that the four smallest *layers* of hereditary graph classes are the following:

1. The *constant* layer contains classes $\mathcal{F}$ with $\log |\mathcal{F}_n| = \Theta(1)$, and hence $|\mathcal{F}_n| = \Theta(1)$,
2. The *polynomial* layer contains classes $\mathcal{F}$ with $\log |\mathcal{F}_n| = \Theta(\log n)$,

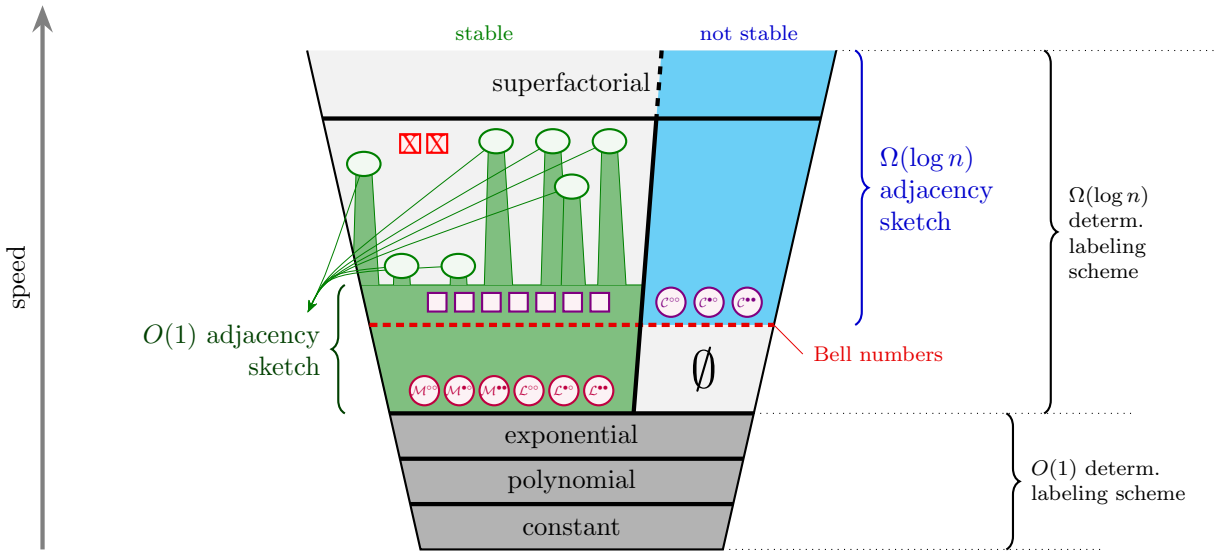


Figure 4.3: The lattice of hereditary graph families, including the minimal factorial classes (circles) and minimal classes above the Bell numbers (purple squares and circles; there are infinitely many of the squares). The bold vertical line separates the stable from unstable hereditary classes. Positive results contained in this thesis (mostly in Chapter 7) are represented in green. The red X boxes are the stable, factorial classes known *not* to be adjacency sketchable, including the construction of [HHH21a] and any monotone factorial class with $\omega(n)$ edges.

3. The *exponential* layer contains classes $\mathcal{F}$ with $\log|\mathcal{F}_n| = \Theta(n)$,

4. The *factorial* layer contains classes $\mathcal{F}$ with $\log|\mathcal{F}_n| = \Theta(n \log n)$.

The graph classes with *subfactorial* speed (the first three layers) have simple structure [SZ94, Ale97]. As demonstrated by earlier examples, the factorial layer is substantially richer and includes many graph classes of theoretical or practical importance. Despite this, no general characterization is known for them apart from the definition.

4.3.2 Constant-Size Deterministic Labeling Schemes

This thesis asks which subset of the hereditary factorial classes correspond to the communication problems with constant-cost randomized protocols. Replacing *randomized* protocols with *deterministic* protocols, we get a question that is quickly answered by the existing literature (this corresponds to the gray-colored areas in Figure 4.1. By the argument in Proposition 4.2.2, these protocols correspond to constant-size (deterministic) adjacency labeling schemes, so our question is answered by a result of Scheinerman [Sch99]: a hereditary class $\mathcal{F}$ admits a constant-size adjacency labeling scheme if and only if it belongs to the *constant, polynomial*, or *exponential* layer. Such classes have a bounded number of equivalence classes of vertices, where two vertices x, y are equivalent if their neighborhoods satisfy $N(x) \setminus \{y\} = N(y) \setminus \{x\}$.

The relationship between constant-size adjacency labels and constant-cost deterministic communication follows from the arguments in Section 4.2.2.

Proposition 4.3.1. *A communication problem f admits a constant-cost deterministic protocol if and only if $\mathfrak{F}(f)$ is in the constant, polynomial, or exponential layer. A hereditary graph class $\mathcal{F}$ is in the constant, polynomial, or exponential layer if and only if there is a constant-cost deterministic protocol for $\textsc{Adj}_{\mathcal{F}}$.*

On the other hand, adjacency labels for a factorial class must have size $\Omega(\log n)$ since graphs in the minimal factorial classes can have $\Omega(n)$ equivalence classes of vertices, and each equivalence class requires a unique label. So there is a jump in label size from $O(1)$ in the subfactorial layers to $\Omega(\log n)$ in the factorial layer.

4.3.3 Minimal Factorial Classes

The factorial layer has a set of 9 *minimal* classes, which satisfy the following:

1. Every factorial class $\mathcal{F}$ contains at least one minimal class;
2. For each minimal class $\mathcal{M}$, any hereditary subclass $\mathcal{M}' \subset \mathcal{M}$ has subfactorial speed.

These classes were identified by Alekseev [Ale97], and similar results were independently obtained by Balogh, Bollobás, & Weinreich [BBW00].

Each minimal factorial class is either a class of bipartite graphs, or a class of *co-bipartite* graphs (i.e. complements of bipartite graphs), or a class of *split* graphs (i.e. graphs whose vertex set can be partitioned into a clique and an independent set). Six of the minimal classes are the following:

- $\mathcal{M}^{\circ\circ}$ is the class of bipartite graphs of degree at most 1.
- $\mathcal{M}^{\bullet\circ}$ is the class of graphs whose vertex set can be partitioned into a clique and an independent set such that ever vertex in each of the parts is adjacent to at most one vertex in the other part.
- $\mathcal{M}^{\bullet\bullet}$ is the class of graphs whose vertex set can be partitioned into two cliques such that every vertex in each of the parts is adjacent to at most one vertex in the other part.
- $\mathcal{L}^{\circ\circ}$, $\mathcal{L}^{\bullet\circ}$, $\mathcal{L}^{\bullet\bullet}$ are defined similarly to the classes $\mathcal{M}^{\circ\circ}$, $\mathcal{M}^{\bullet\circ}$, $\mathcal{M}^{\bullet\bullet}$, respectively, with the difference that vertices in each of the parts are adjacent to all but at most one vertex in the other part.

The other three minimal classes motivate our focus on the *stable* factorial classes. They are defined as follows (see Figure 4.2).

Definition 4.3.2 (Chain-Like Graphs)**.** For any $k \in \mathbb{N}$, the *half-graph* is the bipartite graph $H_k^{\circ\circ}$ with vertex sets $\{a_1, \dots, a_k\}$ and $\{b_1, \dots, b_k\}$, where the edges are exactly the pairs (a_i, b_j) that satisfy $i \leq j$. The *threshold graph* $H_k^{\bullet\circ}$ is the graph defined the same way, except including all edges (a_i, a_j) where $i \neq j$. The *co-half-graph* $H_k^{\bullet\bullet}$ is the graph defined the same way as the threshold graph but also including all edges (b_i, b_j) for $i \neq j$. We define the following hereditary classes, which we collectively refer to as *chain-like graphs* ($\mathcal{C}^{\circ\circ}$ is sometimes called the class of *chain graphs*):

$$\mathcal{C}^{\circ\circ} := \mathsf{her}\{H_k^{\circ\circ} : k \in \mathbb{N}\}, \qquad \mathcal{C}^{\bullet\circ} := \mathsf{her}\{H_k^{\bullet\circ} : k \in \mathbb{N}\}, \qquad \mathcal{C}^{\bullet\bullet} := \mathsf{her}\{H_k^{\bullet\bullet} : k \in \mathbb{N}\}.$$

Proposition 4.3.3 ([Ale97])**.** *The minimal factorial classes are*

$$\mathcal{M}^{\circ\circ}, \mathcal{M}^{\bullet\circ}, \mathcal{M}^{\bullet\bullet}, \mathcal{L}^{\circ\circ}, \mathcal{L}^{\bullet\circ}, \mathcal{L}^{\bullet\bullet}, \mathcal{C}^{\circ\circ}, \mathcal{C}^{\bullet\circ}, \mathcal{C}^{\bullet\bullet}.$$

It is clear from the definitions that the classes $\mathcal{C}^{\circ\circ}, \mathcal{C}^{\bullet\circ}, \mathcal{C}^{\bullet\bullet}$ are not stable, while the other minimal classes are. A consequence of Ramsey's theorem is that a hereditary graph class $\mathcal{F}$ is stable if and only if it does not include any of $\mathcal{C}^{\circ\circ}, \mathcal{C}^{\bullet\circ}, \mathcal{C}^{\bullet\bullet}$:

Proposition 4.3.4. *Let $\mathcal{F}$ be a hereditary class of graphs. Then $\mathcal{F}$ has bounded chain number if and only if $\mathcal{C}^{\circ\circ}, \mathcal{C}^{\bullet\circ}, \mathcal{C}^{\bullet\bullet} \not\subseteq \mathcal{F}$.*

Proof. Let $** \in \{\bullet\bullet, \bullet\circ, \circ\circ\}$ and suppose $\mathcal{C}^{**} \subseteq \mathcal{F}$. By definition, $\mathcal{C}^{**}$ contains H_k^{**} for any $k \in \mathbb{N}$, so $\mathsf{ch}(\mathcal{C}^{**}) = \infty$ and if $\mathcal{C}^{**} \subseteq \mathcal{F}$ then $\mathsf{ch}(\mathcal{F}) \geq \mathsf{ch}(\mathcal{C}^{**}) = \infty$.

Now suppose $\mathcal{C}^{**} \not\subseteq \mathcal{F}$ for every $** \in \{\bullet\bullet, \bullet\circ, \circ\circ\}$. Then for every $** \in \{\bullet\bullet, \bullet\circ, \circ\circ\}$ there is some m^{**} such that all graphs $G \in \mathcal{F}$ are $H_{m^{**}}^{**}$-free. Hence, for $m = \max(m^{\bullet\bullet}, m^{\bullet\circ}, m^{\circ\circ})$, all graphs $G \in \mathcal{F}$ are $\{H_m^{\bullet\bullet}, H_m^{\bullet\circ}, H_m^{\circ\circ}\}$-free.

It was proved in [CS18] that, due to Ramsey's theorem, for every $m \in \mathbb{N}$ there exists a sufficiently large $k = k(m)$ such that any $\{H_m^{\bullet\bullet}, H_m^{\bullet\circ}, H_m^{\circ\circ}\}$-free graph G has $\mathsf{ch}(G) < k$. Hence $\mathsf{ch}(\mathcal{F}) < k$. □

The following statement is easily proved from Lemma 4.2.23. This shows that the minimal factorial classes are adjacency sketchable if and only if they are stable.

Fact 4.3.5. *$\mathcal{M}^{\circ\circ}, \mathcal{M}^{\bullet\circ}, \mathcal{M}^{\bullet\bullet}, \mathcal{L}^{\circ\circ}, \mathcal{L}^{\bullet\circ}, \mathcal{L}^{\bullet\bullet}$ admit constant-size equality-based adjacency labels (and therefore constant-size PUGs), while $\mathcal{C}^{\circ\circ}, \mathcal{C}^{\bullet\circ}, \mathcal{C}^{\bullet\bullet}$ have adjacency sketches of size $\Omega(\log n)$ (and therefore have PUGs of size $n^{\Omega(1)}$).*

Unlike standard universal graphs and adjacency labels, PUGs and adjacency sketches exhibit a large quantitative gap between the chain-like graphs and the other minimal factorial classes, suggesting that stable factorial classes behave much differently than other factorial classes and may be worth studying separately, which has not yet been done in the context of understanding the factorial layer of graph classes.

4.3.4 The Bell Numbers Threshold

The Bell numbers threshold is another speed threshold within the factorial layer. The Bell number B_n is the number of different set partitions of $[n]$, or equivalently the number of n-vertex equivalence graphs; asymptotically it is $B_n \sim (n/\log n)^n$. Similarly to the factorial layer itself, there is a set of *minimal* classes above the Bell numbers. Unlike the factorial layer, the set of minimal classes above the Bell numbers is infinite. It has been characterized explicitly [BBW05, ACFL16]. Once again, the classes $\mathcal{C}^{\circ\circ}, \mathcal{C}^{\bullet\circ}, \mathcal{C}^{\bullet\bullet}$ are minimal. This means

that *all* hereditary classes below the Bell numbers are stable. Structural properties of these classes were given in [BBW00], which we use to prove the following.

Theorem 4.3.6. *Let $\mathcal{F}$ be a hereditary graph class. Then:*

1. *If $\mathcal{F}$ is a minimal class above the Bell numbers, then $\mathcal{F}$ admits a constant-size equality-based labeling scheme (and therefore a constant-size PUG), unless $\mathcal{F} \in \{\mathcal{C}^{\circ\circ}, \mathcal{C}^{\bullet\circ}, \mathcal{C}^{\bullet\bullet}\}$.*
2. *If $\mathcal{F}$ has speed below the Bell numbers, then $\mathcal{F}$ admits a constant-size equality-based labeling scheme (and therefore a constant-size PUG).*

The proof of this theorem is straightforward; it nearly follows by definition, but there are many definitions. We present this proof in the remainder of the section.

Tools

We will use the following tools to prove our results for the classes below the Bell numbers, and the minimal classes above the Bell numbers.

Proposition 4.3.7 (Bounded vertex addition)**.** *Let $c \in \mathbb{N}$ and let $\mathcal{X}$ be a class of graphs. Denote by $\mathcal{F}$ the class of all graphs each of which can be obtained from a graph in $\mathcal{X}$ by adding at most c vertices. If $\mathcal{X}$ admits a constant-size equality-based labeling scheme, then so does $\mathcal{F}$.*

Proof. Let $G \in \mathcal{F}$ and let $W \subseteq V(G)$ be such that $G[V \setminus W] \in \mathcal{X}$ and $|W| \leq c$. Given a constant-size adjacency sketch for $G[V \setminus W]$ we will construct a constant-size adjacency sketch for G. Identify the vertices W with numbers $[c]$. For every vertex $x \in V \setminus W$, assign the label $(0, a(x), p(x) \mid q(x))$ where $a(x) \in \{0,1\}^c$ satisfies $a(x)_i = 1$ if and only if x is adjacent to vertex $i \in [c]$, and $(p(x) \mid q(x))$ is the label of x in $G[V \setminus W]$. For every vertex $i \in W = [c]$, assign the label $(1, i, a(i) \mid -)$. On inputs $(0, a(x), p(x) \mid q(x))$ and $(0, a(y), p(y) \mid q(y))$ for $x, y \in V \setminus W$, the decoder for $\mathcal{F}$ simulates the decoder for $\mathcal{X}$ on $(p(x) \mid q(x))$ and $(p(y) \mid q(y))$. On inputs $(0, a(x), p(x) \mid q(x))$ and $(1, i, a(i) \mid -)$, the decoder outputs $a(x)_i$. On inputs $(1, i, a(i) \mid -)$ and $(1, j, a(j) \mid -)$ the decoder outputs $a(i)_j$. □

Definition 4.3.8. Let $k \in \mathbb{N}$ and $\mathcal{F}$ a graph class. We denote by $\mathcal{S}(\mathcal{F}, k)$ the class of all graphs that can be obtained by choosing a graph $G \in \mathcal{F}$, partitioning $V(G)$ into at most k sets $V_1, V_2, \ldots, V_r$, $r \leq k$, and complementing edges between some pairs of sets V_i, V_j, $i \neq j$, and within some of the sets V_i.

Proposition 4.3.9 (Bounded complementations)**.** *Let $k \in \mathbb{N}$ and let $\mathcal{F}$ be a class that admits a constant-size equality-based labeling scheme. Then $\mathcal{S}(\mathcal{F}, k)$ admits a constant-size equality-based labeling scheme.*

Proof. Let G be a graph in $\mathcal{S}(\mathcal{F}, k)$ that is obtained from a graph in $H \in \mathcal{F}$ by partitioning $V(H)$ into at most k subsets and complementing the edges between some pairs of sets and also within some of the sets. To construct a constant-size equality-based labeling for G, we use a constant-size equality-based labeling for H and extend the label of every vertex by an extra $\lceil \log k \rceil + k$ bits. The first $\lceil \log k \rceil$ of these extra bits are used to store the index of the subset in the partition to which the vertex belongs, and the remaining k bits are used to store the information of whether the edges within the vertex's partition class are complemented or not and also whether the edges between the vertex's partition class and each of the other partition classes are complemented or not.

Now given new labels of two vertices we first extract the old labels and apply the decoder to infer the adjacency in H. Then we use the extra information about the partition classes and their complementations to deduce whether the adjacency needs to be flipped or not. □

Classes Below the Bell Number

We need the following definition for describing the structure of hereditary classes below the Bell number.

Definition 4.3.10. Let k be a positive integer, let D be a graph with loops allowed on the vertex set $[k]$, and let F be a simple graph on the same vertex set $[k]$. Let H' be the disjoint union of infinitely many copies of F, and for $i = 1, \ldots, k$, let V_i be the subset of $V(H')$ containing vertex i from each copy of F. Now we define H to be the graph obtained from H' by connecting two vertices $u \in V_i$ and $v \in V_j$ if and only if (u, v) is an edge in H' but not an edge in D, or (u, v) is not an edge in H' but an edge in D. Finally, we denote by $\mathcal{R}(D, F)$ the hereditary class consisting of all the finite induced subgraphs of H.

To better explain the above definition, we note that the infinite graph H' consists of k independent sets $V_1, V_2, \ldots, V_k$ such that a pair of distinct sets V_i, V_j induce a perfect matching if (i, j) is an edge in F, and V_i, V_j induce a graph without edges otherwise. The connected components of H' are each isomorphic to F, so H' has maximum degree at most k. Then the graph H is obtained from H' by complementing $H'[V_i]$ whenever i has a loop in D, and applying the bipartite complementation to $H'[V_i, V_j]$ whenever (i, j) is an edge in D.

For any $k \in \mathbb{N}$, let $\mathcal{R}(k) = \bigcup \mathcal{R}(D, F)$, where the union is over all graphs D, F satisfying Definition 4.3.10. We show the following result.

Proposition 4.3.11. *For any natural number k, the class $\mathcal{R}(k)$ admits a constant-size equality-based labeling scheme.*

Proof. From the description above, it follows that H, and therefore any of its induced subgraphs, can be partitioned into at most k sets, each of which is either a clique or an independent set, and the bipartite graph spanned by the edges between any pair of sets is either of maximum degree at most 1 or of maximum co-degree at most 1, i.e. the complement has degree at most 1. Observe that by applying to H (respectively, any of its induced subgraphs) the same complementations according to D again, we turn the graph into H' (respectively an induced subgraph of H'), i.e. to a graph of degree at most k. This shows that, for $\mathcal{Y}_k$ the class of graphs with maximum degree at most k, $\mathcal{R}(k) \subseteq \mathcal{S}(\mathcal{Y}_k, k)$. The claim then follows from Lemma 4.2.16 and Proposition 4.3.9. □

Lemma 4.3.12 ([BBW00, BBW05])**.** *For every hereditary class $\mathcal{F}$ below the Bell numbers, there exist constants c, k such that for all $G \in \mathcal{F}$ there exists a set W of at most c vertices so that $G[V \setminus W]$ belongs to $\mathcal{R}(D, F)$ for some k-vertex graphs D and F.*

Corollary 4.3.13. *Any hereditary class $\mathcal{F}$ below the Bell numbers admits a constant-size equality-based labeling scheme, and therefore a constant-size adjacency sketch.*

Proof. Let $\mathcal{F}$ be a hereditary class below the Bell numbers, and let c and k be natural numbers as in Lemma 4.3.12, i.e. for every graph G in $\mathcal{F}$ there exist a k-vertex graph D with loops allowed and a simple k-vertex graph F so that after removing at most c vertices from G we obtain a graph from $\mathcal{R}(D, F) \subseteq \mathcal{R}(k)$.

By Proposition 4.3.7, $\mathcal{F}$ admits a constant-size equality-based labeling scheme if $\mathcal{R}(k)$ does, and the latter follows from Proposition 4.3.11. □

Minimal Classes Above the Bell Number

We denote by $\mathcal{P}$ the class of path forests, i.e. graphs in which every component is a path. The following theorem of [BBW05, ACFL16] enumerates the minimal hereditary classes above the Bell number.

Theorem 4.3.14 ([BBW05, ACFL16])**.** *Let $\mathcal{F}$ be a minimal hereditary class above the Bell numbers, i.e. every proper hereditary subfamily of $\mathcal{F}$ is below the Bell numbers. Then either $\mathcal{F} \subseteq \mathcal{S}(\mathcal{P}, k)$ for some integer k, or $\mathcal{F}$ is one of the following 13 classes:*

(1) *The class $\mathcal{K}_1$ of all graphs whose connected components are cliques (equivalence graphs);*

(2) *The class $\mathcal{K}_2$ of all graphs whose connected components are stars (star forests);*

(3) *The class $\mathcal{K}_3$ of all graphs whose vertices can be partitioned into an independent set I and a clique Q, such that every vertex in Q has at most one neighbor in I;*

(4) *The class $\mathcal{K}_4$ of all graphs whose vertices can be partitioned into an independent set I and a clique Q, such that every vertex in I has at most one neighbor in Q;*

(5) *The class $\mathcal{K}_5$ of all graphs whose vertices can be partitioned into two cliques Q_1, Q_2, such that every vertex in Q_2 has at most one neighbor in Q_1*

(6) *The classes $\overline{\mathcal{K}_i}$ for $i \in [5]$, where $\overline{\mathcal{K}_i}$ is the class of complements of graphs in $\mathcal{K}_i$;*

(7) *$\mathcal{C}^{\circ\circ}, \mathcal{C}^{\bullet\circ}$, or $\mathcal{C}^{\bullet\bullet}$.*

Corollary 4.3.15. *Any minimal hereditary class $\mathcal{F}$ above the Bell numbers, except $\mathcal{C}^{\circ\circ}, \mathcal{C}^{\bullet\circ}$, and $\mathcal{C}^{\bullet\bullet}$, admits a constant-size equality-based labeling scheme.*

Proof. If $\mathcal{F} \subseteq \mathcal{S}(\mathcal{P}, k)$ for some k, then the result follows from Proposition 4.3.9 and Lemma 4.2.16 as the graphs in $\mathcal{P}$ have arboricity 1. The result for $\mathcal{K}_1$ follows by Fact 4.2.14. The result for $\mathcal{K}_2$ follows from Lemma 4.2.16 as the graphs in $\mathcal{K}_2$ have arboricity 1. The result for $\mathcal{K}_3$, $\mathcal{K}_4$, and $\mathcal{K}_5$ follows from Proposition 4.3.9 as each of these classes is a subclass of $\mathcal{S}(\mathcal{K}_2, 2)$. The result for $\overline{\mathcal{K}_i}$, $i \in [5]$ also follows from Proposition 4.3.9 as $\overline{\mathcal{K}_i} \subseteq \mathcal{S}(\mathcal{K}_i, 1)$ for every $i \in [5]$. □

Chapter 5

Sketching and Labeling for Cartesian Products

Left and right, up and down,
so many directions that we could go,
forward and back, or all around,
but all we really need to know,
in any dimension we have found,
is whether we're adjacent though.

In this chapter, we study the effect of the Cartesian product operation on the size of sketches and labels. This chapter includes results from [HWZ22] (coauthored with Sebastian Wild and Viktor Zamaraev) and [EHZ22] (coauthored with Louis Esperet and Viktor Zamaraev). Recall that for a class $\mathcal{F}$, we define

$$\mathcal{F}^{\square} = \{G_1 \square G_2 \square \cdots \square G_d : d \in \mathbb{N}, G_i \in \mathcal{F}\},$$

where $\square$ denotes the Cartesian product operation. In Section 5.1, we show that the Cartesian product essentially preserves the efficiency of sketches and labels for the base class $\mathcal{F}$. As a consequence, we obtain optimal adjacency labeling schemes for the classes $\mathsf{her}(\mathcal{F}^{\square})$ and $\mathsf{mon}(\mathcal{F}^{\square})$ for any base class $\mathcal{F}$. The proofs in this section follow a technique from [HWZ22] but have been significantly simplified and improved using arguments from our follow-up work [EHZ22].

In Section 5.2, we show that one cannot obtain constant-size adjacency sketches for the induced subgraphs of hypercubes (i.e. the class $\mathsf{her}(\{K_2\}^{\square})$) using equality-based sketches.

In other words, any deterministic communication protocol with access to an oracle for EQUALITY cannot have constant cost.

5.1 Sketches and Labeling Schemes

Our strategy for designing adjacency labeling schemes for the monotone classes $\mathsf{mon}(\mathcal{F}^{\square})$ is as follows. Suppose $G \subset G_1 \square \cdots \square G_d$ is a subgraph of a Cartesian product. Then $V(G) \subseteq V(G_1) \times \cdots \times V(G_d)$. Let $H \subset_I G_1 \square \cdots \square G_d$ be the subgraph induced by $V(G)$, so that $E(G) \subseteq E(H)$. One may think of G as being obtained from the induced subgraph H by deleting some edges. Then two vertices $x, y \in V(G)$ are adjacent if and only if:

1. There exists exactly one coordinate $i \in [d]$ where $x_i \neq y_i$;
2. On this coordinate, $x_i y_i \in E(G_i)$; and,
3. The edge $xy \in E(H)$ has not been deleted in $E(G)$.

We construct the labels for vertices in G in three phases, which check these conditions in sequence.

Our construction for Phase 2 is similar to our method for proving that the Cartesian product preserves constant-size adjacency and small-distances sketches, so we present the argument for the latter, which is the following theorem from Chapter 1:

Theorem 1.3.14. *Let $\mathcal{F}$ be a hereditary class of graphs that admits a small-distance sketch of size $s(n,k)$. Then $\mathcal{F}^{\square}$ admits a small-distance sketch of size $O(s(n,k) \cdot k^2 \log k)$. Consequently, if $\mathcal{F}$ is adjacency sketchable, then $\mathsf{her}(\mathcal{F}^{\square})$ is adjacency sketchable and admits an adjacency labeling scheme of size $O(\log n)$.*

We briefly mention how to use a similar argument to obtain the required lemma for Phase 2 of our labeling schemes. We then proceed with Phase 3 to prove

Theorem 1.3.15. *Let $\mathcal{F}$ be a hereditary class with an adjacency labeling scheme of size $s(n)$. Then:*

1. *$\mathsf{her}(\mathcal{F}^{\square})$ has a labeling scheme of size at most $4s(n) + O(\log n)$.*
2. *$\mathsf{mon}(\mathcal{F}^{\square})$ has a labeling scheme where each $G \in \mathsf{mon}(\mathcal{F}^{\square})$ on n vertices is given labels of size at most $4s(n) + O(\delta(G) + \log n)$, where $\delta(G)$ is the degeneracy of G.*

The optimality of our results on labeling schemes are presented in Section 5.1.4. This implies

Corollary 1.3.16. *If a hereditary class $\mathcal{F}$ has an efficient labeling scheme, then so do* $\mathsf{her}(\mathcal{F}^{\square})$ *and* $\mathsf{mon}(\mathcal{F}^{\square})$.

5.1.1 Phase 1: Hamming Distance

In this section, for any finite alphabet Σ and $d \in \mathbb{N}$, and any two strings $x, y \in \Sigma^d$, we will write $\mathsf{dist}(x, y)$ for the Hamming distance between x and y. For a vector $x \in \{0,1\}^d$ we will write $|x|$ for the Hamming weight of x, i.e. the number of nonzero coordinates. Our goal is to find a sketch which allows us to determine $\mathsf{dist}(x, y) \leq k$ in any finite alphabet. We begin with the binary alphabet and then show how to reduce to this case in general.

Proposition 5.1.1. *For any $d, k \in \mathbb{N}$, there is a probability distribution over functions $h : \{0,1\}^d \to \{0,1\}^q$, where $q = O(k^2)$, such that for any $x, y \in \{0,1\}^d$ it holds that*

$$\begin{aligned} \mathsf{dist}(x,y) \leq k &\implies \mathbb{P}\left[\mathsf{dist}(h(x),h(y)) \leq k\right] = 1\,, \\ &\quad \textit{and } \mathbb{P}\left[\mathsf{dist}(h(x),h(y)) = \mathsf{dist}(x,y)\right] > 3/4\,, \\ \mathsf{dist}(x,y) > k &\implies \mathbb{P}\left[\mathsf{dist}(h(x),h(y)) \leq k\right] < 1/4\,. \end{aligned}$$

Proof. The function $h : \{0,1\}^d \to \{0,1\}^q$ is chosen randomly as follows. For each $i \in [d]$, let $r(i) \sim [q]$ be independently and uniformly random. For each $j \in [q]$, write $R(j) = r^{-1}(j)$, and for any $x \in \{0,1\}^d$ write $x_{R(j)} = \bigoplus_{i \in R(j)} x_i$. We then define any $x \in \{0,1\}^d$, we then define

$$h(x) = (x_{R(1)}, x_{R(2)}, \ldots, x_{R(q)})\,.$$

Suppose that $x, y \in \{0,1\}^d$ have $\mathsf{dist}(x, y) \leq k$ and write $\Delta = \{i \in [d] : x_i \neq y_i\}$. If $x_{R(j)} \neq y_{R(j)}$ then there is $i \in R(j)$ such that $x_i \neq y_i$. So it must be that $\mathsf{dist}(h(x), h(y)) \leq \mathsf{dist}(x, y) \leq k$, with probability 1. Now, by the union bound

$$\mathbb{P}\left[\exists i, i' \in \Delta : i \neq i' \wedge r(i) = r(i')\right] \leq \frac{k^2}{2} \cdot \frac{1}{q} < 1/4\,,$$

for an appropriate choice of $q = O(k^2)$. If this bad event does not occur, then we have $\mathsf{dist}(h(x), h(y)) = \mathsf{dist}(x, y)$ as desired, since there are exactly $|\Delta|$ values $j \in [d]$ such that $x_{R(j)} \oplus y_{R(j)} \neq 0$. On the other hand, if $\mathsf{dist}(x, y) > k$, then we have

$$\mathsf{dist}(h(x),h(y)) = |h(x) \oplus h(y)| = \Big|\bigoplus_{i \in \Delta} e_{r(i)}\Big|\,.$$

Then $\mathbb{P}[\mathsf{dist}(h(x), h(y))] < 1/4$ for the appropriate choice of $q = O(k^2)$ due to Proposition 5.1.4, proved below. □

We may generalize this to arbitrary alphabets as follows.

Lemma 5.1.2. *For any finite alphabet Σ, any $d \in \mathbb{N}$, and any $k \in \mathbb{N}$, there exists a probability distribution over maps $h : \Sigma^d \to \{0,1\}^m$ such that $m = O(k^2)$ and, for all $x, y \in \Sigma^d$,*

$$\begin{aligned} \mathsf{dist}(x,y) \le k &\implies \mathbb{P}[\mathsf{dist}(h(x), h(y)) \le 2k] = 1 \\ &\quad \text{and } \mathbb{P}[\mathsf{dist}(h(x), h(y)) = 2 \cdot \mathsf{dist}(x,y)] > 3/4 \\ \mathsf{dist}(x,y) > k &\implies \mathbb{P}[\mathsf{dist}(h(x), h(y)) \le 2k] < 1/4\,. \end{aligned}$$

Proof. We may assume without loss of generality that $\Sigma = [n]$. We may then define the map $\sigma : [n] \to \{0,1\}^n$ as $\sigma(i) = e_i$, where e_i is the vector that has all 0 coordinates except coordinate i. We then define $\phi : \Sigma^d \to \{0,1\}^{dn}$ as

$$\phi(x) = (\sigma(x_1), \sigma(x_2), \ldots, \sigma(x_d))\,.$$

This has the property that for all $x, y \in \Sigma^d$,

$$\mathsf{dist}(\phi(x), \phi(y)) = 2 \cdot \mathsf{dist}(x,y)\,,$$

which can be seen by observing that when $x_i \neq y_i$, $\sigma(x_i)$ and $\sigma(y_i)$ differ on exactly on the two coordinates $x_i, y_i \in [n]$. The conclusion then follows by Proposition 5.1.4, proved below. □

It remains to prove the proposition used above. We will require the next claim. Here we write $e_i \in \{0,1\}^d$ for the standard basis vector.

Claim 5.1.3. *For any $k \in \mathbb{N}$ and $\delta > 0$, let $q \ge 2^{\frac{(k+1)(2k+1)}{\delta}}$. Let $z \in \{0,1\}^d$. Then for $t = k+1$ and $i_1, \ldots, i_t \sim [q]$ chosen independently and uniformly at random, we have*

$$\mathbb{P}[|z \oplus e_{i_1} \oplus \cdots \oplus e_{i_t}| \le k] < \delta\,.$$

Proof. Let $Z_1 = \{j \in [q] : z_j = 1\}$. First consider the case $|Z_1| > k + t$. Then $\mathbb{P}[|z \oplus e_{i_1} \oplus \ldots \oplus e_{i_t}| \le k] = 0$. So we restrict our attention to the case $|Z_1| \le k + t$. By the union bound,

$$\mathbb{P}[\exists a \in [t] : i_a \in Z_1] \le t \cdot \frac{|Z_1|}{q} \le \frac{t(k+t)}{q} = \frac{(k+1)(2k+1)}{q} < \delta/2\,.$$

Under the condition that $\forall a \in [t] : i_a \notin Z_1$, we have $|z \oplus e_{i_1} \oplus \ldots \oplus e_{i_t}| \geq |Z_1|$. If $|Z_1| > k$ we are done. Otherwise, let D be the event that $i_a \neq i_b$ for each distinct $a, b \in [t]$. Then by the union bound and the fact that $k \leq q/2$,

$$\mathbb{P}\left[\neg D \quad | \quad \forall a : i_a \notin Z_1\right] \leq \binom{t}{2} \frac{1}{q - |Z_1|} \leq \frac{(k+1)^2}{2(q-k)} \leq \frac{(k+1)(2k+1)}{q} < \delta/2\,.$$

In the event of $|Z_1| \leq k$, $\forall a \in [t] : i_a \notin Z_1$, and D, we have $|z \oplus e_{i_1} \oplus \cdots \oplus e_{i_t}| \geq |Z_1| + t \geq k + 1$. By the union bound, this occurs with probability less than δ. □

Proposition 5.1.4. *For any $r, k \in \mathbb{N}$ with $r > k$ and $\delta > 0$, let $q \geq 2^{\frac{(k+1)(2k+1)}{\delta}}$, and let $i_1, \ldots, i_r \sim [q]$ be independently and uniformly random. Then*

$$\mathbb{P}\left[|e_{i_1} \oplus \cdots \oplus e_{i_r}| \leq k\right] < \delta\,.$$

Proof. This follows by using Claim 5.1.3 by choosing $s = r - (k+1)$ (where we have $s \geq 0$), with $z = e_{i_1} \oplus \cdots \oplus e_{i_s}$ if $s \geq 1$ and $z = \vec{0}$ if $s = 0$. □

5.1.2 Phase 2: Combining Coordinate-Wise Sketches

In the second phase of the sketch, we are guaranteed that there are at most k coordinates $i \in [d]$ in our Cartesian product graph where vertices x and y differ, and we wish to combine the sketches for each coordinate $i \in [d]$ in such a way that the sketches for the differing coordinates can be recovered. It is convenient to have sketches for the factors G_i which can be combined by the XOR operation while retaining the ability to compute the output. For this purpose we define an *XOR encoding*.

Definition 5.1.5 (XOR encoding)**.** For any $s, t \in \mathbb{N}$, an (s,t)-*XOR encoding* is a function $\mathsf{enc} : \{0,1\}^s \to \{0,1\}^t$ such that, for all distinct unordered pairs $\{x, y\}, \{x', y'\} \in \{0,1\}^s \times \{0,1\}^s$ with $\{x, y\} \neq \{x', y'\}$, $x \neq y$, and $x' \neq y'$, it holds that

$$\mathsf{enc}(x) \oplus \mathsf{enc}(y) \neq \mathsf{enc}(x') \oplus \mathsf{enc}(y')\,.$$

We will also require the property that the all-zero vector $\vec{0}$ is not in the image of enc.

An XOR encoding has the property that for any $z = \mathsf{enc}(x) \oplus \mathsf{enc}(y)$, one can uniquely recover the unordered pair $\{x, y\}$.

Lemma 5.1.6. *For any $s \in \mathbb{N}$, there exists an $(s, 4s)$-XOR encoding.*

Proof. Let $\phi : \{0,1\}^s \to \{0,1\}^{4s}$ be uniformly randomly chosen, so that for every $z \in \{0,1\}^s$, $\phi(z) \sim \{0,1\}^{4s}$ is a uniform and independently random variable. For any two distinct pairs $\{z_1, z_2\}, \{z_1', z_2'\} \in \binom{\{0,1\}^s}{2}$ where $z_1 \neq z_2$, $z_1' \neq z_2'$, and $\{z_1, z_2\} \neq \{z_1', z_2'\}$, the probability that $\phi(z_1) \oplus \phi(z_2) = \phi(z_1') \oplus \phi(z_2')$ is at most 2^{-4s}, since at least one of the variables $\phi(z_1), \phi(z_2), \phi(z_1'), \phi(z_2')$ is independent of the other ones. Therefore, by the union bound,

$$\mathbb{P}\left[\exists \{z_1, z_2\}, \{z_1', z_2'\} : \phi(z_1) \oplus \phi(z_2) = \phi(z_1') \oplus \phi(z_2')\right] \leq \binom{2^s}{2}^2 2^{-4s} \leq \frac{1}{4}\,.$$

The probability that $\vec{0}$ is in the image of ϕ is at most $2^s \cdot 2^{-4s}$. Then there is $\phi : \{0,1\}^s \to \{0,1\}^{4s}$ such that $\vec{0}$ is not included in the image, where each distinct pair $\{z_1, z_2\} \in \binom{\{0,1\}^s}{2}$ is assigned has a distinct unique value $\phi(z_1) \oplus \phi(z_2)$. So the function $\Phi(\{z_1, z_2\}) = \phi(z_1) \oplus \phi(z_2)$ is a one-to-one map $\binom{\{0,1\}^s}{2} \to \{0,1\}^{4s}$. □

In the next theorem, keep in mind that, unlike adjacency, distances in G are not necessary preserved in the induced subgraphs of G. So the result for the hereditary closure $\mathsf{her}(\mathcal{F}^\square)$ holds for adjacency but not distances.

Theorem 1.3.14. *Let $\mathcal{F}$ be a hereditary class of graphs that admits a small-distance sketch of size $s(n,k)$. Then $\mathcal{F}^\square$ admits a small-distance sketch of size $O(s(n,k) \cdot k^2 \log k)$. Consequently, if $\mathcal{F}$ is adjacency sketchable, then $\mathsf{her}(\mathcal{F}^\square)$ is adjacency sketchable and admits an adjacency labeling scheme of size $O(\log n)$.*

Proof. Let $D : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$ be the decoder for the k-distance sketch for $\mathcal{F}$. We design the sketches for $\mathcal{F}^\square$ as follows. Consider a graph $G \in \mathcal{F}^\square$ on n vertices, so that $G = G_1 \square G_2 \square \cdots \square G_d$ for some $d \in \mathbb{N}$ and $G_i \in \mathcal{F}$ for each $i \in [d]$. Each G_i has at most n vertices and therefore a k-distance sketch of size at most $s = s(n)$. We may boost this sketch to have error probability at most $1/8k$ and size $t = O(s \log k)$. Observe that for any two vertices x, y we have

$$\mathsf{dist}_G(x,y) = \sum_{i=1}^{d} \mathsf{dist}_{G_i}(x_i, y_i)\,.$$

We construct the sketch as follows.

1. Treating the vertices in each G_i as characters of the alphabet $[n]$, randomly choose the function $h : [n] \to \{0,1\}^m$ given by Lemma 5.1.2. To each x, use $m = O(k^2)$ bits to assign the value $h(x) = h(x_1 \cdots x_d)$.

2. Choose a uniformly random function $r : [d] \to [8k^2]$.

3. Let $\mathsf{enc} : \{0,1\}^t \to \{0,1\}^{4t}$ be the XOR encoding given by Lemma 5.1.6. For each $i \in [d]$, choose the random function $\mathsf{sk}_i : V(G_i) \to \{0,1\}^t$ defined by the (boosted) k-distance sketch for G_i. For each vertex $x = (x_1, \ldots, x_d)$, and each $j \in [8k^2]$, use $4t = O(s \log k)$ bits to append to the sketch the string

$$\bigoplus_{i \in r^{-1}(j)} \mathsf{enc}(\mathsf{sk}_i(x_i)) \,,$$

using $8k^2 \cdot O(s \log k)$ bits in total for each x.

The decoder operates as follows. Given the sketches for $x, y \in V(G)$:

1. If $\mathsf{dist}(h(x), h(y)) > k$, output "$> k$", otherwise continue to the next step. By Lemma 5.1.2, if there are more than k coordinates on which x, y differ, the decoder will output "$> k$" with probability at least $3/4$. Below, we assume that this has occurred, and assume that there are at most k coordinates $I \subseteq [d]$ where x, y differ.

2. For each $j \in [8k^2]$, the decoder may compute

$$z_j = \bigoplus_{i \in r^{-1}(j)} \mathsf{enc}(\mathsf{sk}_i(x_i)) \oplus \mathsf{enc}(\mathsf{sk}_i(y_i)) \,,$$

using the XOR of the relevant parts of each sketch. The probability that there exist $i, i' \in I$ such that $r(i) = r(i')$ is at most $|I|/8k^2 \leq 1/8$. Assuming that this does not occur, we have for each $i \in I$ that

$$\begin{aligned} z_{r(i)} &= \bigoplus_{i' : r(i') = r(i)} \mathsf{enc}(\mathsf{sk}_{i'}(x_{i'})) \oplus \mathsf{enc}(\mathsf{sk}_{i'}(y_{i'})) \\ &= \mathsf{enc}(\mathsf{sk}_i(x_i)) \oplus \mathsf{enc}(\mathsf{sk}_i(y_i)) \oplus \left(\bigoplus_{i' \neq i : r(i') = r(i)} \mathsf{enc}(\mathsf{sk}_{i'}(x_{i'})) \oplus \mathsf{enc}(\mathsf{sk}_{i'}(y_{i'})) \right) \\ &= \mathsf{enc}(\mathsf{sk}_i(x_i)) \oplus \mathsf{enc}(\mathsf{sk}_i(y_i)) \neq \vec{0} \,. \end{aligned}$$

We may then recover each pair $\{\mathsf{sk}_i(x_i), \mathsf{sk}_i(y_i)\}$ for $i \in I$ from the values z_j. Then we compute $D(\mathsf{sk}_i(x_i), \mathsf{sk}_i(y_i)) = D(\mathsf{sk}_i(y_i), \mathsf{sk}_i(x_i))$ (it is important that D is symmetric, since we do not recover the order of $\mathsf{sk}_i(x_i)$ and $\mathsf{sk}_i(y_i)$). If any of these report that $\mathsf{dist}_{G_i}(x_i, y_i) > k$, output "$> k$". Otherwise, continue. The probability that any of these outputs of D is incorrect is at most $k \cdot 1/8k = 1/8$.

3. Finally, output $\sum_{i \in I} D(\mathsf{sk}_i(x_i), \mathsf{sk}_i(y_i))$. From the previous step, we are guaranteed that $D(\mathsf{sk}_i(x_i), \mathsf{sk}_i(y_i)) = \mathsf{dist}_{G_i}(x_i, y_i)$.

This concludes the proof of the first part of the theorem. The second part about adjacency sketching follows from the fact that a constant-size adjacency sketch for $\mathcal{F}^{\square}$ is also a constant-size adjacency sketch for its hereditary closure $\mathsf{her}(\mathcal{F}^{\square})$. □

It is easy to modify the above proof to obtain the following lemma for adjacency labeling. In this case we may XOR together the labels of all coordinates, instead of partitioning them as was done in step 2 and 3 of the sketch above. The additive $O(\log n)$ in the theorem comes from using Lemma 5.1.2 for $k = 1$, derandomized via Lemma 4.2.8.

Lemma 5.1.7. *Let $\mathcal{F}$ be a hereditary class of graphs that admits an adjacency labeling scheme of size $s(n)$. Then $\mathsf{her}(\mathcal{F}^{\square})$ admits an adjacency labeling scheme of size $4s(n) + O(\log n)$.*

5.1.3 Phase 3: Subgraphs

We now abandon sketching and complete our proof of Theorem 1.3.15. In the third phase of our adjacency labeling scheme for Cartesian products, we are promised that the vertices x, y form an edge in the *induced* subgraph $H \sqsubset G_1 \square \cdots \square G_d$, and we must check if this edge has been deleted in $E(G)$. There is a minimal and perfect tool for this task:

Theorem 5.1.8 (Minimal Perfect Hashing)**.** *For every $m, k \in \mathbb{N}$, there is a family $\mathcal{P}_{m,k}$ of hash functions $[m] \to [k]$ such that, for any $S \subseteq [m]$ of size k, there exists $h \in \mathcal{P}_{m,k}$ where the image of S under h is $[k]$. The function h can be stored in $k \ln e + \log\log m + o(k + \log\log m)$ bits of space and it can be computed by a randomized algorithm in expected time $O(k + \log\log m)$.*

Minimal perfect hashing has been well-studied and we are very grateful to Sebastian Wild for preventing us from trying to reinvent this. A proof of the space bound appears in [Meh84] and significant effort has been applied to improving the construction and evaluation time. We take the above statement from [HT01]. We now conclude the proof of Theorem 2.2.10 by applying the next lemma to the class $\mathcal{G} = \mathsf{her}(\mathcal{F}^{\square})$, using the labeling scheme for $\mathsf{her}(\mathcal{F}^{\square})$ obtained in Lemma 5.1.7 (note that $\mathsf{mon}(\mathsf{her}(\mathcal{F}^{\square})) = \mathsf{mon}(\mathcal{F}^{\square})$).

Lemma 5.1.9. *Let $\mathcal{G}$ be any graph class which admits an adjacency labeling scheme of size $s(n)$. Then $\mathsf{mon}(\mathcal{G})$ admits an adjacency labeling scheme where each $G \in \mathsf{mon}(\mathcal{G})$ on n vertices has labels of size $s(n) + O(\delta(G) + \log n)$, where $\delta(G)$ is the degeneracy of G.*

Proof. Let $G \in \mathsf{mon}(\mathcal{G})$ have n vertices, so that it is a subgraph of $H \in \mathcal{G}$ on n vertices. The labeling scheme is as follows.

1. Fix a total order $\prec$ on $V(H)$ such that each vertex x has at most $\delta = \delta(G)$ neighbors y in H such that $x \prec y$; this exists by definition. We will identify each vertex x with its position in the order.
2. For each vertex x, assign the label as follows:
   (a) Use $s(n)$ bits for the adjacency label of x in H.
   (b) Use $\log n$ bits to indicate x (the position in the order).
   (c) Let $N^+(x)$ be the set of neighbors $x \prec y$. Construct a perfect hash function $h_x : N^+(x) \to [\delta]$ and store it, using $O(\delta + \log\log n)$ bits.
   (d) Use δ bits to write the function $\mathsf{edge}_x : [\delta] \to \{0, 1\}$ which takes value 1 on $i \in [\delta]$ if and only if xy is an edge of G, where y is the unique vertex in $N^+(x)$ satisfying $h_x(y) = i$.

Given the labels for x and y, the decoder performs the following:

1. If xy are not adjacent in H, output "not adjacent".
2. Otherwise xy are adjacent. If $x \prec y$, we are guaranteed that y is in the domain of h_x, so output "adjacent" if and only if $\mathsf{edge}_x(h_x(y)) = 1$. If $y \prec x$, output "adjacent" if and only if $\mathsf{edge}_y(h_y(x)) = 1$.

This concludes the proof. □

5.1.4 Optimality

We now prove the optimality of our labeling schemes, and Corollary 1.3.16. We require:

Proposition 5.1.10. *For any hereditary class $\mathcal{F}$, let $\delta(n)$ be the maximum degeneracy of an n-vertex graph $G \in \mathsf{her}(\mathcal{F}^{\square})$. Then $\mathsf{her}(\mathcal{F}^{\square})$ contains a graph H on n vertices with at least $n \cdot \delta(n)/4$ edges, so $\mathsf{mon}(\mathcal{F}^{\square})$ contains all $2^{n\cdot\delta(n)/4}$ spanning subgraphs of H.*

Proof. Since G has degeneracy $\delta = \delta(n)$, it contains an induced subgraph $G' \sqsubset G$ with minimum degree δ and $n_1 \leq n$ vertices. If $n_1 \geq n/2$ then G itself has at least $\delta n_1/2 \geq \delta n/4$ edges, and we are done. Now assume $n_1 < n/2$. Since $G \in \mathsf{her}(\mathcal{F}^{\square})$, $G \sqsubset H_1 \square \cdots \square H_t$ for some $t \in \mathbb{N}$ and $H_i \in \mathcal{F}$. So for any $d \in \mathbb{N}$, the graph $(G')^d \sqsubset (H_1 \square \cdots \square H_t)^d$ belongs to $\mathsf{her}(\mathcal{F}^{\square})$. Consider the graph $H \sqsubset (G')^d$ defined as follows. Choose any $w \in V(G')$, and for each $i \in [d]$ let

$$V_i := \{(v_1, v_2, \ldots, v_d) : v_i \in V(G') \text{ and } \forall j \neq i, v_j = w\},$$

and let H be the graph induced by vertices $V_1 \cup \cdots \cup V_d$. Then H has dn_1 vertices, each of degree at least δ, since each $v \in V_i$ is adjacent to δ other vertices in V_i. Set $d = \lceil n/n_1 \rceil$, so that H has at least n vertices, and let $m = dn_1 - n$, which satisfies $m < n_1$. Remove any m vertices of V_1. The remaining graph H' has n vertices, and at least $(d-1)n_1 \geq n - n_1 > n/2$ vertices of degree δ. Then H' has at least $\delta n/4$ edges. □

The next proposition shows that Theorem 2.2.10 is optimal up to constant factors. It is straightforward to check that this proposition implies Corollary 1.3.16.

Proposition 5.1.11. *Let $\mathcal{F}$ be a hereditary class whose optimal adjacency labeling scheme has size $s(n)$ and which contains a graph with at least one edge. Then any adjacency labeling scheme for $\mathsf{her}(\mathcal{F}^{\square})$ has size at least $\Omega(s(n) + \log n)$, and any adjacency labeling scheme for $\mathsf{mon}(\mathcal{F}^{\square})$ has size at least $\Omega(s(n) + \delta(n) + \log n)$, where $\delta(n)$ is the maximum degeneracy of any n-vertex graph in $\mathsf{mon}(\mathcal{F}^{\square})$.*

Proof. Since $\mathcal{F} \subseteq \mathsf{her}(\mathcal{F}^{\square})$ and $\mathcal{F} \subseteq \mathsf{mon}(\mathcal{F}^{\square})$, we have a lower bound of $s(n)$ for the labeling schemes for both of these classes. Since $\mathcal{F}$ contains a graph G with at least one edge, the Cartesian products contain the class of hypercubes: $\mathsf{her}(\{K_2\}^{\square}) \subseteq \mathsf{her}(\mathcal{F}^{\square}) \subseteq \mathsf{mon}(\mathcal{F}^{\square})$. A labeling scheme for $\mathsf{her}(\{K_2\}^{\square})$ must have size $\Omega(\log n)$ (which can be seen since each vertex of K_2^d has a unique neighborhood and thus requires a unique label). This establishes the lower bound for $\mathsf{her}(\mathcal{F}^{\square})$, since the labels must have size $\max\{s(n), \Omega(\log n)\} = \Omega(s(n) + \log n)$. Finally, by Proposition 5.1.10, the number of n-vertex graphs in $\mathsf{mon}(\mathcal{F}^{\square})$ is at least $2^{\Omega(n\delta(n))}$, so there is a lower bound on the label size of $\Omega(\delta(n))$, which implies a lower bound of $\max\{s(n), \Omega(\log n), \Omega(\delta(n))\} = \Omega(s(n) + \delta(n) + \log n)$ for $\mathsf{mon}(\mathcal{F}^{\square})$. □

5.2 Impossibility Results for Equality-Based Sketches

In this section we give a characterization of the graph classes that admit constant-size equality-based labeling schemes (equivalently, constant-cost equality-based communication

protocols). Independently of our work, Hambardzumyan, Hatami, & Hatami [HHH21b] gave a different characterization of Boolean matrices (i.e. bipartite graphs) that admit constant-cost equality-based communication protocols: they show that any such matrix M is a linear combination of a constant number of adjacency matrices of bipartite equivalence graphs.

Our characterization applies to a bipartite transformation of a graph class, as follows. For any graph $G = (V, E)$, we define the colored bipartite graph

$$\mathsf{bip}(G) := (V_1, V_2, E')$$

where V_1 and V_2 are copies of V and $(v_1, v_2) \in E'$ is an edge in $\mathsf{bip}(G)$ if and only if $(v_1, v_2) \in E$. In other words, $\mathsf{bip}(G)$ is obtained by treating the (symmetric) adjacency matrix of G as the adjacency matrix for a bipartite graph instead.

It is convenient to put equality-based labeling schemes into a restricted form, which we call *diagonal*.

Definition 5.2.1 (Diagonal Labeling Scheme)**.** We call an (s, t, k)-equality-based labeling scheme *k-diagonal* if $s = 0$ and $t = 1$, so that the labels are of the form

$$(- \mid \vec{q}(x)),$$

and there is a function $\eta : \{0,1\}^k \to \{0,1\}$ such that the decoder satisfies, for all x, y,

$$D(Q_{x,y}) = \eta(Q_{x,y}(1,1), Q_{x,y}(2,2), \ldots, Q_{x,y}(k,k)).$$

It is possible to transform any constant-cost equality-based communication protocol for graphs $G \in \mathcal{F}$ into a diagonal labeling scheme for $\mathsf{bip}(\mathcal{F})$. We remark that it is *not* necessarily possible, in general, to get a diagonal labeling scheme for $\mathcal{F}$ itself (as opposed to $\mathsf{bip}(\mathcal{F})$). Bipartiteness allows us to achieve the symmetry required by diagonal labeling.

Lemma 5.2.2. *Let $\mathcal{F}$ be any hereditary graph class. If there is a constant-cost equality-based communication protocol for $\textsc{Adj}_{\mathcal{F}}$, then, for some constant k, $\mathsf{bip}(\mathcal{F})$ has a k-diagonal labeling scheme. As a consequence, if $\mathcal{F}$ is any hereditary graph class that admits an equality-based adjacency labeling scheme, then for some constant k, $\mathsf{bip}(\mathcal{F})$ admits a k-diagonal labeling scheme.*

Proof. We design a labeling scheme for $\mathsf{bip}(\mathcal{F})$ as follows. Write $d = \mathsf{CC}^{\mathrm{EQ}}(\textsc{Adj}_{\mathcal{F}})$, which is a constant. For any $G \in \mathcal{F}_n$, there is an equality-based communication tree T with depth

at most d that computes adjacency in G. Write $\mathsf{bip}(G) = (X, Y, E)$ where X, Y are copies of the vertex set of G.

Order the nodes of T such that each vertex precedes its children, and the subtree below the 0-valued edge precedes the subtree below the 1-valued edge. By Proposition 4.2.19 we may assume that all nodes in T are equality nodes. We may also assume that the tree is complete, and that leaf nodes alternate between 0 and 1 outputs in the order just defined. Let $(a_1, b_1), \ldots, (a_t, b_t)$ be the inner nodes of T (which are all equality nodes), in the order just defined.

- For each $x \in X$, define the equality codes $q_i(x) = a_i(x)$ for each $i \in [t]$. Set $q_{t+1}(x) = 0$.
- For each $y \in Y$, define the equality codes $q_i(x) = b_i(x)$ for each $i \in [t]$. Set $q_{t+1}(y) = 1$.

It holds that $t \leq 2^d$ since all trees have depth at most d. We define $\eta : \{0,1\}^{t+1} \to \{0,1\}$ as the function that, on input $w \in \{0,1\}^{t+1}$, outputs 0 if $w_{t+1} = 1$, and otherwise simulates the decision tree with node i having output w_i. (Note that we have assumed that all trees are complete, with depth d, with the same outputs on each leaf, so that the output of the tree is determined by the output of each node.) Then on input $x \in X, y \in Y$, we get

$$\begin{aligned}&\eta\left(\textsc{Eq}(q_1(x), q_1(y)), \ldots, \textsc{Eq}(q_t(x), q_t(y)), \textsc{Eq}(0,1)\right)\\ &\quad = \eta\left(\textsc{Eq}(a_1(x), b_1(y)), \ldots, \textsc{Eq}(a_t(x), b_t(y))\textsc{Eq}(0,1)\right) = T(x,y)\end{aligned}$$

which is 1 if and only if x, y is an edge in $\mathsf{bip}(G)$. On inputs $x, y \in X$ or $x, y \in Y$, we get

$$\eta\left(\textsc{Eq}(q_1(x), q_1(y)), \ldots, \textsc{Eq}(q_t(x), q_t(y)), \textsc{Eq}(0,0)\right) = 0$$

or

$$\eta\left(\textsc{Eq}(q_1(x), q_1(y)), \ldots, \textsc{Eq}(q_t(x), q_t(y)), \textsc{Eq}(1,1)\right) = 0$$

respectively, as desired. The final consequence in the statement of this lemma follows from Proposition 4.2.20. □

Recall that a graph G is an equivalence graph if it is the disjoint union of cliques, and a colored bipartite graph $G = (X, Y, E)$ is a bipartite equivalence graph if it is a colored disjoint union of bicliques.

Definition 5.2.3. For a constant $t \in \mathbb{N}$, we say that a class $\mathcal{F}$ of bipartite graphs is *t-equivalence interpretable* if there exists a function $\eta : \{0,1\}^t \to \{0,1\}$, such that the following holds. For every $G = (X, Y, E)$ in $\mathcal{F}$, there exists a vertex-pair coloring $\kappa : X \times Y \to \{0,1\}^t$, where:

1. For every $i \in [t]$, the graph (X, Y, E_i), where $E_i = \{(x, y) \in X \times Y : \kappa(x, y)_i = 1\}$, is a bipartite equivalence graph;
2. For every $x \in X, y \in Y$, $(x, y) \in E$ if and only if $\eta(\kappa(x, y)) = 1$.

Definition 5.2.4. For a constant $t \in \mathbb{N}$, we say that a class $\mathcal{F}$ of graphs is *strongly t-equivalence interpretable* if there exists a function $\eta : \{0,1\}^t \to \{0,1\}$, such that the following holds. For every $G = (V, E)$ in $\mathcal{F}$, there exists a vertex-pair coloring $\kappa : V \times V \to \{0,1\}^t$, where:

1. $\kappa(x, y) = \kappa(y, x)$ for every pair x, y;
2. For every $i \in [t]$, the graph (V, E_i), where $E_i = \{(x, y) : \kappa(x, y)_i = 1\}$, is an equivalence graph;
3. For every $x, y \in V$, $(x, y) \in E$ if and only if $\eta(\kappa(x, y)) = 1$.

Lemma 5.2.5. *A hereditary graph class $\mathcal{F}$ has a constant-size equality-based labeling scheme if and only if $\mathsf{bip}(\mathcal{F})$ is t-equivalence interpretable for some constant t.*

Proof. Suppose that $\mathcal{F}$ has a constant-size equality-based labeling scheme. By Proposition 4.2.20 and Lemma 5.2.2, $\mathsf{bip}(\mathcal{F})$ has a size t diagonal labeling for some constant t. Let $\eta : \{0,1\}^t \to \{0,1\}$ be the function in the diagonal labeling.

Let $G \in \mathcal{F}$ so that $\mathsf{bip}(G) \in \mathsf{bip}(\mathcal{F})$. Write $\mathsf{bip}(G) = (X, Y, E)$ where X, Y are copies of the vertices of G. Each vertex $x \in X \cup Y$ has a label of the form $(q_1(x), \ldots, q_t(x))$.

For each $i \in [t]$ and $x \in X, y \in Y$, define the color $\kappa(x, y)_i = \textsc{Eq}(q_i(x), q_i(y))$. Consider the graph with edges $(x, y) \in X \times Y$ if and only if $\kappa(x, y)_i = 1$. Let $x, x' \in X$ and $y, y' \in Y$ satisfy $\kappa(x, y)_i = \kappa(x', y)_i = \kappa(x', y')_i = 1$, so that (x, y, x', y') forms a path. Then $q_i(x) = q_i(y) = q_i(x') = q_i(y')$, so $\kappa(x, y')_i = 1$ and (x, y') is an edge. So this graph must be P_4-free; i.e. it is a bipartite equivalence graph.

Finally, it holds that for any $x \in X, y \in Y$,

$$\eta(\kappa(x, y)_1, \ldots, \kappa(x, y)_t) = \eta(\textsc{Eq}(q_1(x), q_1(y)), \ldots, \textsc{Eq}(q_t(x), q_t(y)))$$

which is 1 if and only if x, y is an edge in $\mathsf{bip}(G)$.

Now suppose that $\mathsf{bip}(\mathcal{F})$ is t-equivalence interpretable. It suffices to construct a labeling scheme for the graphs $\mathsf{bip}(G)$ for $G \in \mathcal{F}$. Write $\mathsf{bip}(G) = (X, Y, E)$ and let $\kappa(x, y) \in \{0, 1\}^t$ be the coloring of vertex pairs $x \in X, y \in Y$. For each $i \in [t]$ we let E_i be the edge set of the equivalence graph such that $(x, y) \in E_i$ if and only if $\kappa(x, y)_i = 1$. Give an arbitrary numbering to the bicliques in E_i and define $q_i(x)$ to be the number of the biclique to which x belongs. It then holds that for any $x \in X, y \in Y$,

$$\eta(\textsc{Eq}(q_1(x), q_1(y)), \ldots, \textsc{Eq}(q_t(x), q_t(y))) = \eta(\kappa(x, y)_1, \ldots, \kappa(x, y)_t)\,,$$

which is 1 if and only if (x, y) is an edge of $\mathsf{bip}(G)$. Therefore we have obtained a t-diagonal labeling scheme. □

Proposition 5.2.6. *Let $\mathcal{F}$ be a hereditary class of (uncolored) bipartite graphs. If $\mathsf{bip}(\mathcal{F})$ is t-equivalence interpretable then $\mathcal{F}$ is strongly $(t+1)$-equivalence interpretable.*

Proof. Let $\eta : \{0, 1\}^t \to \{0, 1\}$ be the function that witnesses $\mathcal{F}$ as t-equivalence interpretable and consider any bipartite graph $G \in \mathcal{F}$ with a fixed bipartition (X, Y) of its vertices. Then $\mathsf{bip}(G) = (X_1 \cup Y_1, X_2 \cup Y_2, E' \cup E'')$ is the disjoint union of $G' = (X_1, Y_2, E')$ and $G'' = (X_2, Y_1, E'')$, where G' and G'' are each isomorphic to G. By definition of equivalence-interpretable, there are graphs $B_1, \ldots, B_t$ with parts $X_1 \cup Y_1$ and $X_2 \cup Y_2$ where each B_i is a bipartite equivalence graph, and such that each pair $x \in X_1 \cup Y_1, y \in X_2 \cup Y_2$ is an edge of $\mathsf{bip}(G)$ if and only if

$$\eta(B_1(x, y), \ldots, B_t(x, y)) = 1,$$

where $B_i(x, y) = 1$ if (x, y) is an edge in B_i, and $B_i(x, y) = 0$ otherwise. Clearly, each B_i induces a bipartite equivalence graph when restricted the vertices of G'. We now consider graphs B'_i on vertex set $X_1 \cup Y_2$ where (x, y) is an edge if and only if x, y belong to the same biclique in B_i. B'_i is an equivalence graph since it is obtained by taking a disjoint union of bicliques and connecting every two vertices belonging to one of the bicliques. Now define the graph B'_{t+1} on vertices $X_1 \cup Y_2$ such that (x, y) is an edge if and only if $x, y \in X_1$ or $x, y \in Y_2$, so that B'_{t+1} is an equivalence graph. We define the function $\eta' : \{0, 1\}^{t+1} \to \{0, 1\}$ by setting $\eta'(w) = 0$ if $w_{t+1} = 1$ and otherwise setting $\eta'(w) = \eta(w_1, \ldots, w_t)$. It then holds that for every $(x, y) \in X_1 \times Y_2$,

$$\eta'\left(B'_1(x, y), \ldots, B'_t(x, y), B'_{t+1}(x, y)\right) = \eta\left(B_1(x, y), \ldots, B_t(x, y)\right)$$

which is 1 if and only if x, y are adjacent in G'. On the other hand, for $x, y \in X_1$ or $x, y \in Y_2$, we have $\eta'\left(B'_1(x, y), \ldots, B'_t(x, y), B'_{t+1}(x, y)\right) = 0$, so the same property holds. Consequently, G' is strongly $(t+1)$-equivalence interpretable. Since G is isomorphic to G', it is also strongly $(t+1)$-equivalence interpretable. □

5.2.1 Equality-Based Labeling Fails for the Hypercube

We now prove the following theorem from the introduction. This will follow from Theorem 5.2.9, which proves the same statement for equality-based labeling schemes. This is equivalent, by Proposition 4.2.20.

Theorem 1.3.17. *There is no constant-cost equality-based protocol for computing adjacency in K_2^d.*

The d-dimensional hypercube H_d is the d-wise Cartesian product $P_2^{\square d}$ of the single edge. The class $\mathcal{H} = \mathsf{her}(\{H_d : d \in \mathbb{N}\})$ of induced subgraphs of the hypercubes is sometimes called the class of *cubical* graphs. In this section, we will use Lemma 5.2.5 to prove that the class of cubical graphs does not admit an equality-based labeling scheme. In our proof, we will employ some results from the literature.

Theorem 5.2.7 ([ARSV06])**.** *For every k and $\ell \geq 6$, there exists $d_0(k,\ell)$ such that for every $d \geq d_0(k,\ell)$, every edge coloring of H_d with k colors contains a monochromatic* induced *cycle of length 2ℓ.*

For a graph G, its *equivalence covering number* $\mathsf{eqc}(G)$ is the minimum number k such that there exist k equivalence graphs $F_i = (V, E_i), i \in [k]$, whose union $(V, \cup_{i=1}^{k} E_i)$ coincides with G. We denote by C_n and P_n the cycle and the path on n vertices, respectively.

Theorem 5.2.8 ([LNP80, Alo86])**.** *For every $n \geq 3$, it holds that and $\mathsf{eqc}(\overline{C_n}), \mathsf{eqc}(\overline{P_n}) \geq \log n - 1$.*

For two binary vectors $x, y \in \{0,1\}^t$, we write $x \preceq y$ if $x_i \leq y_i$ for all $i \in [t]$, and we also write $x \prec y$ if $x \preceq y$ and $x \neq y$.

Theorem 5.2.9. *The class $\mathcal{H}$ does not admit a constant-size equality-based labeling scheme.*

Proof. Suppose, towards a contradiction, that $\mathcal{H}$ admits a constant-size equality-based labeling scheme. Then, since $\mathcal{H}$ is a class of bipartite graphs, by Lemma 5.2.5 and Proposition 5.2.6, there exists a t such that $\mathcal{H}$ is strongly t-equivalence interpretable.

Let $k = 2^t$, $\ell = 2^{t+1}$ and let $n \geq n_0(k,\ell)$, where $n_0(k,\ell)$ is the function from Theorem 5.2.7. Let V and E be the vertex and the edge sets of the hypercube H_n respectively. Let $\kappa : V \times V \to \{0,1\}^t$ and $\eta : \{0,1\}^t \to \{0,1\}$ be the functions as in Definition 5.2.4 witnessing that the hypercube H_n is strongly t-equivalence interpretable. Color every edge (a,b) of H_n with $\kappa(a,b)$. Since the edges of H_n are colored in at most $k = 2^t$ different

colors, by Theorem 5.2.7 it contains a monochromatic induced cycle $C = (V', E')$ of length $2\ell = 2^{t+2}$. Let $\kappa^* \in \{0,1\}^t$ be the color of the edges of C.

Claim 1. *For every distinct $a, b \in V'$ that are* not *adjacent in C, we have $\kappa^* \prec \kappa(a,b)$.*

Proof. Since every connected component of an equivalence graph is a clique, it follows that for every $i \in [t]$, $\kappa^*_i = 1$ implies $\kappa(x,y)_i = 1$ for every $x, y \in V'$. Hence, $\kappa^* \preceq \kappa(a,b)$. Furthermore, since a and b are not adjacent in C, we have that $\kappa(a,b) \neq \kappa^*$, as otherwise we would have $\eta(\kappa(a,b)) = \eta(\kappa^*) = 1$ and hence a and b would be adjacent. □

Now let $I \subseteq [t]$ be the index set such that $i \in I$ if and only if $\kappa^*_i = 0$ and there exist $a, b \in V'$ with $\kappa(a,b)_i = 1$. For every $i \in I$ let $F_i = (V', E'_i)$, where $E'_i = \{(a,b) \mid a, b \in V', \kappa(a,b)_i = 1\}$. Clearly, all these graphs are equivalence graphs. By construction and Claim 1, we have that the union $\cup_{i \in I} F_i$ contains none of the edges of C and contains all non-edges of C, in other words the union coincides with $\overline{C}$. Thus $\mathsf{eqc}(\overline{C}) \leq |I| \leq t$. However, by Theorem 5.2.8, $\mathsf{eqc}(\overline{C}) \geq \log |V'| - 1 \geq t + 1$, a contradiction. □

We now describe a generalization of this proof due to Bonamy & Girão, which was communicated to us by Louis Esperet. We require the following lemma:

Lemma 5.2.10. *For any $k, t, \ell \in \mathbb{N}$, there is an integer d such that if a $K_{t,t}$-free graph G has average degree at least d, and its edges are colored with at most k colors, then G contains a monochromatic path of length at least ℓ as an induced subgraph.*

Proof. We first claim that it suffices to consider the case $t = 2$. Suppose $t > 2$. It was proved in [KLST20] that every K_{2t}-free graph (including any $K_{t,t}$-free graph) of sufficiently large average degree contains a bipartite induced subgraph with large average degree. It was proved in [McC21] that any $K_{t,t}$-free bipartite graph of sufficiently large average degree contains a $K_{2,2}$-free induced subgraph of large average degree. Combining these results, we have that any $K_{t,t}$-free graph G of sufficiently large average degree contains an induced subgraph G' of large average degree which is also $K_{2,2}$-free. Therefore it suffices to consider the case $t = 2$.

Consider a $K_{2,2}$-free graph G with large average degree, whose edges are colored with at most k colors. Then there exists a color c such that the graph G_c induced by the edges of color c has large average degree $2d$, where d is an arbitrarily-large integer. Then G_c has an induced subgraph G'_c with *minimum* degree at least d. Since d may be arbitrarily large, we can get $d > \ell$.

We now construct a monochromatic induced path in G as follows. Suppose have have obtained a path $P_{t-1} = (v_1, \ldots, v_{t-1})$, for $t - 1 < \ell$ where each (v_i, v_{i+1}) is an edge of

G'_c. Let $N'_c(v_{t-1})$ be the neighbors of v_{t-1} in G'_c and suppose for contradiction that all vertices $u \in N'_c(v_{t-1}) \setminus P$ are adjacent in G to some v_i with $i < t-1$. Since v has at least $d > \ell > t-1$ neighbors, there are two vertices $u, w \in N'_c(v_{t-1}) \setminus P$ that are adjacent in G to both v_{t-1} and v_i, for some $i < t-1$. But then $\{v_i, v_{t-1}\}, \{u, w\}$ form an induced $K_{2,2}$. Therefore we may find some vertex $u \in N'_c(v_{t-1})$ which is not adjacent in G to any of the previous vertices of the path. Since this holds for any $t-1 < \ell$, therefore we may construct a path in G'_c of length ℓ which is an induced path in G. □

We then obtain the following result from Chapter 1, following the same reasoning as in Theorem 5.2.9.

Theorem 1.3.18. *For any $t \in \mathbb{N}$, if $\mathcal{F}$ is any class of bipartite graphs with no $K_{t,t}$ subgraph, then $\mathcal{F}$ admits a constant-size equality-based adjacency sketch if and only if it has bounded degeneracy.*

Chapter 6

Sketching for Monotone Graph Classes

As our abilities continue to expand,
we reach out with an extended hand,
towards a question yet more grand.
Our grasp exceeded, we understand,
that it was only weakly reachable!

The previous chapter established the basics of graph sketching. We introduced adjacency, small-distance, first-order (FO), and approximate distance threshold (ADT) sketches. The goal is to identify the hereditary graph classes which admit constant-size sketches of these types. In this chapter, we develop a theory of these sketching problems for the special case of *monotone* graph classes. Recall the questions from Chapter 1:

Question 1.3.8. *Which hereditary graph classes are adjacency sketchable?*

Question 1.3.9. *Which hereditary graph classes are small-distance sketchable?*

Question 1.3.10. *Which hereditary graph classes are ADT sketchable?*

Question 1.3.11. *What is the relationship, if any, between adjacency, small-distance, and ADT sketching?*

This chapter will answer Questions 1.3.8, 1.3.9, and 1.3.11, and make progress towards an answer of Question 1.3.10, for monotone classes of graphs. This chapter is derived from the paper [EHK22], coauthored with Louis Esperet and Andrey Kupavskii.

6.1 Preliminaries: Bounded Expansion

We require the notion of expansion from sparsity theory, as discussed in [NO12], and some of its equivalences stated in Theorem 6.1.5.

Definition 6.1.1 (Bounded Expansion)**.** Given a graph G and an integer $r \geq 0$, a *depth-r minor* of G is a graph obtained by contracting pairwise disjoint connected subgraphs of radius at most r in a subgraph of G. For any function f, we say that a class of graphs $\mathcal{G}$ has *expansion* at most f if any depth-r minor of a graph of $\mathcal{G}$ has average degree at most $f(r)$ (see [NO12] for more details on this notion). We say that a class $\mathcal{G}$ has *bounded expansion* if there is a function f such that $\mathcal{G}$ has expansion at most f.

Note that, for example, every proper minor-closed family has constant expansion.

Definition 6.1.2 (Weakly r-reachable)**.** Given a total order $(V, <)$ on the vertex set V of a graph G and an integer $r \geq 0$, we say that a vertex a vertex $v \in V$ is *weakly r-reachable* from a vertex $u \in V$ if there is a path of length at most r connecting v to u in G, and such that for any vertex w on the path, $v \leq w$ (in words, v is the smallest vertex on the path with respect to $(V, <)$). For a graph G and an integer $r \geq 0$, we denote by $\mathrm{wcol}_r(G)$ the smallest integer k for which the vertex set of G has a total order $(V, <)$ such that for any vertex $u \in V$, at most k vertices are weakly r-reachable from u with respect to $(V, <)$. For a graph class $\mathcal{F}$, we write $\mathrm{wcol}_r(\mathcal{F})$ for the supremum of $\mathrm{wcol}_r(G)$, for $G \in \mathcal{F}$.

Definition 6.1.3 ((k, ℓ)-Subdivisions)**.** For a graph G and two integers $0 \leq k \leq \ell$, a (k, ℓ)-subdivision of G is any graph obtained from G by subdividing each edge of G at least k times and at most ℓ times (i.e. , we replace each edge of G by a path with at least k and an most ℓ internal vertices). A (k, k)-subdivision is also called a k-subdivision for simplicity;

Definition 6.1.4 (Depth-r Topological Minor)**.** We say that H is a *depth-r topological minor* of a graph G if G contains a $(0, 2r)$-subdivision of H as a subgraph. In the proof below it will be convenient to use the following equivalent definition of bounded expansion [NO12].

Theorem 6.1.5. *For a class $\mathcal{F}$ of graphs, the following are equivalent:*

1. *$\mathcal{F}$ has bounded expansion.*
2. *There is a function $f : \mathbb{N} \to \mathbb{N}$ such that for any $r \in \mathbb{N}$, $\mathrm{wcol}_r(\mathcal{F}) \leq f(r)$.*

3. *There is a function $f : \mathbb{N} \to \mathbb{N}$ such that for any $r \in \mathbb{N}$ and any $G \in \mathcal{F}$, any depth-r topological minor of G has average degree at most $f(r)$.*

We will also require the following fact about the expansion of monotone classes, which is a simple consequence of Theorem 6.1.5 (see for instance [NO15]) combined with a result of Kühn & Osthus [KO04]. The *girth* of a graph G is defined as the size of a shortest cycle in G (if G is acyclic, its girth is infinite).

Corollary 6.1.6. *Let $\mathcal{F}$ be a monotone class of unbounded expansion. Then there is a constant $r \geq 0$, so that for any $d \geq 0$, $\mathcal{F}$ contains an r-subdivision of a bipartite graph of minimum degree at least d and girth at least 6.*

Proof. Since $\mathcal{F}$ has unbounded expansion, it follows from Theorem 6.1.5 that there exists an $r \geq 0$, such that depth-r topological minors of $\mathcal{F}$ have unbounded average degree. Since each edge in a depth-r topological minor is subdivided at most $2r$ times, if $\mathcal{F}$ contains a depth-r topological minor H of average degree at least d, $\mathcal{F}$ also contains a $2r'$-subdivision of a subgraph H' of H, for some $r' \leq r$, such that H' has average degree at least $\frac{d}{2r+1}$ (recall that $\mathcal{F}$ is monotone). It follows that there exists an integer $r'' \leq 2r$ such that for infinitely many d, $\mathcal{F}$ contains an r''-subdivision of a graph of average degree at least d. It was proved by Kühn and Osthus [KO04] that any graph of sufficiently large average degree contains a bipartite subgraph of large minimum degree and girth at least 6. As $\mathcal{F}$ is monotone, the desired result follows. □

6.2 Adjacency Sketching

In this section, we prove Theorem 1.3.19, and include the additional equivalent statement that $\mathcal{F}$ admits a constant-size *disjunctive* adjacency sketch. We think of disjunctive sketches as the simplest possible use of randomization in a sketch, with the theorem establishing that the simplest possible sketches are sufficient for monotone classes.

Theorem 6.2.1. *Let $\mathcal{F}$ be a monotone class of graphs. Then the following are equivalent:*

1. *$\mathcal{F}$ is adjacency sketchable.*
2. *$\mathcal{F}$ admits a constant-size disjunctive adjacency labeling scheme.*
3. *$\mathcal{F}$ has bounded arboricity.*

A disjunctive labeling scheme for graphs of arboricity k can be obtained from the adjacency labeling scheme of [KNR92], as in Example 4.2.11. This leads to a sketch of size $O(k \log k)$ by Proposition 4.2.12, which was improved slightly in [HWZ22]:

Proposition 6.2.2 ([HWZ22])**.** *Let $\mathcal{F}$ be any class with arboricity at most k. Then $\mathcal{F}$ admits a $(0, 1, k+1)$-disjunctive adjacency labeling scheme, and an adjacency sketch of size $O(k)$.*

Therefore, to prove Theorem 6.2.1, it suffices to prove (1) $\implies$ (3), which we will prove by contrapositive.

Consider a graph $G = (V, E)$, let $f : V \times V \to \{0, 1, *\}$ be a partial function, and let μ be a probability distribution over $V \times V$ that is supported on pairs (x, y) which satisfy $f(x, y) \neq *$. Let $X, Y \subseteq V$. Then we define the *discrepancy* of $R = X \times Y$ as

$$\mathrm{Disc}_{\mu,f}(G, R) = \left| \mathop{\mathbb{P}}_{\mu} \left[(x, y) \in R \cap f^{-1}(1)\right] - \mathop{\mathbb{P}}_{\mu} \left[(x, y) \in R \cap f^{-1}(0)\right] \right| ,$$

where (x, y) is drawn from μ. The discrepancy of G under μ is defined as

$$\mathrm{Disc}_{\mu,f}(G) = \max_{R} \mathrm{Disc}_{\mu,f}(G, R) ,$$

where the maximum is over all sets $R = X \times Y$ with $X, Y \subseteq V$. The following lemma is essentially a restatement of a standard lower-bound technique in communication complexity. For completeness (and because we are using sketches instead of communication protocols), we present a proof.

Lemma 6.2.3. *Let $G = (V, E)$ be any graph on n vertices, let $\mathcal{F}$ be any class of graphs containing G, and let f be a partial function parameterized by graphs in $\mathcal{F}$. Let μ be any probability distribution over $V \times V$ supported on a subset of $\{(x, y) : f_G(x, y) \neq *\}$. Then any f-sketch for $\mathcal{F}_n$ has size at least $\frac{1}{2} \log \frac{1}{3\mathrm{Disc}_{\mu,f}(G)}$.*

Proof. Let s denote the size of the adjacency sketch for $\mathcal{F}_n$. Let $D : \{0,1\}^s \times \{0,1\}^s \to \{0, 1\}$ be the decoder, and let $\mathsf{sk} : V \to \{0,1\}^s$ be the (random) adjacency sketch function for G. By definition, for any $x, y \in V$, we have

$$\mathbb{P}\left[D(\mathsf{sk}(x), \mathsf{sk}(y)) = f_G(x, y)\right] \geq 2/3 ,$$

so

$$\mathop{\mathbb{P}}_{\mathsf{sk}}\left[D(\mathsf{sk}(x), \mathsf{sk}(y)) = f_G(x, y)\right] - \mathop{\mathbb{P}}_{\mathsf{sk}}\left[D(\mathsf{sk}(x), \mathsf{sk}(y)) \neq f_G(x, y)\right] \geq 1/3 .$$

Using linearity of expectation, we get

$$\mathop{\mathbb{E}}_{sk}\left[\mathbb{1}\left[D(\mathsf{sk}(x),\mathsf{sk}(y)) = f_G(x,y)\right] - \mathbb{1}\left[D(\mathsf{sk}(x),\mathsf{sk}(y)) \neq f_G(x,y)\right]\right] \geq 1/3\,.$$

Taking the expectation over randomly chosen $(x,y) \sim \mu$, we have

$$\begin{aligned}
1/3 &\leq \mathop{\mathbb{E}}_{(x,y)\sim\mu}\mathop{\mathbb{E}}_{\mathsf{sk}}\left[\mathbb{1}\left[D(\mathsf{sk}(x),\mathsf{sk}(y)) = f_G(x,y)\right] - \mathbb{1}\left[D(\mathsf{sk}(x),\mathsf{sk}(y)) \neq f_G(x,y)\right]\right] \\
&= \mathop{\mathbb{E}}_{\mathsf{sk}}\mathop{\mathbb{E}}_{(x,y)\sim\mu}\left[\mathbb{1}\left[D(\mathsf{sk}(x),\mathsf{sk}(y)) = f_G(x,y)\right] - \mathbb{1}\left[D(\mathsf{sk}(x),\mathsf{sk}(y)) \neq f_G(x,y)\right]\right] \\
&= \mathop{\mathbb{E}}_{\mathsf{sk}}\left[\mathop{\mathbb{P}}_{(x,y)\sim\mu}[D(\mathsf{sk}(x),\mathsf{sk}(y)) = f_G(x,y)] - \mathop{\mathbb{P}}_{(x,y)\sim\mu}[D(\mathsf{sk}(x),\mathsf{sk}(y)) \neq f_G(x,y)]\right]\,.
\end{aligned}$$

Therefore, there exists a fixed, deterministic function $\ell : V \to \{0,1\}^s$ such that

$$1/3 \leq \mathop{\mathbb{P}}_{(x,y)\sim\mu}[D(\ell(x),\ell(y)) = f_G(x,y)] - \mathop{\mathbb{P}}_{(x,y)\sim\mu}[D(\ell(x),\ell(y)) \neq f_G(x,y)]\,.$$

Since the range of ℓ has size at most $S = 2^s$, we can partition $V = V_1 \cup \cdots \cup V_S$ such that ℓ is constant on each set V_i. For each $i,j \leq S$, write $R_{i,j} = V_i \times V_j$. Note that $D(\ell(x),\ell(y))$ takes the same value for each $(x,y) \in R_{i,j}$, so

$$\begin{aligned}
&\mathop{\mathbb{P}}_{(x,y)\sim\mu}[(x,y) \in R_{i,j} \wedge D(\ell(x),\ell(y)) = f_G(x,y)] \\
&\qquad - \mathop{\mathbb{P}}_{(x,y)\sim\mu}[(x,y) \in R_{i,j} \wedge D(\ell(x),\ell(y)) \neq f_G(x,y)] \\
&\leq \left|\mathop{\mathbb{P}}_{(x,y)\sim\mu}[(x,y) \in R_{i,j} \wedge f_G(x,y) = 1] - \mathop{\mathbb{P}}_{(x,y)\sim\mu}[(x,y) \in R_{i,j} \wedge f_G(x,y) = 0]\right| \\
&= \mathrm{Disc}_{\mu,f}(G, R_{i,j})\,.
\end{aligned}$$

Then, since the sets $R_{i,j}$ partition the pairs $V \times V$, we have

$$1/3 \leq \sum_{1\leq i,j\leq S} \mathrm{Disc}_{\mu,f}(G, R_{i,j}) \leq S^2 \cdot \mathrm{Disc}_{\mu,f}(G)\,.$$

We therefore must have

$$s = \log S \geq \log\sqrt{\frac{1}{3\cdot\mathrm{Disc}_{\mu,f}(G)}} = \frac{1}{2}\log\frac{1}{3\cdot\mathrm{Disc}_{\mu,f}(G)}\,,$$

as desired. □

A *spanning subgraph* of a graph $G = (V, E)$ is a subgraph of G with vertex set V. Our next lemma will give a lower bound on the adjacency sketch size for the class $\mathcal{G}$ of spanning sugraphs of a graph G of minimum degree d. We will actually prove the lower bound for a weaker type of adjacency sketch, which is only required to be correct on pairs (x, y) that were originally edges in G. This stronger statement is not necessary for the current section, but will be used in the proof of Theorem 6.3.11.

For a graph $G = (V, E)$ and the class $\mathcal{G}$ of spanning subgraphs of G, and any subgraph $H \in \mathcal{G}$, we will define the partial function $\mathrm{adj}_H^E : V \times V \to \{0, 1, *\}$ as

$$\mathrm{adj}_H^E(x, y) = \begin{cases} \mathrm{adj}_H(x, y) & \text{if } (x, y) \in E \\ * & \text{otherwise.} \end{cases}$$

We show by the probabilistic method that there is a distribution μ and a subgraph of G with discrepancy $O(1/\sqrt{d})$ with respect to μ. We will require the standard Chernoff bound for the binomial distribution with parameters n and $\frac{1}{2}$ (see Corollary A.1.2 in [AS16]): for any $t > 0$,

$$\mathbb{P}\left[|\mathrm{Bin}(n, \tfrac{1}{2}) - \tfrac{n}{2}| > t\right] < 2\exp(-2t^2/n).$$

Lemma 6.2.4. *Let $G = (V, E)$ be a graph of minimum degree d, and let $\mathcal{G}$ be the class of spanning subgraphs of G. Then any adj^E-sketch for $\mathcal{G}$ requires size at least $\Omega(\log d)$.*

Proof. Let $\mathbf{H}$ be a random spanning subgraph of G obtained by including each edge of G with independently with probability $1/2$. Note that $\mathbf{H} \in \mathcal{G}$ with probability 1. Let $m = |E|$ and let μ be the probability distribution over $V \times V$ such that for every $(x, y) \in V \times V$, we have $\mu((x, y)) = 1/m$ if $(x, y) \in E$, and $\mu((x, y)) = 0$ otherwise (so that μ is uniform over the edges of G). For simplicity, write Disc_μ for $\mathrm{Disc}_{\mu,f}$ where $f = \mathrm{adj}^E$.We will prove that $\mathrm{Disc}_\mu(\mathbf{H})$ is small, with nonzero probability over $\mathbf{H}$.

Consider a set $R = X \times Y$ with $X, Y \subseteq V$, and let $k \leq m$ be the number of edges (x, y) of G with $(x, y) \in R$. Let H be any subgraph of G with $|E(H) \cap R| = \ell \leq k$. Then

$$\begin{aligned} \mathrm{Disc}_\mu(H, R) &= \left| \mathop{\mathbb{P}}_{\mu}[(x, y) \in E(H) \cap R] - \mathop{\mathbb{P}}_{\mu}[(x, y) \in R \setminus E(H)] \right| \\ &= \left| \frac{\ell}{m} - \frac{k - \ell}{m} \right| = \frac{|2\ell - k|}{m}. \end{aligned}$$

For fixed $R = X \times Y$, it then holds that $\mathrm{Disc}_\mu(\mathbf{H}, R)$ is a random variable $\frac{|2\ell - k|}{m}$, where $\ell \sim \mathrm{Bin}(k, 1/2)$. Then, by the Chernoff bound, we have for any $\varepsilon > 0$ that

$$\mathbb{P}\left[\mathrm{Disc}_\mu(\mathbf{H}, R) > \varepsilon\right] = \mathbb{P}\left[\left|\mathrm{Bin}\left(k, \tfrac{1}{2}\right) - \tfrac{k}{2}\right| > \varepsilon m\right] \leq 2\exp\left(-2\varepsilon^2 m^2/k\right) \leq 2\exp\left(-2\varepsilon^2 m\right),$$

where the last inequality is due to $k \leq m$. There are at most 2^{2n} sets $R = X \times Y \subseteq V \times V$, so by the union bound,

$$\begin{aligned}\mathbb{P}\left[\exists R = X \times Y \subseteq V \times V : \mathrm{Disc}_\mu(\mathbf{H}, R) > \varepsilon\right] &\leq 2^{2n+1}\exp\left(-2\varepsilon^2 m\right) \\ &= \exp\left((2n+1)\ln(2) - 2\varepsilon^2 m\right).\end{aligned}$$

Now, since G has minimum degree d, we have $m \geq dn/2$. Setting $\varepsilon = \Omega\left(\frac{1}{\sqrt{d}}\right)$ with a sufficiently large implicit multiplicative constant, we get an upper bound on this probability of

$$\exp\left((2n+1)\ln(2) - 2\varepsilon^2 m\right) \leq \exp\left((2n+1)\ln(2) - 2\varepsilon^2 dn\right) < 1.$$

Therefore there exists a subgraph H with $\mathrm{Disc}_\mu(H) = O(1/\sqrt{d})$. Applying Lemma 6.2.3, we see that any adjacency sketch for $\mathcal{G}$ must have size at least $\Omega(\log(\sqrt{d})) = \Omega(\log d)$. □

We may now complete the proof of Theorem 6.2.1. We aim to prove (3) $\implies$ (1), which we will prove by contrapositive: i.e. that any class of unbounded arboricity has non-constant adjacency sketch size.

Lemma 6.2.5. *Let $\mathcal{F}$ be any monotone class of graphs with unbounded arboricity. Then $\mathcal{F}$ does not admit a constant-size adjacency sketch.*

Proof. It is well-known that the degeneracy of a graph is within factor 2 of the arboricity, so the degeneracy of $\mathcal{F}$ must also be unbounded. Then for any integer $d \in \mathbb{N}$, there is a graph $G \in \mathcal{F}$ with degeneracy at least d. By definition, G contains a subgraph H of minimum degree at least d. Let $\mathcal{G}$ be the class of spanning subgraphs of G, which satisfies $\mathcal{G} \subseteq \mathcal{F}$, since $\mathcal{F}$ is monotone. Then by Lemma 6.2.4, any adjacency sketch for $\mathcal{G}$ must have size at least $\Omega(\log d)$. Then for any integer d, we obtain a lower bound of $\Omega(\log d)$ on the size of an adjacency sketch for $\mathcal{F}$; it follows that any adjacency sketch for $\mathcal{F}$ is of non-constant size. □

6.3 Small-Distance Sketching

In this section we prove Theorem 1.3.20. As in Theorem 6.2.1 from the previous section, we refine the theorem by showing that the sketches are in fact disjunctive.

Theorem 6.3.1. *Let $\mathcal{F}$ be a monotone class of graphs. Then the following are equivalent:*

1. *$\mathcal{F}$ is small-distance sketchable.*
2. *For some function $f : \mathbb{N} \to \mathbb{N}$ and every $r \in \mathbb{N}$, $\mathcal{F}$ admits a disjunctive small-distance labeling scheme of size $f(r)$.*
3. *$\mathcal{F}$ is first-order sketchable.*
4. *$\mathcal{F}$ has bounded expansion.*

It holds by definition that (3) $\implies$ (1) and (2) $\implies$ (1), even without the assumption of monotonicity. We will prove (4) $\implies$ (3) and (4) $\implies$ (2) using different methods. We prove (4) $\implies$ (3) (again without the assumption of monotonicity) in Section 6.3.1 using the structural result of [GKN+20]. This proof does not give explicit bounds on the sketch size. (4) $\implies$ (2) is proved in Section 6.3.2 and gives explicit upper bounds on the sketch size. The final piece of the theorem, (1) $\implies$ (4), is proved in Section 6.3.3.

A consequence of Theorem 6.3.1 is that the set of small-distance sketchable monotone classes is a proper subset of the adjacency sketchable classes. We can also reach this conclusion with following simple but illustrative example.

Example 6.3.2. Consider any graph G. The *subdivision* G' of G is obtained by replacing each edge xy with two edges xz and zy, for a newly-added vertex z. The subdivision G' always has arboricity at most 2, and two vertices of G are adjacent if and only if they have distance 2 in G'. Therefore, if there was a constant-size distance-$(2, 2)$ sketch for the class of arboricity 2 graphs, we would obtain a constant-size adjacency sketch for the class of all graphs, which is a contradiction of Theorem 4.2.21.

6.3.1 Bounded Expansion Implies FO Labeling Schemes

To prove that any class of bounded expansion is first-order sketchable, we use the result of [GKN+20] that shows how to decompose any class of (structurally) bounded expansion into a number of graphs of bounded shrubdepth. We will require an adjacency sketch for classes of bounded shrubdepth, given below.

Adjacency Sketching for Bounded Shrubdepth

We must first define shrubdepth. A *connection model* for a graph G is a rooted tree T whose nodes are colored with a bounded number of colors such that:

- the vertices of G are the leaves of T; and
- for two vertices $u, v \in V(G)$, whether u and v are adjacent in G depends only on the colors of u and v in T, and the color of the lowest common ancestor of u and v in T.

To avoid ambiguity, we say G has *vertices* while T has *nodes*. Note that we can assume without loss of generality that all leaves are at the same distance from the root in T. A class $\mathcal{G}$ has *bounded shrubdepth* if there are some $d, k \in \mathbb{N}$ such that every $G \in \mathcal{G}$ has a connection model of depth d with colors in $[k]$ (we recall that the depth of a rooted tree T is the maximum number of vertices on a root-to leaf path in T).

Lemma 6.3.3. *Any class $\mathcal{G}$ of bounded shrub-depth admits a constant-size equality-based adjacency labeling scheme.*

Proof. Let d, k be such that any graph $G \in \mathcal{G}$ has a connection model T_G of depth d using color set $[k]$. We denote by $\varphi_G : [k]^3 \to \{0, 1\}$ the function such that if u has color a, v has color b, and the lowest common ancestor of u and v has color c in T_G, then u and v are adjacent in G if and only if $\varphi_G(a, b, c) = 1$. For every node u of T, write $\chi(u)$ for the color of u in the connection model.

We now construct our equality-based labels for G. For any vertex x, let $t_1(x)$, $t_2(x)$, $\ldots$, $t_d(x)$ be the leaf-to-root path for x, where $t_1(x) = x$ and $t_d(x)$ is the root of T. Then the label for x is the sequence $(\varphi_G \mid -), (\chi(t_1(x)) \mid t_1(x)), \ldots, (\chi(t_d(x)) \mid t_d(x))$. On inputs

$$(\varphi_G \mid -), (\chi(t_1(x)) \mid t_1(x)), \ldots, (\chi(t_d(x)) \mid t_d(x)),$$
$$(\varphi_G \mid -), (\chi(t_1(y)) \mid t_1(y)), \ldots, (\chi(t_d(y)) \mid t_d(y)),$$

the decoder operates as follows. It finds the smallest $i \in [d]$ such that $\mathbb{1}[t_i(x) = t_i(y)]$ and outputs $\varphi_G(\chi(t_1(x)), \chi(t_1(y)), \chi(t_i(x)))$.

The correctness of this labeling scheme follows from the fact that we will have $t_i(x) = t_i(y)$ if and only if the node $t_i(x) = t_i(y)$ is an ancestor of both x and y in T, so the smallest $i \in [d]$ such that $t_i(x) = t_i(y)$ identifies the lowest common ancestor of x and y. □

Structurally Bounded Expansion Implies First-Order Sketching

Following [GKN+20], we say that a class of graphs has *structurally bounded expansion* if it can be obtained from a class of bounded expansion by first-order (FO) transductions. We omit the precise definition of FO transductions in this thesis, as they are not necessary to our discussion, and instead refer the reader to [GKN+20]. We just note that a particular case of FO transduction is the notion of *FO interpretation*, which is of specific interest to us. Consider an FO formula $\phi(x, y)$ with two free variables and relational vocabulary $\Sigma = \{F, R_1, \ldots, R_k\}$ where F is symmetric of arity 2. We will say that a graph class $\mathcal{F}'$ is an FO interpretation of a graph class $\mathcal{F}$ with respect to ϕ if for any graph $G' = (V, E') \in \mathcal{F}'$ there is a graph $G = (V, E) \in \mathcal{F}$ and a Σ-structure with domain V where E is the interpretation of the symbol F, such that for any pair $u, v \in V$, $uv \in E'$ if and only if $\phi(u/x, v/y)$ evaluates to true. For instance, if $\phi(u/x, v/y)$ encodes the property $\mathsf{dist}_G(u, v) \leq r$ for some fixed integer $r \geq 1$ (which can be written as an FO formula), then the corresponding FO interpretation of the class $\mathcal{F}$ is the class of all graph powers $\{G^r \,|\, G \in F\}$. FO transductions are slightly more involved, as it is allowed to consider a bounded number of copies of a graph before applying the formula, and then it is possible to delete vertices. We will use the following structural result for classes of structurally bounded expansion, proved in [GKN+20].

Theorem 6.3.4 ([GKN+20])**.** *A class $\mathcal{G}$ of graphs has structurally bounded expansion if and only if the following condition holds. For every $p \in \mathbb{N}$, there is a constant $m = m(p)$ such that for every graph $G \in \mathcal{G}$, one can find a family $\mathcal{F}(G)$ of vertex subsets of G with $|\mathcal{F}(G)| \leq m$ and the following properties:*

- *for every $X \subseteq V(G)$ with $|X| \leq p$, there is $A \in \mathcal{F}(G)$ such that $X \subseteq A$; and*
- *the class $\{G[A] \,|\, G \in \mathcal{G}, A \in \mathcal{F}(G)\}$ of induced subgraphs has bounded shrubdepth.*

We directly deduce the following result.

Lemma 6.3.5. *Any class $\mathcal{G}$ of structurally bounded expansion admits a constant-size equality-based adjacency labeling scheme.*

Proof. Let m and $\mathcal{F}$ be given by applying Theorem 6.3.4 to $\mathcal{G}$ with $p = 2$. By definition, for every graph $G \in \mathcal{G}$ and every pair of vertices $u, v \in V(G)$, there is a set $A \in \mathcal{F}(G)$ containing u and v. Moreover, $\mathcal{F}(G)$ contains at most m sets and the family $\mathcal{C}$ of all graphs $G[A]$, for $G \in \mathcal{G}$, and $A \in \mathcal{F}(G)$, has bounded shrubdepth. It follows from Lemma 6.3.3

that there is a constant-size equality-based adjacency labeling scheme for $\mathcal{C}$. We denote the decoder of this scheme by D, and the corresponding labels as $\ell'_{G[A]}$.

Consider some graph $G \in \mathcal{G}$, and let $\mathcal{F}(G) = \{A_1, \ldots, A_m\}$. For each vertex x of G and $i \in [m]$, we write $a(x) = (a_1(x), \ldots, a_m(x))$ where $a_i(x) = \mathbb{1}[x \in A_i]$. Then we define the label for x by taking the prefix $a(x)$ and appending the labels $\ell'_{G[A_i]}(x)$ for each induced subgraph $G[A_i] \in \mathcal{C}$ to which x belongs. Given the labels for vertices x and y, the decoder finds any $i \in [m]$ such that $a_i(x) = a_i(y) = 1$; and outputs $D'(\ell'_{G[A_i]}(x), \ell'_{G[A_i]}(y))$. Such a number $i \in [m]$ always exists due to Theorem 6.3.4. The correctness of this labeling scheme follows from Theorem 6.3.4 and Lemma 6.3.3. □

Since FO-transductions compose (see e.g. [NOdMS22]), sketching FO formulas in a class of structurally bounded expansion is equivalent to sketching adjacency in another class of structurally bounded expansion. We obtain the following direct corollary of Lemma 6.3.5.

Corollary 6.3.6. *Any class $\mathcal{G}$ of structurally bounded expansion is first-order sketchable.*

As the property $\mathsf{dist}_G(x, y) \leq r$ can be written as an FO formula, this directly implies that classes of bounded expansion are small-distance sketchable. However, this does not tell anything on the size of the sketches as a function of r, unlike the approach using weak coloring numbers described in the next section.

6.3.2 Bounded Expansion Implies Small-Distance Sketching

Recall the definition of weak reachability from Definition 6.1.2. We give a quantitative bound on the small-distance sketch of any graph class $\mathcal{F}$ in terms of $\mathrm{wcol}_r(\mathcal{F})$. Recall from Theorem 6.1.5 that any class with bounded expansion has $\mathrm{wcol}_r(\mathcal{F}) \leq f(r)$ for some function $f(r)$; therefore we obtain the existence of small-distance sketches for any class of bounded expansion.

Theorem 6.3.7. *For any $r \in \mathbb{N}$, any class $\mathcal{F}$ has an $(0, r, \mathrm{wcol}_r(\mathcal{F}))$-disjunctive distance-$(r, r)$ labeling scheme.*

Proof. Let $G \in \mathcal{F}$, and consider a total order $(V, \prec)$ such that for any vertex $x \in V$, at most $\mathrm{wcol}_r(\mathcal{F})$ vertices are weakly r-reachable from v in G with respect to $(V, \prec)$. We say that vertex $y \in V$ has *x-rank k* if y is weakly k-reachable from x but not weakly $(k-1)$-reachable from x. For each vertex x and $k \in [r]$, write $S_k(x)$ for the set of vertices y with x-rank k.

We construct a disjunctive labeling scheme as follows. Each vertex x is assigned the label

$$(- \mid \vec{q}_1(x)), (- \mid \vec{q}_2(x)), \ldots, (- \mid \vec{q}_{r'}(x))$$

where $r' \leq r$ is the maximum number such that $S_{r'}(x) \neq \emptyset$, and the equality codes $\vec{q}_i(x)$ are names of vertices in the set $S_i(x)$. Each label contains at most $\mathrm{wcol}_r(G)$ equality codes, plus a constant number of bits per equality code and $O(r)$ bits to separate the elements of the list. Given labels for x and y, the decoder outputs 1 if and only if there exist $0 \leq i, j \leq r$ such that $i + j \leq r$ and $S_i(x) \cap S_j(y) \neq \emptyset$, which can be checked using the equality codes in $\vec{q}_i(x)$ and $\vec{q}_j(y)$.

Suppose that $\mathsf{dist}_G(x, y) \leq r$ and let $P \subseteq V(G)$ be a path of length $\mathsf{dist}_G(x, y)$. Let $z \in P$ be the minimal element of P with respect to $\prec$. Then z is weakly i-reachable from x and weakly j-reachable from y, for some values i, j such that $i + j \leq r$. Then $z \in S_i(x) \cap S_j(y)$, so the decoder will output 1 given the labels for x and y. On the other hand, if the decoder outputs 1, then there are values i, j such that $i + j \leq r$ and $S_i(x) \cap S_j(y) \neq \emptyset$. Let $z \in S_i(x) \cap S_j(y)$, so that z is weakly i-reachable from x and weakly j-reachable from y. Then $\mathsf{dist}_G(x, y) \leq \mathsf{dist}_G(x, z) + \mathsf{dist}_G(z, y) \leq i + j \leq r$. □

We noticed after proving this result that a similar idea was used in [GKS17, Lemma 6.10] to obtain sparse neighborhood covers in nowhere-dense classes.

We will need the following quantitative results for planar graphs and graphs avoiding some specific minor, due to [vdHOQ+17].

Theorem 6.3.8 ([vdHOQ+17])**.** *For any planar graph G, and any integer $r \geq 0$, $\mathrm{wcol}_r(G) \leq (2r + 1)\binom{r+2}{2} = O(r^3)$.*

Theorem 6.3.9 ([vdHOQ+17])**.** *For any integer $t \geq 3$, any graph G with no K_t-minor, and any integer $r \geq 0$, $\mathrm{wcol}_r(G) \leq \binom{r+t-2}{t-2}(t-3)(2r+2) = O(r^{t-1})$.*

In the proof of Theorem 6.3.7, the equality codes are just the names of vertices; so we can use $\lceil \log n \rceil$ bits to encode each of the $\mathrm{wcol}_r(\mathcal{F})$ equality codes to obtain an adjacency label. Then, combined with Proposition 4.2.12, we obtain the following corollary:

Corollary 6.3.10. *If a class $\mathcal{F}$ has bounded expansion, then $\mathcal{F}$ has a small-distance sketch of size at most $O(r + \mathrm{wcol}_r(\mathcal{F}) \log(\mathrm{wcol}_r(\mathcal{F})))$. If $\mathcal{F}$ is the class of planar graphs, then the sketch has size $O(r^3 \log r)$ and if $\mathcal{F}$ is the class of K_t-minor free graphs for some fixed integer $t \geq 3$, then the sketch has size $O(r^{t-1} \log r)$. Furthermore, $\mathcal{F}$ admits a distance-(r, r) labeling scheme of size $O(r + \mathrm{wcol}_r(\mathcal{F}) \log n)$; if $\mathcal{F}$ is the class of planar graphs, then the scheme has size $O(r^3 \log n)$ and if $\mathcal{F}$ is the class of K_t-minor free graphs, then the scheme has size $O(r^{t-1} \log n)$.*

6.3.3 Small-Distance Sketching Implies Bounded Expansion

To complete the proof of Theorem 6.3.1, we must show that any monotone class of graphs that is small-distance sketchable has bounded expansion, which we do by contrapositive. In fact, we will prove a stronger statement: even having a weaker $(r, 5r-1)$-distance sketch of size $f(r)$ implies bounded expansion.

Theorem 6.3.11. *Let $\mathcal{F}$ be a monotone class of graphs and assume that there is a function f such that for any $r \geq 1$, $\mathcal{F}$ has a $(r, 5r-1)$-distance sketch of size $f(r)$. Then $\mathcal{F}$ has bounded expansion.*

Proof. Assume for the sake of contradiction that $\mathcal{F}$ has unbounded expansion. By Corollary 6.1.6, there is a constant k such that for every $d \geq 0$, $\mathcal{F}$ contains a k-subdivision of some bipartite graph $G = (V, E)$ of minimum degree at least d and girth at least 6. Let $\mathcal{G}$ be the class consisting of the graph G, together with all its spanning subgraphs. By monotonicity, $\mathcal{F}$ contains k-subdivisions of all the graphs of $\mathcal{G}$.

Recall the definition of the partial function adj^E parameterized by graphs $H \in \mathcal{G}$, from the discussion preceding Lemma 6.2.4. We will show that the $(k+1, 5(k+1)-1)$-distance sketch of size $f(k+1)$ for $\mathcal{F}$ can be used to obtain a adj^E-sketch for $\mathcal{G}$, which must have size $\Omega(\log d)$ due to Lemma 6.2.4. This is a contradiction since we must have $f(k) = \Omega(\log d)$ for arbitrarily large d, whereas $f(k+1)$ is a constant independent of d.

Let H be any spanning subgraph of G and let $H^{(k)}$ denote the k-subdivision of H. Consider two vertices $u, v \in V(H) \subseteq V(G)$ that are adjacent in G. Observe that $\mathsf{dist}_{H^{(k)}}(u, v) = (k+1)\mathsf{dist}_H(u, v)$, and thus if u, v are adjacent in H then $\mathsf{dist}_{H^{(k)}}(u, v) \leq k+1$. Assume now that u, v are non-adjacent in H. Since u, v are adjacent in G, G has girth at least 6, and H is a spanning subgraph of G, it follows that in this case $\mathsf{dist}_H(u, v) \geq 5$, and thus $\mathsf{dist}_{H^{(k)}}(u, v) \geq 5(k+1)$. Therefore, by using the same decoder as the $(k+1, 5(k+1)-1)$-distance sketch for $\mathcal{F}$, and using the random sketch sk defined for G, we obtain an adj^E_H-sketch for H. This gives an adj^E-sketch for $\mathcal{G}$ of size $f(k+1)$. □

In our proof of Theorem 6.3.11 we have used Corollary 6.1.6, which is based on the result of [KO04], stating that every graph of large minimum degree contains a bipartite subgraph of girth at least 6 and large minimum degree. The following stronger statement was conjectured by Thomassen [Tho83].

Conjecture 6.3.12 ([Tho83])**.** *For every integer k, every graph of sufficiently large minimum degree contains a bipartite subgraph of girth at least k and large minimum degree.*

Note that if Conjecture 6.3.12 is true, it readily follows from our proof that the constant 5 in Theorem 6.3.11 can be replaced by an arbitrarily large constant. Compare this with Theorem 6.4.7 in the next section, where we prove this result for randomized labeling schemes whose label size is constant (independent or r).

6.4 Approximate Distance Threshold Sketching

In this section, we prove Theorems 1.3.21 and 1.3.23. Recall that a class $\mathcal{F}$ admits an α-ADT sketch of size $s(n)$ if for every $r \in \mathbb{N}$ there is a function $D_r : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$ such that every graph $G \in \mathcal{F}$ on n vertices admits a probability distribution over functions $\mathsf{sk} : V(G) \to \{0,1\}^{s(n)}$ such that, for all $x, y \in V(G)$,

$$\begin{aligned} \mathsf{dist}_G(u,v) \le r &\implies \mathbb{P}\left[D_r(\mathsf{sk}(u), \mathsf{sk}(v)) = 1\right] \ge 2/3\,, \\ \mathsf{dist}_G(u,v) > \alpha r &\implies \mathbb{P}\left[D_r(\mathsf{sk}(u), \mathsf{sk}(v)) = 0\right] \ge 2/3\,. \end{aligned}$$

We emphasize that the size of the sketch should not depend on r, especially when $s(n)$ is constant (unlike in the notion of small-distance sketches studied in previous sections). A desirable property of these sketches is that the decoder D_r does not depend on r either, so that $D_r = D_{r'} = D$ for every r, r'. We will call such sketches *distance-invariant*. We remark that any α-ADT sketch can be made distance-invariant by including the value of r in the sketch using $\log n$ bits, which is useful to keep in mind for some of the lower bounds below. However, our main goal is to determine when *constant-size* sketches are possible, and for this goal distance-invariance does not make any qualitative difference, as shown in the following simple proposition:

Proposition 6.4.1. *If $\mathcal{F}$ admits a constant-size α-ADT sketch, then $\mathcal{F}$ admits a constant-size distance-invariant α-ADT sketch.*

Proof. Let s be the size of the α-approximate distance sketch. Then it holds that for every $r \in \mathbb{N}$, the function D_r has domain $\{0,1\}^s \times \{0,1\}^s$. There are at most $2^{2^{2c}}$ functions $\{0,1\}^s \times \{0,1\}^s \to \{0,1\}$; therefore there are at most $2^{2^{2s}}$ distinct functions $D_r : \{0,1\}^s \times \{0,1\}^s \to \{0,1\}$. We obtain a distance-invariant α-approximate distance sketch as follows. For each $G \in \mathcal{F}$, each $r \in \mathbb{N}$, and each $u \in V(G)$, we sample sk_r from the distribution defined by the constant-size sketch with decoder D_r, and we construct $\mathsf{sk}'(u)$ by concatenating at most 2^{2s} bits to specify the function D_r. We then define the decoder $D : \{0,1\}^{2^{2s}+s} \times \{0,1\}^{2^{2s}+s} \to \{0,1\}$ on inputs $\mathsf{sk}'(u), \mathsf{sk}'(v)$ as $D_r(\mathsf{sk}(u), \mathsf{sk}(v))$, where D_r is the function specified by $\mathsf{sk}'(u)$. □

We will show that for monotone classes, ADT sketching implies small-distance sketching (and therefore adjacency sketching). Any class (monotone or not) that is adjacency sketchable contains at most $2^{O(n \log n)}$ graphs on n vertices, so any monotone ADT-sketchable class must satisfy this condition also. One may wonder if these conditions hold for non-monotone ADT-sketchable classes. The next simple example shows that this is not so.

Example 6.4.2. Consider the class of graphs obtained by choosing any graph G and adding an arbitrary path, with one endpoint connected to all vertices of G. The set of n-vertex graphs in this class contains all $(n-1)$-vertex graphs as induced subgraphs, so it has more than $2^{O(n \log n)}$ graphs and is not adjacency sketchable. But it is 2-ADT sketchable: every pair of vertices in G have distance at most 2 and are equidistant to all other vertices, so we may essentially reduce the problem to a single path. Here we have included a path instead of a single vertex so that the graph class has unbounded diameter.

6.4.1 Lower Bound for the Class of All Graphs

Recall that $\mathfrak{G}$ is the class of all graphs, and for an integer N, we will write $\mathfrak{G}_N$ for the class of all graphs with vertex set $[N]$. Recall that Theorem 4.2.21 proved an $\Omega(N)$ lower bound for adjacency sketching in $\mathfrak{G}$; we now prove the analogous result for ADT sketching. It follows from Theorem 3.4 in [TZ05] that for any r, $\mathfrak{G}$ admits an $(r, \alpha r)$-distance labeling scheme with labels of size $O(n^{2/\alpha} \log^{2-2/\alpha} n)$. In [TZ05] a lower bound was also given, but this was for deterministic approximate distance labels, which must allow an approximate computation of all distance. We give a stronger result (although the proof is nearly the same) that holds even for the case where we allow only a $(1, \alpha)$-distance sketch; our bound of $\Omega(n^{1/\alpha})$ has nearly the same dependence on α as the upper bound.

We will need the following classical result (see Lemma 15.3.2 in [Mat13] and the references therein).

Lemma 6.4.3 ([Mat13])**.** *For any $\ell \geq 2$ and $n \geq 2$, there is an n-vertex graph with at least $\frac{1}{9} n^{1+1/\ell}$ edges and without any cycle of length at most $\ell + 1$.*

The proof of the following lower bound is inspired by a seminal proof of Matoušek on non-embeddability of graph metrics in Euclidean space [Mat96] (see also Proposition 5.1 in [TZ05] for a closer application on approximate distance oracles).

Theorem 6.4.4. *For any $\alpha \geq 2$ and $n \geq 2$, there exists a class $\mathcal{F}$ of n-vertex graphs such that any distance-$(1, \alpha)$ labeling scheme for $\mathcal{F}$ requires labels of size at least $\frac{1}{9} n^{1/\alpha}$.*

Proof. For $\alpha \geq 2$ and $n \geq 2$, let G be an n-vertex graph with $m \geq \frac{1}{9}n^{1+\frac{1}{\alpha}}$ edges and without any cycle of length at most $\alpha + 1$ (given by Lemma 6.4.3). Consider a (deterministic) distance-$(1, \alpha)$ distance labeling scheme for the class of all spanning subgraphs of G. Let H be a subgraph of G. Note that for any edge $uv \in E(G)$, u and v are at distance 1 in H if $uv \in E(H)$, and are at distance greater than α otherwise (since G has no cycle of length at most $\alpha+1$). It follows that given the labels of u and v in H, the decoder outputs 1 if $uv \in E(G)$ and 0 otherwise. Consequently, for any two distinct subgraphs H_1, H_2 of G, the sequences of labels of the vertices $v_1, v_2, \ldots, v_n$ of G in H_1 and H_2 are distinct. As there are 2^m such subgraphs, some subgraph H of G is such that the sequence of labels of $v_1, v_2, \ldots, v_n$ in H takes at least $m \geq \frac{1}{9}n^{1+\frac{1}{\alpha}}$ bits, and thus some vertex of H has a label of size at least $\frac{1}{9}n^{1/\alpha}$, as desired. □

Due to Lemma 4.2.8, we obtain the following immediate corollary.

Corollary 6.4.5. *For any $\alpha \geq 2$ and $n \geq 2$, there exists a class $\mathcal{F}$ of n-vertex graphs such that any distance-$(1, \alpha)$ sketch for $\mathcal{F}$ requires labels of size $\Omega(n^{1/\alpha}/\log n)$.*

Note that in Theorem 6.4.4 and Corollary 6.4.5, we do not assume that the distance labeling scheme under consideration is distance-invariant (indeed, we only use the case $r = 1$ to obtain the lower bound).

Lower Bound for Bounded-Degree Graphs

We now prove that a monotone class may have bounded expansion but still have a lower bound of $n^{\Omega(1/\alpha)}$ on the α-ADT sketch size. This bound holds for the class of graphs of maximum degree 3, which has expansion exponential in r [NO12].

Write $\mathcal{F}_{n,3}$ for the class of all n-vertex graphs of maximum degree at most 3. We will need the following construction: Given an N-vertex graph G and an integer $\ell \geq 2\log N + 1$, let $G[\ell]$ be any graph obtained from G as follows: each vertex v of G is associated with a rooted balanced binary tree T_v in H, whose leaves are indexed by the neighbors of v in G (the trees T_v are balanced, so they have depth at most $\log N$). Then H consists in the disjoint union of all trees T_v, for $v \in V(G)$, together with paths connecting the leaf of T_v indexed by u to the leaf of T_u indexed by v, for any edge uv of G. The length of the path connecting these two leaves is such that the distance in H between the root of T_v and the root of T_u is precisely ℓ.

Theorem 6.4.6. *Assume that there is a real α such that the class $\mathcal{F}_{n,3}$ has a distance-invariant α-ADT sketch of size $s(n)$. Then for any $\varepsilon > 0$, we have $s(n) = \Omega\left(n^{\frac{1}{4\alpha}-\varepsilon}\right)$.*

Proof. Recall that $\mathfrak{G}$ is the class of all graphs, and $\mathfrak{G}_N$ is the class of all graphs on vertex set $[N]$. For a graph $G \in \mathcal{G}_N$, consider the graph $H := G[\ell]$ as defined above, with $\ell = \lceil 4 \log N \rceil$. We denote the root of each tree T_v in H by r_v (see the paragraph above for the definition of T_v). Observe that for any $u, v \in V(G)$,

$$\tfrac{\ell}{2} \cdot \mathsf{dist}_G(u,v) \leq (\ell - 2\log N)\mathsf{dist}_G(u,v) \leq \mathsf{dist}_H(r_u, r_v) \leq \ell \cdot \mathsf{dist}_G(u,v).$$

Note that $H \in \mathcal{F}_n$, with $n \leq N \cdot 2N + \binom{N}{2} \cdot 5 \log N \leq 8N^2 \log N$, for sufficiently large N. We construct a distance-$(1, 2\alpha)$ sketch for $\mathfrak{G}_N$ as follows. Let D be the decoder for the α-ADT sketch for $\mathcal{F}$. Given $G \in \mathfrak{G}_N$, the encoder computes a graph H as above. Since $H \in \mathcal{F}_n$, for any $r \in \mathbb{N}$ there is a probability distribution over functions $\mathsf{sk}_r : V(H) \to \{0,1\}^{s(n)}$ such that for all $u, v \in V(H)$:

$$\begin{aligned} \mathsf{dist}_H(u,v) \leq r &\implies \mathbb{P}\left[D(\mathsf{sk}_r(u), \mathsf{sk}_r(v)) = 1\right] \geq 2/3\,, \\ \mathsf{dist}_H(u,v) > \alpha r &\implies \mathbb{P}\left[D(\mathsf{sk}_r(u), \mathsf{sk}_r(v)) = 0\right] \geq 2/3\,. \end{aligned}$$

To obtain a sketch for G, draw $\mathsf{sk}_\ell : V(H) \to \{0,1\}^{s(n)}$ from the appropriate distribution, and assign to each vertex $v \in V(G)$ the value $\mathsf{sk}'(v) = \mathsf{sk}(r_v)$. We establish the correctness of this sketch as follows. Let $u, v \in V(G)$ and suppose that $\mathsf{dist}_G(u,v) \leq 1$. Then $\mathsf{dist}_H(r_u, r_v) \leq \ell$, so we have

$$\mathbb{P}\left[D(\mathsf{sk}_\ell(r_u), \mathsf{sk}_\ell(r_v)) = 1\right] \geq 2/3\,.$$

Now suppose $\mathsf{dist}_G(u,v) > 2\alpha$. Then $\mathsf{dist}_H(r_u, r_v) > \alpha\ell$, so we have

$$\mathbb{P}\left[D(\mathsf{sk}_\ell(r_u), \mathsf{sk}_\ell(r_v)) = 0\right] \geq 2/3\,.$$

We therefore have a $(1, 2\alpha)$-distance sketch for $\mathcal{F}_N$ of size $s(n)$. Assume for contradiction that $s(n) = O\left(n^{\frac{1}{4\alpha} - \varepsilon}\right)$ for some $\varepsilon > 0$. By Corollary 6.4.5, it must be that any distance-$(1, 2\alpha)$ sketch for $\mathcal{G}_N$ has size $\Omega(N^{1/2\alpha}/\log N)$. Therefore we must have $s(n) = \Omega(N^{1/2\alpha}/\log N)$, so $N^{1/2\alpha}/\log N = O\left(n^{\frac{1}{4\alpha} - \varepsilon}\right)$. But $n \leq 8N^2 \log N$ for sufficiently large N, so

$$\frac{N^{1/2\alpha}}{\log N} = O\left(N^{\frac{1}{2\alpha} - 2\varepsilon} \log^{\frac{1}{4\alpha} - \varepsilon} N\right),$$

which is a contradiction. □

6.4.2 ADT Sketching Implies Bounded Expansion

We now prove that if a monotone class $\mathcal{F}$ is ADT sketchable, then $\mathcal{F}$ has bounded expansion. This is an extension of a similar (unpublished) result for classes of bounded Assouad-Nagata dimension.

Theorem 6.4.7. *Let $\mathcal{F}$ be any monotone class of graphs that is α-ADT sketchable, for some $\alpha > 1$. Then $\mathcal{F}$ has bounded expansion.*

Proof. Let $\mathcal{F}$ have unbounded expansion, and suppose for the sake of contradiction that it admits an α-approximate distance sketch of constant size s. By Proposition 6.4.1, we may assume that the sketch is distance-invariant. Write $D : \{0,1\}^s \times \{0,1\}^s \to \{0,1\}$ for the decoder, and for every $G \in \mathcal{F}$ and integer r, write $\mathsf{sk}_{G,r} : V(G) \to \{0,1\}^s$ for the associated (random) sketch.

Since $\mathcal{F}$ has unbounded expansion, by Corollary 6.1.6, there exists an integer $p \geq 0$ such that for any integer t, $\mathcal{F}$ contains a p-subdivision of a graph of minimum degree at least t. Then, by a recent result of Liu & Montgomery [LM20], for any integer t there is an integer $k_t \geq 0$ such that $\mathcal{F}$ contains a k_t-subdivision of the complete graph K_t.

Recall that $\mathfrak{G}$ is the class of all graphs. We will design an α-approximate distance sketch for $\mathfrak{G}$. For any $t \in \mathbb{N}$, let $G \in \mathfrak{G}_t$. Then for k_t defined above, $\mathcal{F}$ contains a k_t-subdivision of the complete graph K_t. Since $\mathcal{F}$ is monotone, it also contains the k_t-subdivision $G^{(k_t)}$ of G. Now observe that, for any $u, v \in V(G) \subseteq V(G^{(t)})$ and integer r, we have

$$\begin{aligned} \mathsf{dist}_G(u,v) \leq r &\implies \mathsf{dist}_{G^{(t)}}(u,v) \leq (k_t+1)r \\ \mathsf{dist}_G(u,v) \geq \alpha r &\implies \mathsf{dist}_{G^{(t)}}(u,v) \leq \alpha(k_t+1)r\,. \end{aligned}$$

Therefore, with probability at least $2/3$ over the choice of $\mathsf{sk}_{G^{(t)},(k_t+1)r}$, we have

$$D(\mathsf{sk}_{G^{(t)},(k_t+1)r}(u), \mathsf{sk}_{G^{(t)},(k_t+1)r}(v)) = \begin{cases} 1 & \text{if } \mathsf{dist}_G(u,v) \leq r \\ 0 & \text{if } \mathsf{dist}_G(u,v) \geq \alpha r\,, \end{cases}$$

as desired; so $\mathfrak{G}$ admits and distance-invariant α-approximate distance sketch of constant size s. But this contradicts Theorem 6.4.4. □

A natural question is whether this can be proved directly, without using the theory of sparsity and the fairly involved result of Liu and Montgomery [LM20].

6.4.3 Lower Bound: Grids

The two-dimensional grid with crosses is a standard example of a class with bounded expansion [NOW12]. Let $\mathcal{G}_d$ denote the class of all finite subgraphs of the d-dimensional grid with all diagonals (the strong product of d paths). It is was proved in [Dvo21] that for even d, the expansion of $\mathcal{G}_d$ is $r \mapsto \Theta(r^{d/2})$, and in particular the expansion of $\mathcal{G}_2$ is $r \mapsto \Theta(r)$.

Lemma 6.4.8. *For any graph G of maximum degree 3, there exists a constant $k = k_G$ such that $\mathcal{G}_2$ contains the k-subdivision of G.*

Proof sketch. Observe first that $\mathcal{G}_2$ contains arbitrarily large complete graphs as minors (see Figure 6.1, left, where the clique minor is obtained by contracting each thick path into a single vertex). It follows that for any graph G, some graph of $\mathcal{G}_2$ contains G as a minor. Note that for a graph G of maximum degree at most 3, a graph H contains G as a minor if and only if H contains a subdivision of G. It follows that for any graph G of maximum degree 3, some sufficiently large grid H of $\mathcal{G}_2$ contains a subdivision of G as a subgraph. For a vertex u of G, we denote by $\pi(u)$ its image in H (in a copy of some subdivision of G in H), and for any edge uv in G, we denote the corresponding path of the subdivision of G in H between $\pi(u)$ and $\pi(v)$ by P_{uv}.

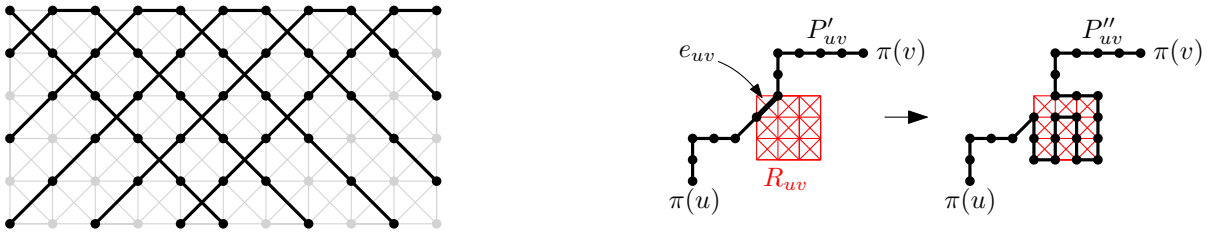


Figure 6.1: A large clique minor in a 2-dimensional grid with diagonals (left), and a way to increase the lengths of the path P'_{uv} using the private areas R_{uv} (right).

For $\lambda \geq 1$, a λ-*refinement* of the grid H is obtained by replacing each (1×1)-cell of the grid H by a $(\lambda \times \lambda)$-grid. Note that if H contains a subdivision of G as above, then any λ-refinement of H contains a subdivision of G, where each path P_{uv} is replaced by a path P'_{uv} of length $\lambda|P_{uv}|$. Note that after performing a λ-refinement, we can assume in addition that for any edge uv of G, H contains a $(\frac{\lambda}{2} \times \frac{\lambda}{2})$-subgrid R_{uv} that intersects the subdivision of G in H in a single edge e_{uv}, which is included in P'_{uv} (see Figure 6.1, right).

Using the subgrid R_{uv}, we can replace e_{uv} by a path of any length between 1 and $(\frac{\lambda}{2}-1)^2 \geq \lambda^2/16$ (assuming $\lambda \geq 4$) between its endpoints, turning P'_{uv} into a new path P''_{uv}

of length $|P'_{uv}| + \ell = \lambda|P_{uv}| + \ell$ (for any possible value of $0 \leq \ell \leq \lambda^2/16$), in such a way that the vertices $\pi(u)$, $u \in V(G)$, and the paths P''_{uv}, $uv \in E(H)$, still form a subdivision of G in H.

Note that a path P_{uv} of maximum length can replaced by a path P''_{uv} of length $\lambda|P_{uv}|$ after the λ-refinement, while a path P_{xy} of minimum length can be replaced by a path P''_{xy} of any length between $\lambda|P_{xy}|$ and $\lambda|P_{xy}| + \lambda^2/16 \geq \lambda|P_{uv}|$, where the inequality holds whenever $\lambda \geq 16|P_{uv}| - 16|P_{xy}|$. It follows that by taking λ sufficiently large, we obtain a subdivision of G in H where all edges of G correspond to paths of the same length in H, as desired. □

Using a similar argument to the above, we obtain a similar result for subgraphs of 3-dimensional grids. Write $\mathcal{P}^3$ for the class of subgraphs of the 3-dimensional grid; i.e. the class of finite subgraphs of the Cartesian product P^3, where P is the infinite path. We omit this proof due to its similarity to the one above.

Lemma 6.4.9. *For any graph G of maximum degree 3, there exists a constant $k = k_G$ such that $\mathcal{P}$ contains the k-subdivision of G.*

We easily deduce the following simple corollary.

Corollary 6.4.10. *The classes $\mathcal{G}_2$ and $\mathcal{P}$ are not ADT sketchable.*

Proof. The proof is similar to that of Theorem 6.4.7. Suppose for contradiction that, for some constant $\alpha > 1$, the class admits a constant-size α-approximate distance sketch. Then by Proposition 6.4.1, we can assume that the sketch is distance-invariant. By Lemma 6.4.8, this can be used to design an α-approximate distance sketch for the class of all graphs of maximum degree 3, contradicting Theorem 6.4.6. □

6.4.4 Lower Bound for Classes of Low Expansion

Now we show that there is no non-constant bound on the expansion that guarantees the existence of constant-size ADT sketches. We achieve this by constructing classes of graphs of arbitrarily low non-constant expansion, which cannot admit constant-size α-ADT sketches for any constant $\alpha > 1$. We start with a simple variant of [GKR+18, Theorem 4.5].

For a function $f : \mathbb{N} \to \mathbb{N}$ such that $f(n) \to \infty$ when $n \to \infty$, we define $f^{-1}(n) := \max\{k \mid f(k) \leq n\}$. We recall that for an N-vertex graph G and an integer $\ell \geq 2\log N + 1$, the graph $G[\ell]$ was defined just before the statement of Theorem 6.4.6.

Lemma 6.4.11. *Let f be a function such that $f(n) \to \infty$ when $n \to \infty$. For any n-vertex graph G, and any integer $r \geq 0$, every depth-r minor of $G[6f(6n^2) + 2\log n]$ has average degree at most $\max\{4, f^{-1}(r)\}$.*

Proof. Let $H = G[6f(6n^2) + 2\log n]$, and let H' be any depth-r minor of H. Observe that $G[2\log n + 1]$ has at most $2n^2 + \binom{n}{2} \leq 3n^2$ vertices, and thus tree-width at most $3n^2$. The graph H itself is obtained from $G[2\log n + 1]$ by subdividing some edges, an operation that leaves the tree-width unchanged. It follows that H also has tree-width at most $3n^2$. Since tree-width is a minor-monotone parameter, H' also has tree-width at most $3n^2$ and is thus $3n^2$-degenerate. It follows that H' has average degree at most $6n^2$. If $r \geq f(6n^2)$, then $6n^2 \leq f^{-1}(r)$ and thus H' has average degree at most $f^{-1}(r)$, as desired. Assume now that $r \leq f(6n^2)$. In this case, since H' is obtained from disjoint trees by connecting their leaves with paths of length at least $6f(6n^2)$, it can be checked that H' is 2-degenerate, and thus has average degree at most 4. □

Theorem 6.4.12. *For any function ρ tending to infinity, there exists a monotone class of expansion $r \mapsto \rho(r)$ that is not ADT sketchable. Moreover, for any $\varepsilon > 0$, there exists a monotone class $\mathcal{F}$ of expansion $r \mapsto O(r^\varepsilon)$, such that, if $\mathcal{F}$ admits an α-ADT sketch of size $s(n)$, then we must have $s(n) = n^{\Omega(1/\alpha)}$.*

Proof. Let $\rho : \mathbb{N} \to \mathbb{N}$ be function tending to infinity, so that ρ^{-1} is a non-decreasing function tending to infinity. We proceed as in the proof of Theorem 6.4.6, setting $\ell(N) = 6\rho^{-1}(6N^2) + 2\log N$ (instead of $\ell = \lceil 4\log N \rceil$). For any N-vertex graph G, $G[\ell(N)] \in \mathcal{F}_{n,3}$ with $n \leq 6N^2(\rho^{-1}(6N^2) + 2\log N)$. By Lemma 6.4.11, any depth-r minor of such a graph $G[\ell(N)]$ has average degree at most $\max\{4, \rho(r)\}$. It follows that the monotone class $\mathcal{F}$ of all graphs $G[\ell(N)]$ for $G \in \mathfrak{G}_N$ and their subgraphs has expansion at most $r \mapsto \max\{4, \rho(r)\}$.

By the same argument as in Theorem 6.4.6, if there is a distance-invariant α-ADT sketch for $\mathcal{F}$ of size $s(n)$ (which is a non-decreasing function in n), we obtain a $(1, 2\alpha)$-distance sketch for $\mathfrak{G}$ of size $N \mapsto s(n)$. Then, due to Corollary 6.4.5,

$$N^{\frac{1}{2\alpha}} / \log N = O\left(s(n)\right) = O\left(s\left(6N^2(\rho^{-1}(6N^2) + 2\log N)\right)\right).$$

It is clear that, for any choice of ρ, we cannot have $s(n)$ constant, which establishes the first part of the theorem. To get the second part, let $\varepsilon > 0$ and suppose that we choose $\rho(r) = r^\varepsilon$ so that $\rho^{-1}(r) = r^{1/\varepsilon}$, and assume for contradiction that $s(n) = n^{o(1/\alpha)}$. Then

$$N^{\frac{1}{2\alpha}} / \log N = O\left(s\left(6N^2((6N^2)^{1/\varepsilon} + 2\log N)\right)\right) = O\left(N^{o(1/\alpha)}\right),$$

which means we must have $s(n) = n^{\Omega(1/\alpha)}$ as desired. □

6.4.5 Upper Bounds

The *weak diameter* of a subset S of vertices of a graph G is the maximum distance in G between two vertices of S. Given a graph G, a (σ, τ, D)-*sparse cover* is a family $\mathcal{C}$ of subsets of $V(G)$ of weak diameter at most D, such that (i) for each $u \in V(G)$, there is a set $C \in \mathcal{C}$ such that $B(u, \frac{D}{\sigma}) \subseteq C$ (where $B(u, r)$ denotes the ball of radius r centered in u), and (ii) each vertex $u \in V(G)$ lies in at most τ sets of $\mathcal{C}$.

We say that a graph G admits a (σ, τ)-*sparse cover scheme* if for any D, it admits a (σ, τ, D)-sparse cover. We say that a graph class $\mathcal{F}$ has a (σ, τ)-sparse cover scheme if any graph of $\mathcal{F}$ has such a scheme[1]). Classes of graphs with (σ, τ)-sparse cover schemes are also known as classes of *Assouad-Nagata dimension* at most $\tau - 1$ in metric geometry [Ass82] (see also [LS05]).

The following result is a simple consequence of the definition of sparse covers. Note that here the size of the labels is independent of r, in contrast with the setting of Theorem 6.3.7 and its corollaries.

Theorem 6.4.13. *If a graph class $\mathcal{F}$ has a (σ, τ)-sparse cover scheme, then for any $r > 0$, $\mathcal{F}$ has distance-invariant, disjunctive σ-ADT labeling scheme with labels of size $O(\tau)$.*

Proof. By the definition of sparse covers, for any $r > 0$, there is a family $\mathcal{C}$ of subsets of $V(G)$ of weak diameter at most σr, such that (i) for each $u \in V(G)$, there is a set $C \in \mathcal{C}$ such that $B(u, r) \subseteq C$, and (ii) each vertex $u \in V(G)$ lies in at most τ sets of $\mathcal{C}$.

We may now define a disjunctive σ-ADT labeling scheme. Assign each set $C \in \mathcal{C}$ a unique number in $\mathbb{N}$, and for each vertex x, let $S(x)$ be the set of names of the (at most τ) sets C containing x. For each vertex $x \in V(G)$, the equality code $\vec{q}(x)$ contains the names $S(x)$, and we assign x the label $(- \mid \vec{q}(x))$. On inputs $(- \mid \vec{q}(x))$ and $(- \mid \vec{q}(y))$, the decoder outputs 1 if and only $S(x) \cap S(y) \neq \emptyset$ (which can be checked using the equality codes).

Suppose $\mathsf{dist}_G(x, y) \leq r$. Since there is a set $C \in \mathcal{C}$ such that $B(x, r) \subseteq C$, we also have $y \in C$, and thus $S(x) \cap S(y) \neq \emptyset$. Now suppose that $\mathsf{dist}_G(x, y) > \sigma r$. Since each set $C \in \mathcal{C}$ has weak diameter at most σr, there is no set $C \in \mathcal{C}$ containing both x and y and thus $S(x) \cap S(y) = \emptyset$. □

Using results of [Fil20] on sparse covers (based on [KPR93, FT03]), we deduce the following immediate corollary.

[1]It is usually assumed that in addition, such schemes can be computed efficiently, that is in time polynomial in the size of the graph.

Corollary 6.4.14. *For any $t \geq 4$, the class of K_t-minor free graphs has a distance-invariant $O(2^t)$-ADT sketch of size $O(t^2 \log t)$.*

Padded Decompositions

We will see now how to get improved sketches using *padded decompositions*. These improved sketches have some disadvantages: they have two-sided error, and they are not equality-based. For a graph G, a probability distribution $\mathcal{P}$ over partitions of $V(G)$ is said to be (β, δ, D)-*padded* if

- for each partition P in the support of $\mathcal{P}$, each set of P has diameter at most D, and
- for any $u \in V(G)$ and any $0 \leq \gamma \leq \delta$, the ball $B(u, \gamma D)$ is included in some set of a random partition from $\mathcal{P}$ with probability at least $2^{-\beta\gamma}$.

We say that a graph G admits a (β, δ)-*padded decomposition scheme* if for any D, it admits a (β, δ, D)-padded distribution over partitions of $V(G)$. We say that a graph class $\mathcal{F}$ has a (β, δ)-padded decomposition scheme if any graph of $\mathcal{F}$ has such a scheme.

Theorem 6.4.15. *If a graph class $\mathcal{F}$ has a (β, δ)-padded decomposition scheme, then $\mathcal{F}$ admits a distance-invariant α-ADT sketch of size 2, where $\alpha = \max(\frac{1}{\delta}, \frac{\beta}{\log(3/2)})$.*

Proof. Fix some $r > 0$, and some graph $G \in \mathcal{F}$. Let $\gamma = \min(\delta, \frac{1}{\beta}\log(3/2))$. Note that $0 \leq \gamma \leq \delta$ and $2^{-\beta\gamma} \geq \frac{2}{3}$. Let $D = r/\gamma$, and let $\mathcal{P}$ be a (β, δ, D)-padded distribution over partitions of $V(G)$. Let $\mathbf{P}$ be a partition of $V(G)$ drawn according to the distribution $\mathcal{P}$. Then each set of $\mathbf{P}$ has diameter at most $D = r/\delta$. Assign a random identifier $\mathrm{id}(S)$ to each set $S \in \mathbf{P}$, drawn uniformly at random from the set $\{1, 2, 3\}$. The label of each vertex $u \in V(G)$ simply consists of the identifier $\mathrm{id}(S)$ of the unique set $S \in \mathbf{P}$ such that $u \in S$. Given the labels of u and v, the decoder outputs 1 if and only the labels are equal. Note that the decoder is clearly distance-invariant.

Assume first that $d(u, v) \leq r$, then since v is in the ball of radius $r = \gamma D$ centered in u, it follows that u and v are in the same set $S \in \mathbf{P}$ with probability at least $2^{-\beta\gamma} \geq \frac{2}{3}$. If u and v are in the same set $S \in \mathbf{P}$, then their labels are equal with probability 1. It follows that if $d(u, v) \leq r$, the decoder outputs 1 with probability at least $2/3$, as desired.

Assume now that $d(u, v) > r/\gamma$. Since each set in $\mathbf{P}$ has diameter at most $D = r/\gamma$, it follows that u and v are in different sets of $\mathbf{P}$ with probability 1. As each set $S \in \mathbf{P}$ is assigned a random element from $\{1, 2, 3\}$, u and v have the same label with probability

$\frac{1}{3}$. It follows that if $d(u,v) > r/\gamma$, the decoder outputs 0 with probability $1 - \frac{1}{3} = \frac{2}{3}$, as desired. □

Although this sketch *does* use randomization for an equality check, it also uses randomization to construct the padded decomposition, and so it is not equality-based.

It was proved in [AGG+19] that for $t \geq 4$, the class of K_t-minor free graphs admits a $(320t, \frac{1}{160})$-padded decomposition scheme. It was also proved in [LS10] (see also [AGG+19]) that for any $g \geq 0$, the class of graphs embeddable on a surface of Euler genus g admits a $(O(\log g), \Omega(1))$-padded decomposition scheme. We obtain the following two corollaries, and again emphasize that these sketches have two-sided error.

Corollary 6.4.16. *For any $t \geq 4$, the class of K_t-minor free graph has a distance-invariant $O(t)$-ADT sketch with labels of at most 2 bits.*

Corollary 6.4.17. *For any $g \geq 0$, the class of graphs embeddable on a surface of Euler genus g has a distance-invariant $O(\log g)$-ADT sketch with labels of at most 2 bits.*

Observe that padded decomposition schemes must include the entire ball $B(u, \gamma D)$ in a random partition with sufficiently large probability. This is not necessary for the purposes of sketching: we only require that any two points u, v of distance at most r are included in a random partition. So a weaker notion that also implies the existence of an α-ADT sketch is the following. We say that a graph G has a (σ, τ, D)-*padded cover* if there is a probability distribution $\mathcal{C}$ on the covers of $V(G)$ such that

- for each cover C in the support of $\mathcal{C}$, each set of C has diameter at most D and each vertex $u \in V(G)$ lies in at most τ sets of C, and
- for any $u, v \in V(G)$ with $d(u,v) \leq \frac{D}{\sigma}$, the pair $\{u, v\}$ is included in some set of a random cover from $\mathcal{C}$ with probability at least $\frac{2}{3}$.

We say that a graph G admits a (σ, τ)-*padded cover scheme* if for any D, it admits a (σ, τ, D)-padded cover. We say that a graph class $\mathcal{F}$ has a (σ, τ)-padded cover scheme if any graph of $\mathcal{F}$ has such a scheme.

Note that there is nothing special about the value $\frac{2}{3}$ in the definition above. Up to sampling multiple times from $\mathcal{C}$ (and thus multiplying τ by a constant), this value can be replaced by any positive constant. The proofs of Theorems 6.4.13 and 6.4.15 can easily be combined to give the following result.

Theorem 6.4.18. *If a graph class $\mathcal{F}$ has a (σ, τ)-padded cover scheme, then $\mathcal{F}$ has a distance-invariant σ-ADT sketch with labels of size $O(\tau \log \tau)$.*

A diagram showing the relation between the notions of sparse covers, padded decompositions, and padded covers is depicted in Figure 6.2.

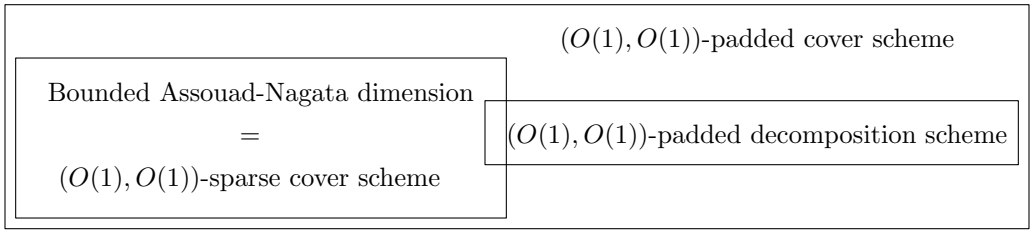


Figure 6.2: Relation between the different notions that imply ADT sketchability.

A natural question is the following: *Which graph classes have (σ, τ)-padded cover schemes, for constant σ and τ?* Is there a significant difference with the class of graphs that admit (σ, τ)-sparse cover schemes? And a significant difference with classes that admit σ-ADT sketches?

We remark that lower bounds for sketching therefore imply lower bounds on sparse covers, padded decompositions, and padded covers. For example, the communication complexity lower bound of [AK08] implies that bounded-degree expanders do not admit padded covers or sparse covers, which is something that we were unable to prove directly.

Chapter 7

Graph Sketching Beyond Monotone Classes

The king was asked, which number is greater?
By his highness's portrait's creator.
As the king boasted, "this is easy to do",
the artist cried "I cannot sketch this for you".
Although the artist was certainly able,
sadly the king was just simply unstable.

Having developed a complete theory of adjacency, small-distance, and first-order sketching for monotone classes of graphs, we now turn our attention to non-monotone classes. Our goal is to study a diverse set of graph classes, to get some idea of what conditions are sufficient for constant-size adjacency sketches. The results in this chapter are from [HWZ22], coauthored with Sebastian Wild and Viktor Zamaraev.

Recall the definition of the *chain number* from Section 1.3.5 and the discussion on its relation to the lattice of hereditary graph classes from Section 4.2.7. We repeat the definition here for convenience. For a graph G, the *chain number* $\mathsf{ch}(G)$ is the maximum number k for which there exist disjoint sets of vertices $\{a_1, \dots, a_k\}, \{b_1, \dots, b_k\} \subseteq V(G)$ such that $(a_i, b_j) \in E(G)$ if and only if $i \leq j$. For a graph class $\mathcal{F}$, we write $\mathsf{ch}(\mathcal{F}) = \max_{G \in \mathcal{F}} \mathsf{ch}(G)$. If $\mathsf{ch}(\mathcal{F}) = \infty$, then $\mathcal{F}$ has *unbounded chain number*, otherwise it has *bounded chain number*. We also refer to classes with bounded chain number as *stable*.

We will see in this chapter three examples of hereditary graph classes which do not admit adjacency sketches because they are not stable, but when we enforce the stability condition

they become sketchable. (A fourth example from the paper [HWZ22] was the class of graphs with bounded *twin-width*, which we do not include here because it is superceded by our result for classes with *structurally bounded expansion* in Chapter 6.)

Our first two examples are the interval and permutation graphs, which are two of the most well-studied geometric intersection graph classes. Here we give explicit upper bounds on the size of the adjacency sketches in terms of the chain number.

Third, we study the *monogenic* classes of bipartite graphs, which are those obtained by excluding a single bipartite graph H as an induced subgraph. We remark that to understand constant-size adjacency sketches, it is sufficient to consider bipartite graphs. Any hereditary class of bipartite graphs is defined by a unique set $\mathcal{H}$ of *forbidden induced subgraphs* (which may be infinite). This motivates our choice to study the classes obtained by forbidding a single induced subgraph.

7.1 Interval and Permutation Graphs

Interval graphs are the intersection graphs of intervals on the real line, and are arguably the most studied class of geometric intersection graphs. It is fairly easy to see that interval graphs are not stable, so by Lemma 4.2.23 any adjacency sketch for this class has size $\Omega(\log n)$, and there is a simple, matching upper bound of $O(\log n)$ achieved by deterministic adjacency labeling [KNR92] (see Example 1.3.3).

Definition 7.1.1 (Interval graph)**.** A graph G is an *interval graph* if there exists an *interval realization* $\ell, r : V(G) \to \mathbb{R}$ with $\ell(v) \leq r(v)$ for all $v \in V(G)$ so that $u, v \in V(G)$ are adjacent in G if and only if $[\ell(u), r(u)] \cap [\ell(v), r(v)] \neq \emptyset$.

Permutation graphs are another important factorial class of geometric intersection graphs (like interval graphs, they are a subclass of line segment intersection graphs). They admit a straightforward $O(\log n)$-bit adjacency labeling scheme that follows from their definition, and it is also simple to show that they are not stable, so there is a matching $\Omega(\log n)$ lower bound on their adjacency sketch size.

Definition 7.1.2 (Permutation Graph)**.** A graph G is a *permutation graph* on n vertices if each vertex can be identified with a number $i \in [n]$, such that there is a permutation π of $[n]$ where i, j are adjacent if and only if $i < j$ and $\pi(i) > \pi(j)$.

Our goal is to determine which hereditary graph classes admit constant-size adjacency sketches. Neither of these classes admit constant-size sketches. Therefore it is informative

to ask how much these classes must be restricted before constant-size sketches become possible. In other words, we would like to know which hereditary *subclasses* of interval and permutation graphs are adjacency sketchable. We find that in both cases it is exactly the *stable* subclasses.

Theorem 1.3.25. *Let $\mathcal{F}$ be any hereditary subclass of interval or permutation graphs. Then $\mathcal{F}$ is adjacency sketchable if and only if it is stable.*

This theorem follows from Theorem 7.1.7 (interval graphs) and Theorem 7.1.19 (permutation graphs), proved below.

7.1.1 Interval Graphs

The proof will rely on bounding the clique number of interval graphs with bounded chain number.

Lemma 7.1.3. *Let $\mathcal{F}$ be a class of interval graphs with bounded clique number, i. e. there is a constant c so that for any graph $G \in \mathcal{F}$, the maximal clique size is at most c. Then $\mathcal{F}$ admits a constant-size equality-based labeling scheme.*

Proof. Any interval graph is *chordal* and the treewidth of a chordal graph is one less its clique number. It follows that any interval graph G with clique number at most c has treewidth at most $c-1$. Graphs of treewidth $c-1$ have arboricity at most $O(c)$, and therefore, by Lemma 4.2.16, $\mathcal{F}$ admits a constant-size equality-based labeling scheme and an adjacency sketch of size $O(c)$. □

It is not possible in general to bound the clique number of interval graphs with bounded chain number, because there may be an arbitrarily large set of vertices realized by identical intervals, which forms an arbitrarily large clique. Our first step is to observe that, for the purpose of designing an equality-based labeling scheme, we can ignore these duplicate vertices (called *true twins* in the literature).

Definition 7.1.4. For a graph $G = (V, E)$, two distinct vertices x, y are called *twins* if $N(x) \setminus \{y\} = N(y) \setminus \{x\}$, where $N(x)$, $N(y)$ are the neighbourhoods of x and y in G. Twins x and y are *true twins* if they are adjacent, and *false twins* if they are not adjacent. The false-twin relation and true-twin relation are equivalence relations.

A graph is *true-twin-free* if it does not contain any vertices x, y that are true twins, and it is *false-twin-free* if it does not contain any vertices x, y that are false twins. It is *twin-free* if it is both true-twin-free and false-twin-free.

Lemma 7.1.5. *Let $\mathcal{F}$ be any hereditary graph class and let $\mathcal{F}'$ be either the set of true-twin free members of $\mathcal{F}$, or the set of false-twin free members of $\mathcal{F}$. If $\mathcal{F}'$ admits an (s,k)-equality based labeling scheme, then $\mathcal{F}$ admits an $(s,k+1)$-equality based labeling scheme.*

Proof. We prove the lemma for $\mathcal{F}'$ being the true-twin free members of $\mathcal{F}$; the proof for the false-twin free members is similar. Let $G \in \mathcal{F}$. We construct a true-twin-free graph $G' \in \mathcal{F}'$ as follows. Let $T_1, \ldots, T_m \subseteq V(G)$ be the equivalence classes under the true-twin relation, so that for any i, any two vertices $x, y \in T_i$ are true twins. For each $i \in [m]$, let $t_i \in T_i$ be an arbitrary element, and let $T = \{t_1, \ldots, t_m\}$. We claim that $G[T]$ is true-twin free.

Suppose for contradiction that t_i, t_j are true twins in $G[T]$. Let $x \in T_i$, $y \in T_j$. Since t_i, t_j are adjacent in $G[T]$, they are adjacent in G. Then x is adjacent to t_j since x, t_i are twins. Since t_j, y are twins, x is adjacent to y. So $G[T_i, T_j]$ is a biclique. Now suppose that $z \in T_k$ for some $k \notin \{i, j\}$, and assume z is adjacent to x. Then z is adjacent to t_i since x, t_i are twins, and t_i is adjacent to t_k since z, t_k are twins. Since t_i, t_j are twins, it also holds that t_j is adjacent to t_k and to z. So y is adjacent to z. Then for any z it holds that x, z are adjacent if and only if y, z are adjacent. So x, y are true twins, for any $x \in T_i, y \in T_j$. But then $T_i \cup T_j$ is an equivalence class under the true-twin relation, which is a contradiction.

Therefore, $G[T]$ is true-twin free and a member of $\mathcal{F}$, so $G[T] \in \mathcal{F}'$. For any $x \in V(G)$, assign the label $(p(t_i) \mid q(t_i), i)$ where $(p(t_i) \mid q(t_i))$ is the label of t_i in the equality-based labeling scheme for $\mathcal{F}'$, and $i \in [m]$ is the unique index such that $x \in T_i$. The decoder for $\mathcal{F}$ performs the following on labels $(p(t_i) \mid q(t_i), i)$ and $(p(t_j) \mid q(t_j), j)$. If $i = j$ output 1; otherwise simulate the decoder for $\mathcal{F}'$ on labels $(p(t_i) \mid q(t_j))$ and $(p(t_j) \mid q(t_j))$. For vertices x, y in G, if x, y are true twins then $i = j$ and the decoder outputs 1. Otherwise, the adjacency between x and y is equivalent to the adjacency between t_i, t_j in $G[T]$, which is computed by the decoder for $\mathcal{F}'$, as desired. □

The true-twin free interval graphs with bounded chain number also have bounded clique number.

Lemma 7.1.6. *Let G be a true-twin free interval graph and let G contain a clique of c vertices. Then G has chain number at least $\lfloor \sqrt{c}/2 \rfloor$.*

Proof. Since G is interval, there is an interval realization $\ell, r : V(G) \to \mathbb{R}$ with $\ell(v) \leq r(v)$ for all $v \in V(G)$ so that $u, v \in V(G)$ are adjacent if and only if $[\ell(u), r(u)] \cap [\ell(v), r(v)] \neq$

$\emptyset$. We can assume without loss of generality that no two endpoints are the same. We abbreviate $i(v) = [\ell(v), r(v)]$. Let $X = \{x_1, \ldots, x_c\}$ be the c vertices that form a c-clique in G, arranged so that $\ell(x_1) < \ell(x_2) < \cdots < \ell(x_c)$. Consider the sequence $r = (r(x_1), \ldots, r(x_c))$ of right endpoints. By the Erdős-Szekeres theorem [ES35], any sequence of at least $R(k) = (k-1)^2+1$ distinct numbers contains either an increasing or a decreasing subsequence of length at least k. Setting $k = \lfloor \sqrt{c} \rfloor$, we have $R(k) \leq c$, so there is a clique over k vertices $y_1, \ldots, y_k$ with $\ell(y_1) < \cdots < \ell(y_k)$ and either $r(y_1) < \cdots < r(y_k)$ or $r(y_1) > \cdots > r(y_k)$. Graphically speaking, the intervals for $y_1, \ldots, y_k$ either form a staircase or a (step) pyramid. In either case, $\ell(y_k) < \min\{r(y_1), r(y_k)\}$, so $m = (\min\{r(y_1), r(y_k)\} + \ell(y_k))/2$ is contained in all $i(y_j)$. We will assume the staircase case, $r(y_1) < \cdots < r(y_k)$; see Figure 7.1. The pyramid case is similar.

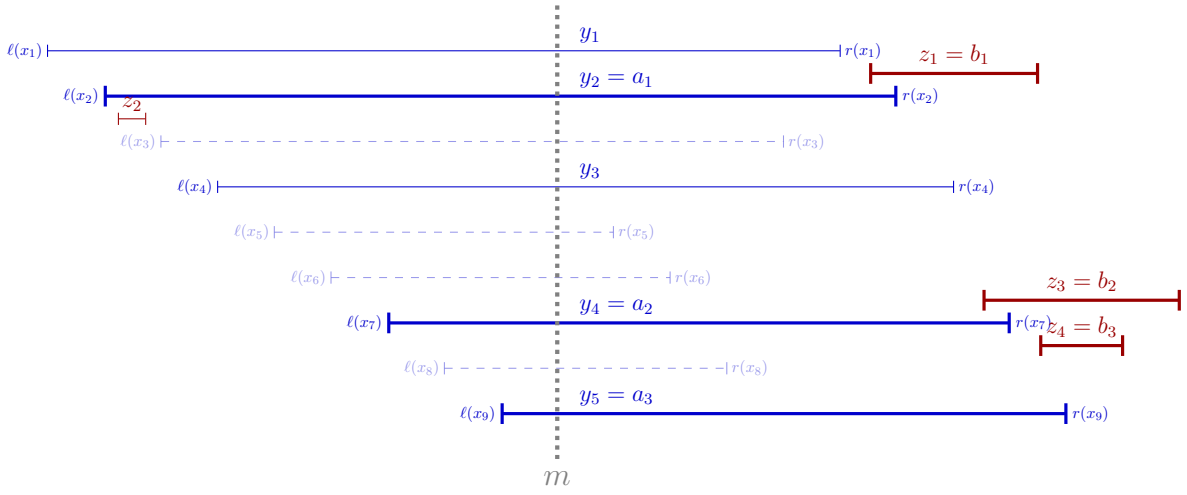


Figure 7.1: Example illustrating the notation from the proof of Lemma 7.1.6. The fat blue and red intervals, $a_1, \ldots, a_3$ resp. $b_1, \ldots, b_3$, form an induced subgraph with chain number 3.

Now, since G is true-twin free, for every $v, v' \in X$, there must be $u \in V(G)$ with $i(v) \cap i(u) = \emptyset$ and $i(v') \cap i(u) \neq \emptyset$ or vice versa, so $i(u)$ must lie entirely to the left or entirely to the right of $i(v)$ or $i(v')$. In particular, for pair y_j, y_{j+1} with $j \in [k-1]$, there must be $z_j \in V(G)$ adjacent to exactly one of these vertices. So the endpoints of $i(z_j)$ are on the same side of m ("left" or "right" of m) and the endpoint closer to m must be either between $\ell(y_j)$ and $\ell(y_{j+1})$ or between $r(y_j)$ and $r(y_{j+1})$.

Among the $k-1$ intervals $i(z_1), \ldots, i(z_{k-1})$, at least $h = \lceil (k-1)/2 \rceil$ are one the same side of m. Assume the majority is on the right; the other case is similar. So for

$1 \leq j_1 < \cdots < j_h \leq k-1$ intervals $i(z_{j_1}), \ldots, i(z_{j_h})$ are all to the right of m. Define $B = (b_1, \ldots, b_h) = (z_{j_1}, \ldots, z_{j_h})$ and $A = (a_1, \ldots, a_h) = (y_{j_1+1}, \ldots, y_{j_h+1})$. By definition, $\ell(b_1) < r(a_1) < \ell(b_2) < r(a_2) < \cdots < \ell(b_h) < r(a_h)$, so a_i is adjacent to b_j if and only if $j \leq i$. Hence $G[A, B]$ is isomorphic to $H_h^{\circ\circ}$. It is easy to check that $h = \lceil (\lfloor \sqrt{c} \rfloor - 1)/2 \rceil = \lfloor \sqrt{c}/2 \rfloor$. □

With these preparations, the proof of the main result of this section becomes easy.

Theorem 7.1.7. *Let $\mathcal{F}$ be a stable class of interval graphs. Then $\mathcal{F}$ admits a constant-size equality-based adjacency labeling scheme.*

Proof. Since $\mathcal{F}$ is stable, we have $\mathsf{ch}(\mathcal{F}) = k$ for some constant k. Let $\mathcal{F}'$ be the set of true-twin-free members of $\mathcal{F}$, and let $G' \in \mathcal{F}'$. Then $\mathsf{ch}(G') \leq k$, and hence the clique number of G' is at most $4(k+1)^2$ by (contraposition of) Lemma 7.1.6. So $\mathcal{F}'$ is a class of interval graphs with clique number bounded by $4(k+1)^2$, and hence by Lemma 7.1.3, it admits a constant-size equality-based labeling scheme (and a size $O(k^2)$ adjacency sketch). By Lemma 7.1.5, so does $\mathcal{F}$. □

Remark 7.1.8. We obtain an adjacency sketch of size $O(k^2)$ for interval graphs with chain number k. There is another, less direct proof of the above theorem that uses *twin-width* instead of Lemma 7.1.5, but does not recover this explicit bound on the sketch size. We prove in [HWZ22] that any stable class $\mathcal{F}$ with bounded twin-width admits a constant-size equality-based labeling scheme. Although interval graphs do not have bounded twin-width, some subclasses of interval graphs (e.g. unit interval graphs) are known to have bounded twin-width [BKTW20]. We observe in [HWZ22] that any stable class of interval graphs has bounded twin-width but we omit the proof from this thesis.

7.1.2 Permutation Graphs

We will denote by $\prec$ the standard partial order on $\mathbb{R}^2$, where $(x_1, x_2) \prec (y_1, y_2)$ if $x_1 \leq y_1$ and $x_2 \leq y_2$ and $(x_1, x_2) \neq (y_1, y_2)$.

The following alternative representation of permutation graphs is well-known (although one should note that adjacency is sometimes defined as between *incomparable* pairs, instead of comparable ones – this is equivalent since the complement of a permutation graph is again a permutation graph).

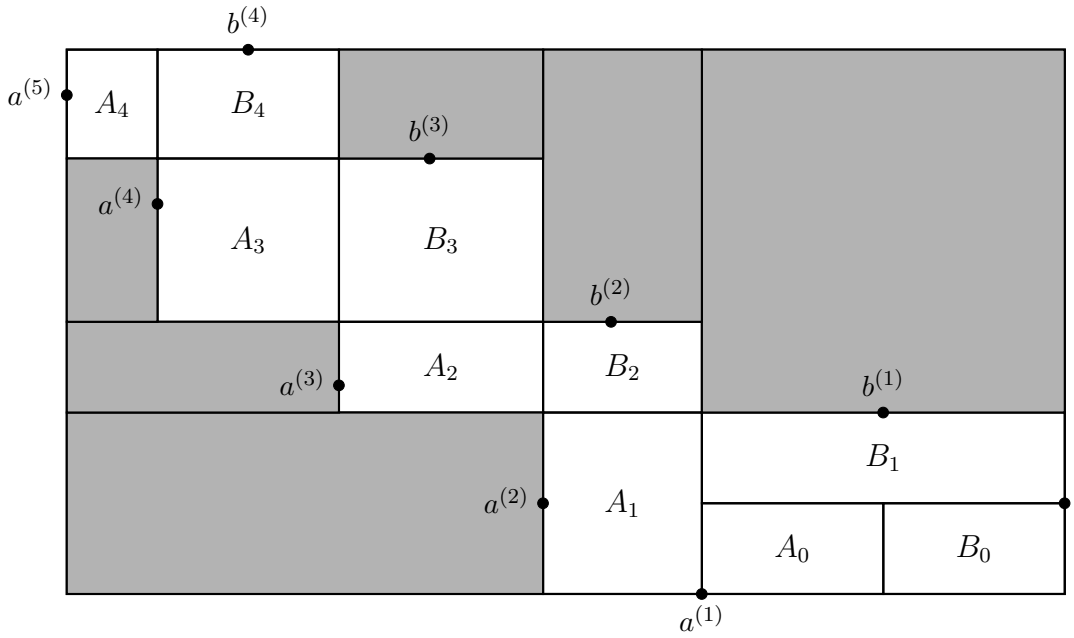


Figure 7.2: The permutation graph decomposition.

Proposition 7.1.9. *For any permutation graph G there is an injective mapping $\phi : V(G) \to \mathbb{R}^2$ such that $x, y \in V(G)$ are adjacent if and only if $\phi(x), \phi(y)$ are comparable in the partial order $\prec$. This mapping also satisfies the property that no two vertices x, y have $\phi(x)_i = \phi(y)_i$ for either $i \in [2]$.*

We will call this the $\mathbb{R}^2$-representation of G. From now on we will identify vertices of G with their $\mathbb{R}^2$-representation, so that a vertex x of G is a pair $(x_1, x_2) \in \mathbb{R}^2$. For a permutation graph G with fixed $\mathbb{R}^2$-representation, any $i \in [2]$, and any $t_1 < t_2$, we define

$$V_i(t_1, t_2) := \{x \in V(G) : t_1 < x_i < t_2\}.$$

We need the following lemma, which gives a condition that allows us to increment the chain number.

Lemma 7.1.10. *For a graph G and a set $A \subset V(G)$, suppose $u, v \in V(G) \setminus A$ are vertices such that u has no neighbors in A, while v is adjacent to u and to every vertex in A. Then $\mathsf{ch}(G[A \cup \{u, v\}]) > \mathsf{ch}(G[A])$.*

Proof. Suppose $\mathsf{ch}(G[A]) = k$ and let $\{a_1, \ldots, a_k, b_1, \ldots, b_k\}$ be the vertices such that a_i, b_j are adjacent if and only if $i \leq j$. Then set $a_{k+1} = u$ and $b_{k+1} = v$, and verify that

vertices $\{a_1, \ldots, a_{k+1}, b_1, \ldots, b_{k+1}\}$ satisfy Definition 4.2.22, so $\mathsf{ch}(G[A \cup \{u, v\}]) \geq k+1 > \mathsf{ch}(G[A])$. □

A bipartite graph $G = (X, Y, E)$ is called a *chain graph* if it belongs to $\mathcal{C}^{\circ\circ}$. Chain graphs are exactly the $2K_2$-free bipartite graphs, where $2K_2$ is the disjoint union of two edges.

Proposition 7.1.11. *For any $t \in \mathbb{R}$, any $\mathbb{R}^2$-representation of a permutation graph G, and for each $i \in [2]$, $G[V_i(-\infty, t), V_i(t, \infty)]$ is a chain graph.*

Proof. Let $V_1(-\infty, t) = \{a^{(1)}, \ldots, a^{(s)}\}$ and $V_1(t, \infty) = \{b^{(1)}, \ldots, b^{(t)}\}$ where the vertices $\{a^{(i)}\}$ and $\{b^{(i)}\}$ are sorted in increasing order in the second coordinate. Since $a_1^{(i)} < t < b_1^{(j)}$ for every i, j, it holds that $a^{(i)}, b^{(j)}$ are adjacent if and only if $a_2^{(i)} \leq b_2^{(j)}$. To prove the statement we will show that $G[V_i(-\infty, t), V_i(t, \infty)]$ is $2K_2$-free. Suppose, towards a contradiction, that $a^{(i_1)}, a^{(i_2)}, b^{(j_1)}, b^{(j_2)}$ induce a $2K_2$ in the graph, where $a^{(i_1)}$ is adjacent to $b^{(j_1)}$ and $a^{(i_2)}$ is adjacent to $b^{(j_2)}$. Assume, without loss of generality, that $a_2^{(i_1)} < a_2^{(i_2)}$. Since $a^{(i_2)}$ is adjacent to $b^{(j_2)}$, we have that $a_2^{(i_2)} \leq b_2^{(j_2)}$, which together with the previous inequality imply that $a_2^{(i_1)} < b_2^{(j_2)}$, and hence $a^{(i_1)}$ is adjacent to $b^{(j_2)}$, a contradiction. □

Any subclass of $\mathcal{C}^{\circ\circ}$ has a constant-size adjacency labeling scheme, because $\mathcal{C}^{\circ\circ}$ is a minimal factorial class. We give an explicit bound on the size of the labeling scheme, so that we can get an explicit bound on the size of the labels for permutation graphs.

Proposition 7.1.12. *Let $\mathcal{F} \subset \mathcal{C}^{\circ\circ}$ be a hereditary class of chain graphs of chain number at most k. Then $\mathcal{F}$ admits an adjacency labeling scheme of size $O(\log k)$.*

Proof. Let $G \in \mathcal{F}$, so that $G \sqsubset H_r^{\circ\circ}$ for some $r \in \mathbb{N}$. Then we can partition $V(G)$ into independent sets A and B, such that the following holds. There is a total order $\prec$ defined on $V(G) = A \cup B$ such that for $a \in A$ and $b \in B$, a, b are adjacent if and only if $a \prec b$. Then we may identify each $a \in A$ and each $b \in B$ with a number in $[n]$, such that the ordering $\prec$ is the natural ordering on $[n]$.

Let $A_1, \ldots, A_p \subset [n]$ be the set of (non-empty) maximal intervals such that each $A_i \subseteq A$, and let $B_1, \ldots, B_q \subseteq [n]$ be the set of (non-empty) maximal intervals such that each $B_i \subseteq B$. We claim that $p, q \leq k+1$. Suppose for contradiction that $p \geq k+2$. Since $A_1, \ldots, A_p$ are maximal, there exist $b_1, \ldots, b_{p-1} \in B$ such that $a_1 < b_1 < a_2 < b_2 < \cdots < b_{p-1} < a_p$, where we choose $a_i \in A_i$ arbitrarily. But then $\{a_1, \ldots, a_{p-1}\}, \{b_1, \ldots, b_{p-1}\}$ is a witness that $\mathsf{ch}(G) \geq p - 1 \geq k+1$, a contradiction. A similar proof shows that $q \leq k+1$.

We construct adjacency labels for G as follows. To each $x \in A$, assign 1 bit to indicate that $x \in A$, and append the unique number $i \in [k+1]$ such that x belongs to interval A_i. To each $y \in B$, assign 1 bit to indicate that $y \in B$, and append the unique number $j \in [k+1]$ such that $y \in B_j$. It holds that for $x \in A, y \in B$, x, y are adjacent if and only if $i \leq j$. Therefore, on seeing the labels for x and y, the decoder simply checks that $x \in A$ and $y \in B$ (or vice versa) and outputs 1 if $i \leq j$. □

Definition 7.1.13 (Permutation Graph Decomposition)**.** For a permutation graph G with a fixed $\mathbb{R}^2$-representation, where $G, \overline{G}$ are both connected, we define the following partition. Let $a^{(1)}$ be the vertex with minimum $a_2^{(1)}$ coordinate, and let b be the vertex with maximum b_2 coordinate. If $b_1 < a_1^{(1)}$, perform the following. Starting at $i = 1$, construct the following sequence:

(1) Let $b^{(i)}$ be the vertex with maximum $b_2^{(i)}$ coordinate among vertices with $b_1^{(i)} > a_1^{(i)}$.

(2) For $i > 1$, let $a^{(i)}$ be the vertex with minimum $a_1^{(i)}$ coordinate among vertices with $a_2^{(i)} < b_2^{(i-1)}$.

Let β be the smallest number such that $b^{(\beta+1)} = b^{(\beta)}$ and α the smallest number such that $a^{(\alpha+1)} = a^{(\alpha)}$. Define these sets:

$$\text{For } 2 \leq i \leq \alpha, \text{ define } A_i := \{a^{(i+1)}\} \cup \left(V_1(a_1^{(i+1)}, a_1^{(i)}) \cap V_2(b_2^{(i-1)}, b_2^{(i)})\right)$$

$$\text{For } 2 \leq i \leq \beta, \text{ define } B_i := \{b^{(i)}\} \cup \left(V_1(a_1^{(i)}, a_1^{(i-1)}) \cap V_2(b_2^{(i-1)}, b_2^{(i)})\right)$$

$$A_1 := \{a^{(2)}\} \cup \left(V_1(a_1^{(2)}, a_1^{(1)}) \cap V_2(a_2^{(1)}, b_2^{(1)})\right)$$

$$A_0 := \{a^{(1)}\} \cup \left(V_1(a_1^{(1)}, b_1^{(1)}) \cap V_2(a_2^{(1)}, a_2^{(2)})\right)$$

$$B_1 := \{b^{(1)}\} \cup \left(V_1(a_1^{(1)}, \infty) \cap V_2(a_2^{(2)}, b_2^{(1)})\right)$$

$$B_0 := \left(V_1(b_1^{(1)}, \infty) \cap V_2(a_2^{(1)}, a_2^{(2)})\right).$$

If $b_1 > a_1^{(1)}$, define the map $\phi : \mathbb{R}^2 \to \mathbb{R}^2$ as $\phi(x) = (-x_1, x_2)$ and apply ϕ to each vertex in the $\mathbb{R}^2$-representation of G; it is easily seen that the result is an $\mathbb{R}^2$-representation of $\overline{G}$. Then apply the above process to $\overline{G}$.

It is necessary to ensure that $b^{(1)}$ is well-defined, i.e. that the set of points x with $x_1 > a_1^{(1)}$ is non-empty, so that the maximum is taken over a non-empty set.

Proposition 7.1.14. *If G is connected then there exists $x \in V(G)$ such that $x_1 > a_1^{(1)}$.*

Proof. Suppose otherwise. Then every $x \in V(G)$ distinct from $a^{(1)}$ has $x_2 > a_2^{(1)}$ by definition, and $x_1 < a_1^{(1)}$. But then x is not adjacent to $a^{(1)}$. So $a^{(1)}$ has no neighbors, contradicting the assumption that G is connected. □

Proposition 7.1.15. *If G is connected and $b_1 < a_1^{(1)}$, then $b^{(1)} \neq b^{(2)}$.*

Proof. Suppose $b^{(2)} = b^{(1)}$. Then $b_2^{(1)} = b_2^{(2)}$ is maximum among all vertices x with $x_1 > a_1^{(2)}$, so $b_1 < a_1^{(2)}$. But all vertices x with $x_1 < a_1^{(2)}$ satisfy $x_2 > b_2^{(1)} = b_2^{(2)}$, so they cannot have an edge to $V_1(a_1^{(2)}, \infty)$. Both $V_1(a_1^{(2)}, \infty)$ and $V_1(-\infty, a_1^{(2)})$ are non-empty, so the graph is not connected. □

Proposition 7.1.16. *If G is connected, the sets $\{A_i\}_{i=0}^{\alpha}, \{B_i\}_{i=0}^{\beta}$ form a partition of $V(G)$.*

Proof. Let $C = \{x : x_1 \geq a_1^{(\alpha)}, x_2 \leq b_2^{(\beta)}\}$. There are no vertices x with $x_1 > a_1^{(\alpha)}$ and $x_2 > b_2^{(\beta)}$, since this would contradict the definition of $b^{(\beta)}$; likewise, there are no vertices x with $x_1 < a_1^{(\alpha)}$ and $x_2 < b_2^{(\beta)}$, since this would contradict the definition of $a^{(\alpha)}$. Now suppose $x_1 < a_1^{(\alpha)}, x_2 > b_2^{(\beta)}$. Then x has no edge to any vertex $y \in C$. Then the set of vertices with $x_1 < a_1^{(\alpha)}, x_2 > b_2^{(\beta)}$ must be empty, otherwise $V(G)$ is partitioned into $C, V(G) \setminus C$ where $V(G) \setminus C \neq \emptyset$ has no edges to C.

Then we may assume that every vertex x is in C; we will show that it belongs to some A_i or B_i. We may assume that x has distinct x_1, x_2 coordinates from all $a^{(i)}, b^{(i)}$, otherwise we would have $x = b^{(i)}$ or $x = a^{(i)}$, so x is an element of some A_i or B_i.

Let i be the smallest number such that $x_2 < b_2^{(i)}$. Suppose $i \geq 2$. By definition it must be that $x_1 > a_1^{(i+1)}$. If $x_1 > a_1^{(i-1)}$, then $x_2 < b_2^{(i-1)}$ by definition, which contradicts the choice of i. So it must be that $a_1^{(i+1)} < x_1 < a_1^{(i-1)}$ and $b_2^{(i-1)} < x_2 < b_2^{(i)}$. The set of points that satisfy this condition is contained in $A_i \cup B_i$. Now suppose $i = 1$. Again, it must be that $x_1 > a_1^{(2)}$ by definition, and also $a_2^{(1)} < x_2 < b_2^{(1)}$. The points satisfying these conditions are easily seen to be partitioned by A_0, B_0, A_1, B_1. □

For any subset $A \subset V(G)$ and any two vertices $u, v \in V(G) \setminus A$, we will say that u, v *cover* A if u has no edge into A and v is adjacent to u and every vertex in A. Then by Lemma 7.1.10, if u, v cover A, then $\mathsf{ch}(G) > \mathsf{ch}(G[A])$.

Proposition 7.1.17. *If G is connected and $b_1 < a_1^{(1)}$, then for each $D \in \{A_i\}_{i=0}^{\alpha} \cup \{B_i\}_{i=0}^{\beta}$, $\mathsf{ch}(G[D]) < \mathsf{ch}(G)$.*

Proof. Each $x \in B_0$ satisfies $x_1 > b_1^{(1)} > a_1^{(1)}$ and $a_2^{(1)} < x_2 < a_2^{(2)} < b_2^{(1)}$, so $b^{(1)}$ has no neighbors in B_0 while $a^{(1)}$ is adjacent to $b^{(1)}$ and all vertices in B_0, so $a^{(1)}, b^{(1)}$ cover B_0.

Each $x \in B_1$ satisfies $x_1 > a_1^{(1)} > b_1^{(2)} > a_1^{(2)}$ and $a_2^{(1)} < x_2 < b_2^{(1)} < b^{(2)}$, so $b^{(2)}$ has no neighbors in B_1 and $a^{(2)}$ is adjacent $b^{(2)}$ and all vertices in B_1, so $a^{(2)}, b^{(2)}$ cover B_1.

Each $x \in A_0$ satisfies $b_1^{(1)} > x_1 \geq a_1^{(1)} > a_1^{(2)}$ and $x_2 < a_2^{(2)} < b_2^{(1)}$, so $a^{(2)}$ has no neighbors in A_0 and $b^{(1)}$ is adjacent to $a^{(2)}$ and all vertices in A_0, so $a^{(2)}, b^{(1)}$ cover A_0.

Next we show that for any $1 \leq i \leq \alpha$, A_i is covered by $a^{(i)}, b^{(i)}$. By definition each $x \in A_i$ satisfies $x_1 < a_1^{(i)} < b_1^{(i)}$ and $a_2^{(i)} < b_2^{(i-1)} < x_2 < b_2^{(i)}$, so $a^{(i)}$ has no neighbors in A_i while $b^{(i)}$ is adjacent to $a^{(i)}$ and all vertices in A_i.

Finally, we show that for any $2 \leq i \leq \beta$, B_i is covered by $a^{(i)}, b^{(i-1)}$. By definition each $x \in B_i$ satisfies $a_1^{(i)} < x_1 < a_1^{(i-1)} < b_1^{(i-1)}$ and $a_2^{(i)} < b_2^{(i-1)} < x_2$, so $b^{(i-1)}$ has no neighbors in B_i while $a^{(i)}$ is adjacent to $b^{(i-1)}$ and all vertices in B_i. □

Lemma 7.1.18. *Let $G \in \mathcal{P}$ be any permutation graph. Then one of the following holds:*

(1) *G is disconnected;*

(2) *$\overline{G}$ is disconnected;*

(3) *There is a partition $V(G) = V_1 \cup \cdots \cup V_m$ such that:*

- *$\mathsf{ch}(G[V_i]) < \mathsf{ch}(G)$ for each $i \in [m]$, or $\mathsf{ch}(\overline{G[V_i]}) < \mathsf{ch}(\overline{G})$ for each $i \in [m]$;*
- *For each $i \in [m]$, there is a set $J(i) \subset \{V_t\}_{t\in[m]}$ of at most 4 parts such that for each $W \in J(i)$, $G[V_i, W]$ is a chain graph; and*
- *One of the following holds:*
  - *For all $i \in [m]$ and $W \in \{V_t\}_{t\in[m]} \setminus J(i)$, $G[V_i, W]$ is a co-biclique; or,*
  - *For all $i \in [m]$ and $W \in \{V_t\}_{t\in[m]} \setminus J(i)$, $G[V_i, W]$ is a biclique.*

Proof. Assume $G, \overline{G}$ are connected. Perform the decomposition of Definition 7.1.13. We will let $m = \alpha + \beta + 2$ and let $V_1, \ldots, V_m$ be the sets $\{A_i\}_{i=0}^{\alpha} \cup \{B_i\}_{i=0}^{\beta}$.

Case 1: $b_1 < a_1^{(1)}$. Then $V_1, \ldots, V_m$ is a partition due to Proposition 7.1.16, and $\mathsf{ch}(G[V_i]) < \mathsf{ch}(G), i \in [m]$ holds by Proposition 7.1.17. For $V_i = A_1$ we define the corresponding set $J(i) = \{A_0, B_0, B_1, B_2\}$. Since all sets V_i, V_j with $i \neq j$ are separated by a horizontal line or a vertical line, it holds by Proposition 7.1.11 that $G[V_i, V_j]$ is a chain graph. Now let $W \notin J(i)$. Observe that all $x \in W$ must satisfy $x_1 < a_1^{(2)}$ and $x_2 > b_2^{(1)}$, so x is not adjacent to any vertex in A_1. So $G[A_1, W]$ is a co-biclique.

Now for $V_i \in \{A_0, B_0, B_1\}$, we let $J(i) = \{A_0, B_0, A_1, B_1\} \setminus V_i$. Similar arguments as above hold in this case to show that $G[V_i, W]$ is a co-biclique for each $W \notin J(i)$.

For $V_i = A_j$ for some $j > 1$, we define $J(i) = \{B_j, B_{j+1}\}$. For any $W \notin J(i)$ with $W \neq A_j$, it holds either that all $x \in W$ satisfy $x_1 < a_1^{(i+1)}$ and $x_2 > b_2^{(j)}$, or that all $x \in W$ satisfy $x_1 \geq a_1^{(j)}$ and $x_2 \leq b_2^{(j-1)}$; in either case x is not adjacent to any vertex in A_j, so $G[A_j, W]$ is a co-biclique.

For $V_i = B_j$ for some $j > 1$, we define $J(i) = \{A_j, A_{j-1}\}$. Similar arguments to the previous case show that $G[B_j, W]$ is a co-biclique for each $W \notin J(i), W \neq B_j$. This concludes the proof for case 1.

Case 2: $b_1 > a_1^{(1)}$. In this case we transform the $\mathbb{R}^2$-representation of G using ϕ to obtain an $\mathbb{R}^2$-representation of $\overline{G}$ and apply the arguments above to obtain $V_1, \ldots, V_m$ such that $\mathsf{ch}(\overline{G}[V_i]) < \mathsf{ch}(\overline{G})$ for each $i \in [m]$, and each $V_j \in \{V_t\}_{t \in [m]} \setminus J(i)$ satisfies that $\overline{G}[V_i, V_j]$ is a co-biclique; then $G[V_i, V_j]$ is a biclique as desired. □

Theorem 7.1.19. *Let $\mathcal{F}$ be a stable subclass of permutation graphs. Then $\mathcal{F}$ admits a constant-size equality-based labeling scheme.*

Proof. Since $\mathcal{F}$ is stable, we have $\mathsf{ch}(\mathcal{F}) = k$ for some constant k.

We apply an argument similar to Lemma 7.2.2. For any $G \in \mathcal{F}$, we construct a decomposition tree where each node is associated with either an induced subgraph of G, or a bipartite induced subgraph of G, with the root node being G itself. For each node G', we decompose G' into children as follows,

1. If G' is a chain graph, the node is a leaf node.
2. If G' is disconnected, call the current node a D-node, and let the children $G_1, \ldots, G_t$ be the connected components of G'.
3. If $\overline{G}'$ is disconnected, call the current node a $\overline{D}$-node, and let $C_1, \ldots, C_t \subseteq V(G')$ be such that $\overline{G}'[C_i], i \in [t]$ are the connected components of $\overline{G}'$. Define the children to be $G_i = G[C_i], i \in [t]$.
4. Otherwise construct $V_1, \ldots, V_m$ as in Lemma 7.1.18 and let the children be $G[V_i]$ for each $i \in [m]$ and $G[V_i, V_j]$ for each i, j such that $i \in [m]$ and $V_j \in J(i)$. Call this node a P-node.

We will show that this decomposition tree has bounded depth. As in the decomposition for bipartite graphs, on any leaf-to-root path there cannot be two adjacent D-nodes or $\overline{D}$-nodes. As in the proof of Claim 7.2.25, if G'' is associated with a D-node and its parent G' is associated with a $\overline{D}$-node, and G''' is any child of G'', then $\mathsf{ch}(G') > \mathsf{ch}(G''')$. On the other hand, if G'' is associated with a $\overline{D}$-node and its parent is associated with a D-node, then $\mathsf{ch}(\overline{G'}) > \mathsf{ch}(\overline{G'''})$.

Now consider any P-node associated with G', with child G''. By Lemma 7.1.18, it holds that either G'' is a bipartite induced subgraph of G' that is a chain graph, or G'' has $\mathsf{ch}(G'') < \mathsf{ch}(G')$ or $\mathsf{ch}(\overline{G''}) < \mathsf{ch}(\overline{G'})$. It is easy to verify that $\mathsf{ch}(\overline{G}) \leq \mathsf{ch}(G) + 1$ for any graph G. Now, since every sequence G''', G'', G' of inner nodes along the leaf-to-root path in the decomposition tree must satisfy $\mathsf{ch}(G''') < \mathsf{ch}(G')$ or $\mathsf{ch}(\overline{G'''}) < \mathsf{ch}(\overline{G'})$ and $\mathsf{ch}(\overline{G}) \leq k + 1$, it must be that the depth of the decomposition tree is at most $2(2k + 1)$.

Now we construct an equality-based labeling scheme. For a vertex x, we construct a label at each node G' inductively as follows.

1. If G' is a leaf node, it is a chain graph with chain number at most k. We may assign a label of size $O(\log k)$ due to Proposition 7.1.12.

2. If G' is a D-node with children $G_1, \ldots, G_t$, append the pair $(D \,|\, i)$ where the equality code i is the index of the child G_i that contains x, and recurse on G_i.

3. If G' is a $\overline{D}$-node with children $G_1, \ldots, G_t$, append the pair $(\overline{D} \,|\, i)$ where the equality code i is the index of the child G_i that contains x, and recurse on G_i.

4. If G' is a P-node, let $V_1, \ldots, V_m$ be partition of $V(G')$ as in Lemma 7.1.18, and for each i let $J(i)$ be the (at most 4) indices such that $G'[V_i, V_j]$ is a chain graph when $j \in J(i)$. Append the tuple

$$(P, b, \ell_1(x), \ell_2(x), \ell_3(x), \ell_4(x) \mid i, j_1, j_2, j_3, j_4)$$

   where b indicates whether all $G'[V_i, V_j]$, $j \notin J(i)$ are bicliques or co-bicliques; the equality code i is the index such that $x \in V_i$, the equality codes $j_1, \ldots, j_4$ are the elements of $J(i)$, and $\ell_s(x)$ is the $O(\log k)$-bits adjacency label for x in the chain graph $G'[V_i, V_{j_s}]$. Then, recurse on the child $G'[V_i]$.

Given labels for x and y, which are sequences of the tuples above, the decoder iterates through the pairs and performs the following. On pairs $(D, i), (D, j)$ the decoder outputs

0 if $i \neq j$, otherwise it continues. On pairs $(\overline{D}, i), (\overline{D}, j)$, the decoder outputs 1 if $i \neq j$, otherwise it continues. On tuples

$$(P, b, \ell_1(x), \ell_2(x), \ell_3(x), \ell_4(x) \mid i, j_1, j_2, j_3, j_4)$$
$$(P, b, \ell_1(y), \ell_2(y), \ell_3(y), \ell_4(y) \mid i', j'_1, j'_2, j'_3, j'_4)\,,$$

the decoder continues to the next tuple if $i = i'$. Otherwise, the decoder outputs 1 if $i \notin \{j'_1, \ldots, j'_4\}$ and $i' \notin \{j_1, \ldots, j_4\}$ and b indicates that $G'[V_i, V_j]$ are bicliques for $j \notin J(i)$; it outputs 0 if b indicates otherwise. If $i = j'_s$ and $i' = j_t$ then the decoder outputs the adjacency of x, y using the labels $\ell_t(x), \ell_s(y)$. On any tuple that does not match any of the above patterns, the decoder outputs 0.

Since the decomposition tree has depth at most $2(2k+1)$, each label consists of $O(k)$ tuples. Each tuple contains at most $O(\log k)$ prefix bits (since adjacency labels for the chain graph with chain number at most k have size at most $O(\log k)$) and at most 5 equality codes. So this is an $(O(k \log k), O(k))$-equality-based labeling scheme.

The correctness of the labeling scheme follows from the fact that at any node G', if x, y belong to the same child of G', the decoder will continue to the next tuple. If G' is the lowest common ancestor of x, y in the decomposition tree, then x and y are adjacent in G if and only if they are adjacent in G'. If G' is a D- or $\overline{D}$-node then adjacency is determined by the equality of i, j in the tuples $(D \mid i), (D \mid j)$ or $(\overline{D} \mid i), (\overline{D} \mid j)$. If G' is a P-node and $i \notin J(i')$ (equivalently, $i' \notin J(i)$) then adjacency is determined by b. If $i \in J(i')$ (equivalently, $i' \in J(i)$) then $i = j'_s$ and $i' = j_t$ for some s, t, and the adjacency of x, y is equivalent to their adjacency in $G[V_i, V_{i'}] = G[V_{j'_s}, V_{j_t}]$, which is a chain graph, and it is determined by the labels $\ell_t(x), \ell_s(y)$. □

Remark 7.1.20. We get an explicit $O(k \log k)$ bound on the size of the adjacency sketch in terms of the chain number k, due to Proposition 4.2.12; this explicit bound would not arise from the alternate proof that goes through bounded expansion (proper subclasses of permutation graphs have bounded twin-width [BKTW20], and therefore stable subclasses of permutation graphs have structurally bounded expansion, so we could apply Lemma 6.3.5.

7.2 Monogenic Bipartite Graphs

We consider here classes of *colored* bipartite graphs, which are bipartite graphs $G = (X, Y, E)$ with a given bipartition of the vertices into X and Y. Recall from Section 4.1.1 the definitions of induced subgraphs for colored bipartite graphs. For a colored bipartite

graph H, a hereditary class $\mathcal{F}$ of colored bipartite graphs is H-free if it does not contain H.

In this section we prove the following theorem from Section 1.3.5.

Theorem 1.3.26. *Let H be a bipartite graph such that the class $\mathcal{H}$ of H-free bipartite graphs has at most factorial speed. Then any hereditary subclass $\mathcal{F}$ of $\mathcal{H}$ is adjacency sketchable if and only if it is stable.*

To accomplish this, we present a general type of decomposition scheme that can be applied in a number of cases. Our proof will proceed by a case analysis of the graphs H for which the family of H-free bipartite graphs is a factorial class, using the structural results of [All09, LZ17].

7.2.1 Decomposition Scheme for Bipartite Graphs

In this section we define a decomposition scheme for bipartite graphs.

Definition 7.2.1 ($(\mathcal{Q},k)$-decomposition tree)**.** Let $G = (X, Y, E)$ be a bipartite graph, $k \geq 2$, and let $\mathcal{Q}$ be a hereditary class of bipartite graphs. A graph G admits a $(\mathcal{Q},k)$-*decomposition tree of depth d* if there is a tree of depth d of the following form, with G as the root. Each node of the tree is a bipartite graph $G' = G[X', Y']$ for some $X' \subseteq X, Y' \subseteq Y$, labelled with either $L, D, \overline{D}$, or P as follows

(1) L (*leaf node*): The graph G' belongs to $\mathcal{Q}$.

(2) D (*D-node*): The graph G' is disconnected. There are sets $X'_1, \ldots, X'_t \subseteq X'$ and $Y'_1, \ldots, Y'_t \subseteq Y'$ such that $G[X'_1, Y'_1], \ldots, G[X'_t, Y'_t]$ are the connected components of G'. The children of this decomposition tree node are $G[X'_1, Y'_1], \ldots, G[X'_t, Y'_t]$.

(3) $\overline{D}$ ($\overline{D}$*-node*): The graph $\overline{\overline{G'}}$ is disconnected. There are sets $X'_1, \ldots, X'_t \subseteq X'$ and $Y'_1, \ldots, Y'_t \subseteq Y'$ such that $\overline{\overline{G[X'_1, Y'_1]}}, \ldots, \overline{\overline{G[X'_t, Y'_t]}}$ are the connected components of $\overline{\overline{G'}}$. The children of this decomposition tree node are $G[X'_1, Y'_1], \ldots, G[X'_t, Y'_t]$.

(4) P (*P-node*): The vertex set of G' is partitioned into at most $2k$ *non-empty* sets $X'_1, X'_2, \ldots, X'_p \subseteq X'$ and $Y'_1, Y'_2, \ldots, Y'_q \subseteq Y'$, where $p \leq k$, $q \leq k$. The children of this decomposition tree node are $G[X'_i, Y'_j]$, for all $i \in [p]$, $j \in [q]$. We say that the P-node G' is *specified* by the partitions $X'_1, X'_2, \ldots, X'_p$ and $Y'_1, Y'_2, \ldots, Y'_q$.

Lemma 7.2.2. *Let $k \geq 2$ and $d \geq 1$ be natural constants, and let $\mathcal{Q}$ be a class of bipartite graphs that admits a constant-size equality-based adjacency labeling scheme. Let $\mathcal{F}$ be a class of bipartite graphs such that each $G \in \mathcal{F}$ admits a $(\mathcal{Q}, k)$-decomposition tree of depth at most d. Then $\mathcal{F}$ admits a constant-size equality-based adjacency labeling scheme.*

Proof. Let $G = (X, Y, E) \in \mathcal{F}$. We fix a $(\mathcal{Q}, k)$-decomposition tree of depth at most d for G. For each node v in the decomposition tree we write G_v for the induced subgraph of G associated with node v. Each leaf node v has $G_v \in \mathcal{Q}$. For some constants s and r, we fix an (s, r)-equality-based adjacency labeling scheme for $\mathcal{Q}$, and for each leaf node v, we denote by ℓ'_v the function that assigns labels to the vertices of G_v under this scheme.

For each vertex x we will construct a label $\ell(x)$ that consists of a constant number of tuples , where each tuple contains one prefix of at most two bits, and at most two equality codes. First, we add to $\ell(x)$ a tuple $(\alpha(x) \mid -)$, where $\alpha(x) = 0$ if $x \in X$, and $\alpha(x) = 1$ if $x \in Y$. Then we append to $\ell(x)$ tuples defined inductively. Starting at the root of the decomposition tree, for each node v of the tree where G_v contains x, we add tuples $\ell_v(x)$ defined as follows. Write $X' \subseteq X, Y' \subseteq Y$ for the vertices of G_v.

- If v is a leaf node, then $G_v \in \mathcal{Q}$, and we define $\ell_v(x) = (L \mid -), \ell'_v(x)$.
- If v is a D-node then G_v is disconnected, with sets $X'_1, \ldots, X'_t \subseteq X', Y'_1, \ldots, Y'_t \subseteq Y$ such that the children $v_1, \ldots, v_t$ are the connected components $G_v[X'_1, Y'_1]$, $\ldots$, $G_v[X'_t, Y'_t]$ of G_v. We define $\ell_v(x) = (D \mid j), \ell_{v_j}(x)$, where $j \in [t]$ is the unique index such that x belongs to the connected component $G_v[X'_j, Y'_j]$, and $\ell_{v_j}(x)$ is the inductively defined label for the child node v_j.
- If v is a $\overline{D}$-node then $\overline{\overline{G_v}}$ is disconnected, with sets $X'_1, \ldots, X'_t \subseteq X', Y'_1, \ldots, Y'_t \subseteq Y$ such that $\overline{\overline{G_v[X'_1, Y'_1]}}, \ldots, \overline{\overline{G_v[X'_t, Y'_t]}}$ are the connected components of $\overline{\overline{G_v}}$, and the children $v_1, \ldots, v_t$ of v are the graphs $G_v[X'_1, Y'_1], \ldots, G_v[X'_t, Y'_t]$. We define $\ell_v(x) = (\overline{D} \mid j), \ell_{v_j}(x)$, where $j \in [t]$ is the unique index such that x belongs to $G_v[X'_j, Y'_j]$, and $\ell_{v_j}(x)$ is the inductively defined label for the child node v_j.
- If v is a P-node then let $X'_1, \ldots, X'_p \subseteq X', Y'_1, \ldots, Y'_q \subseteq Y'$ be the partitions of X', Y' with $p, q \leq k$. For each $(i, j) \in [p] \times [q]$, let $v_{i,j}$ be the child node of v corresponding to the subgraph $G_v[X'_i, Y'_j]$. If $x \in X$, then there is a unique $i \in [p]$ such that $x \in X'_i$, and we define $\ell_v(x) = (P \,|\, i, q), \ell_{v_{i,1}}(x), \ldots, \ell_{v_{i,q}}(x)$, where $\ell_{v_{i,j}}(x)$ is the label assigned to x at node $v_{i,j}$. If $x \in Y$, then we define $\ell_v(x) = (P \,|\, i, p), \ell_{v_{1,i}}(x), \ldots, \ell_{v_{p,i}}(x)$, where $i \in [q]$ is the unique index such that $x \in Y'_i$.

First, we will estimate the size of the label $\ell(x)$ produced by the above procedure. For every leaf node v, the label $\ell_v(x)$ of x is a tuple consisting of an s-bit prefix and r equality codes. Let $f(i)$ be the maximum number of tuples added to $\ell(x)$ by a node v at level i of the decomposition tree, where the root node belongs to level 0. Then, by construction, $f(i) \leq 1+k \cdot f(i+1)$ and $f(d-1) = 1$, which implies that the total number of tuples in $\ell(x)$ does not exceed $f(0) \leq k^d$. Since every tuple contains a prefix with at most $s' = \max\{2, s\}$ bits, and at most $r' = \max\{2, r\}$ equality codes, we have that the label $\ell(x)$ contains a prefix with at most $s'k^d$ bits, and at most $r'k^d$ equality codes.

We will now show how to use the labels to define an equality-based adjacency decoder. Let x and y be two arbitrary vertices of G. The decoder first checks the first tuples $(\alpha(x) \mid -)$ and $(\alpha(y) \mid -)$ of the labels $\ell(x)$ and $\ell(y)$ respectively, to ensure that x, y are in different parts of G and outputs 0 if they are not. We may now assume $x \in X, y \in Y$. The remainder of the labels are of the form $\ell_v(x)$ and $\ell_v(y)$, where v is the root of the decomposition tree.

- If the labels $\ell_v(x), \ell_v(y)$ are of the form $(L \mid -), \ell'_v(x)$ and $(L \mid -), \ell'_v(y)$, then the decoder simulates the decoder for the labeling scheme for $\mathcal{Q}$, on inputs $\ell'_v(x), \ell'_v(y)$, and outputs the correct adjacency value.
- If the labels $\ell_v(x), \ell_v(y)$ are of the form $(D \mid i), \ell_{v_i}(x)$ and $(D \mid j), \ell_{v_j}(y)$, the decoder outputs 0 when $i \neq j$ (i.e. x, y are in different connected components of G_v), and otherwise it recurses on $\ell_{v_i}(x), \ell_{v_i}(y)$.
- If the labels $\ell_v(x), \ell_v(y)$ are of the form $(\overline{D} \mid i), \ell_{v_i}(x)$ and $(\overline{D} \mid j), \ell_{v_j}(y)$, the decoder outputs 1 when $i \neq j$ (i.e. x, y are in different connected components of $\overline{\overline{G_v}}$ and therefore they are adjacent in G_v), and otherwise it recurses on $\ell_{v_i}(x), \ell_{v_i}(y)$.
- If the labels $\ell_v(x)$, $\ell_v(y)$ are of the form $(P \mid i, q), \ell_{v_{i,1}}(x), \ldots, \ell_{v_{i,q}}(x)$ and $(P \mid j, p)$, $\ell_{v_{1,j}}(y), \ldots, \ell_{v_{p,j}}(y)$ the decoder recurses on $\ell_{v_{i,j}}(x)$ and $\ell_{v_{i,j}}(y)$.

It is routine to verify that the decoder will output the correct adjacency value for x, y. □

Remark 7.2.3 ($(\mathcal{Q}, k)$-tree for general graphs)**.** A similar decomposition scheme can be used for non-bipartite graph classes; we do this for permutation graphs in Section 7.1.2.

7.2.2 Monogenic Bipartite Graph Classes

Let $\mathcal{H}$ be a *finite* set of bipartite graphs. It is known [All09] that if the class of $\mathcal{H}$-free bipartite graphs is at most factorial, then $\mathcal{H}$ contains a forest and a graph whose bipartite

complement is a forest. The converse was conjectured in [LZ17], where it was verified for monogenic classes of bipartite graphs. More specifically, it was shown that, for a colored bipartite graph H, the class of H-free bipartite graphs is at most factorial if and only if both H and its bipartite complement is a forest. It is not hard to show that a colored bipartite graph H is a forest and its bipartite complement is a forest if and only if H is an induced subgraph of $S_{1,2,3}, P_7$, or one of the graphs $F^*_{p,q}$, $p, q \in \mathbb{N}$ defined below.

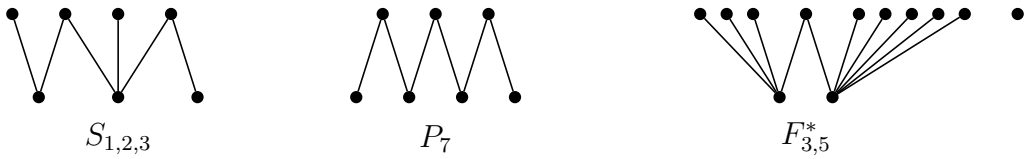


Figure 7.3: The bipartite graphs from Definition 7.2.4

Definition 7.2.4 ($S_{1,2,3}$, P_7, $F^*_{p,q}$)**.** See Figure 7.3 for an illustration.

(1) $S_{1,2,3}$ is the (colored) bipartite graph obtained from a star with three leaves by subdividing one of its edges once and subdividing another edge twice.

(2) P_7 is the (colored) path on 7 vertices.

(3) $F^*_{p,q}$ is the colored bipartite graph with vertex color classes $\{a, b\}$ and $\{a_1, \dots, a_p, c, b_1, \dots, b_q, d\}$. The edges are $\{(a, a_i) \mid i \in [p]\}$, $\{(b, b_j) \mid j \in [q]\}$, and $(a, c), (b, c)$.

Combining results due to Allen [All09] (for the $S_{1,2,3}$ and $F^*_{p,q}$ cases) and a result of Lozin & Zamaraev [LZ17] (for the P_7 case), we formally state

Theorem 7.2.5 ([All09, LZ17])**.** *Let H be a colored bipartite graph, and let $\mathcal{F}$ be the class of H-free bipartite graphs. If $\mathcal{F}$ has at most factorial speed, then $\mathcal{F}$ is a subclass of either the $S_{1,2,3}$-free bipartite graphs, the P_7-free bipartite graphs, or the $F^*_{p,q}$-free bipartite graphs, for some $p, q \in \mathbb{N}$.*

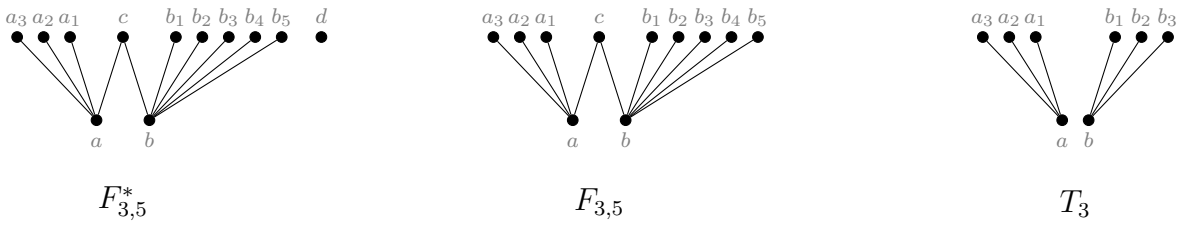
By the above result, in order to prove Theorem 1.3.26, it is enough to consider the *maximal* monogenic factorial classes of bipartite graphs defined by $S_{1,2,3}$, P_7, $F^*_{p,q}$. The first of these results follows from the results on structurally bounded expansion from the previous chapter:

Lemma 7.2.6. *Let $\mathcal{F}$ be any stable class of $S_{1,2,3}$-free bipartite graphs. Then $\mathcal{F}$ admits a constant-size equality-based adjacency labeling scheme, and therefore it is adjacency sketchable.*

Proof. It is known that the class of $S_{1,2,3}$-free bipartite graphs has bounded clique-width [LV08] and therefore bounded twin-width [BKTW20]. A stable class has bounded twin-width if and only if it is a first-order transduction of a class of bounded sparse twin-width [GPT21]. Every class of bounded sparse twin-width has bounded expansion [BGK+21]. Therefore $S_{1,2,3}$-free bipartite graphs are first-order transductions of a class with bounded expansion, i.e. they have structurally bounded expansion. That they admit a constant-size equality-based adjacency labeling follows from Lemma 6.3.5. □

The remaining two cases, $F^*_{p,q}$- and P_7-free bipartite graphs, are treated below.

$F^*_{p,q}$-Free Bipartite Graphs



$F^*_{3,5}$ $F_{3,5}$ T_3

Figure 7.4: The bipartite graphs considered in Section 7.2.2.

In this section, we prove Theorem 1.3.26 for classes of $F^*_{p,q}$-free bipartite graphs by developing a constant-size equality-based adjacency labeling scheme for stable classes of $F^*_{p,q}$-free bipartite graphs via a sequence of labeling schemes for special subclasses each generalizing the previous one.

We denote by $F_{p,q}$ the bipartite graph with parts $\{a, b\}$ and $\{c, a_1, \ldots, a_p, b_1, \ldots, b_q\}$, and with edges $(a, c), (b, c), \{(a, a_i) \mid i \in [p]\}, \{(b, b_i) \mid j \in [q]\}$. We also denote by T_p the bipartite graph on vertex sets $\{a, b\}, \{a_1, \ldots, a_p, b_1, \ldots, b_p\}$, where (a, a_i) and (b, b_i) are edges for each $i \in [p]$. So T_p is the disjoint union of two stars with $p + 1$ vertices.

Definition 7.2.7. For $q, s \in \mathbb{N}$ we denote by $Z_{q,s}$ the bipartite graph (X, Y, E) with $|X| = q, |Y| = qs$, where $X = \{x_1, \ldots, x_q\}$, Y is partitioned into q sets $Y = Y_1 \cup \ldots \cup Y_q$ each of size s, and for every $i \in [q]$:

(1) x_i is adjacent to all vertices in Y_j for all $1 \leq j \leq i$, and

(2) x_i is adjacent to no vertices in Y_j for all $i < j \leq q$.

Note that $Z_{q,s}$ is obtained from $H_q^{\circ\circ}$ by duplicating every vertex in one of the parts $s-1$ times. In particular, $H_q^{\circ\circ}$ is and induced subgraph of $Z_{q,s}$

We start with structural results and an equality-based labeling scheme for *one-sided* T_p-free bipartite graphs. A colored bipartite graph $G = (X, Y, E)$ is one-sided T_p-free if it does not contain T_p as an induced subgraph such that the centers of both stars belong to X. Note that any T_p-free bipartite graph is also a one-sided T_p-free graph.

Proposition 7.2.8. *Let $G = (X, Y, E)$ be any one-sided T_p-free bipartite graph and let $u, v \in X$ satisfy $\deg(u) \leq \deg(v)$. Then $|N(u) \cap N(v)| > |N(u)| - p$.*

Proof. For contradiction, assume $|N(u) \cap N(v)| \leq |N(u)| - p$ so that $|N(u) \setminus N(v)| \geq p$. Then since $\deg(v) \geq \deg(u)$ it follows that $|N(v) \setminus N(u)| \geq p$. But then T_p is induced by $\{u, v\}$ and $(N(u) \setminus N(v)) \cup (N(v) \setminus N(u))$. □

Proposition 7.2.9. *Suppose $S_1, \ldots, S_t \subseteq [n]$ each have $|S_i| \geq n - p$ where $n > pt$. Then $\left|\bigcap_{j=1}^{t} S_j\right| \geq n - pt$.*

Proof. Let R be the set of all $i \in [n]$ such that for some $S_j, i \notin S_j$. Then

$$|R| \leq \sum_{j=1}^{t} (n - |S_j|) \leq \sum_{j=1}^{t} p = pt\,,$$

so $\left|\bigcap_{j=1}^{t} S_j\right| \geq n - |R| \geq n - pt$. □

Lemma 7.2.10. *Fix any constants k, q, p such that $k \geq qp + 1$ and let $G = (X, Y, E)$ be any one-sided T_p-free bipartite graph. Then there exists $m \geq 0$ and partitions $X = A_0 \cup A_1 \cup \ldots \cup A_m$ and $Y = B_1 \cup \ldots \cup B_m \cup B_{m+1}$, where $A_i \neq \emptyset$, $B_i \neq \emptyset$ for every $i \in [m]$, such that the following hold*

(1) *$|B_i| \geq k$, for all $i \in [m]$.*

(2) *For every $j \in \{0, 1, \ldots, m\}$, every $x \in A_j$ has less than k neighbours in $\bigcup_{i \geq j+1} B_i$.*

(3) *For every $i, j, 1 \leq i \leq j \leq m$, every $x \in A_j$ has more than $|B_i| - p$ neighbours in B_i.*

(4) *If $m \geq q$, then $Z_{q,k-qp}$ is an induced subgraph of G.*

Proof. Let A_0 be the set of vertices in X that have less than k neighbours. If $A_0 = X$, then $m = 0$, A_0, and $B_1 = Y$ satisfy the conditions of the lemma. Otherwise, we construct the remaining parts of partitions using the following procedure. Initialize $X' = X \setminus A_0, Y' = Y$, and $i = 1$.

1. Let a_i be a vertex in X' with the least number of neighbours in Y'.
2. Let B_i be the set of all neighbors of a_i in $G[X', Y']$.
3. Let A_i be the set of vertices in X' with degree less than k in $G[X', Y' \setminus B_i]$. Note that A_i contains a_i.
4. $X' \leftarrow X' \setminus A_i$, $Y' \leftarrow Y' \setminus B_i$.
5. If $X' = \emptyset$, then $B_{i+1} = Y'$, let $m = i$, and terminate the procedure; Otherwise increment i and return to step 1.

Conditions (1) and (2) follow by definition. Next we will prove condition (3) by showing that for every $1 \leq i \leq j \leq m$, every $x \in A_j$ has more than $|B_i| - p$ neighbours in B_i. Suppose, towards a contradiction, that $|N(x) \cap B_i| \leq |B_i| - p$. Consider X', Y' as in round i of the construction procedure, so B_i is the neighbourhood of a_i in $G[X', Y']$. Then x has degree at least that of a_i in $G[X', Y']$, and hence the conclusion holds by Proposition 7.2.8.

Finally, to prove condition (4) we will show that for any $q \leq m$ there exist sets $B'_1 \subseteq B_1, \ldots, B'_q \subseteq B_q$ so that the vertices $\{a_1, \ldots, a_q\}$ and the sets $B'_1, \ldots, B'_q$ induce $Z_{q,k-pq}$. First, observe that by construction for every $1 \leq i < j \leq m$, a_i has no neighbours in B_j. Now, let $i \in [m]$, then by condition (3), for all $i \leq j \leq m$ it holds that $|N(a_j) \cap B_i| > |B_i| - p$. Since $|B_i| \geq k > pq$, it holds by Proposition 7.2.9 that

$$\left| B_i \cap \bigcap_{j=i}^{q} N(a_j) \right| \geq |B_i| - pq \geq k - pq \,.$$

We define $B'_i = B_i \cap \bigcap_{j=i}^{q} N(a_j)$. Then for each $i \in [m]$ it holds that a_i is adjacent to all vertices in B'_j for all $1 \leq j \leq i$, but a_i is adjacent to no vertices in B'_j for $i < j \leq m$. Hence the vertices $\{a_1, \ldots, a_q\}$ and the sets $B'_1, \ldots, B'_q$ induce $Z_{q,k-pq}$, which proves condition (4) and concludes the proof of the lemma. □

Lemma 7.2.11. *Let $p \in \mathbb{N}$ and let $\mathcal{T}$ be a stable class of one-sided T_p-free bipartite graphs. Then $\mathcal{T}$ admits a constant-size equality-based adjacency labeling scheme.*

Proof. Since $\mathcal{T}$ is stable, it does not contain $\mathcal{C}^{\circ\circ}$ as a subclass. Let q be the minimum number such that $H_q^{\circ\circ} \notin \mathcal{T}$, and let $G = (X, Y, E)$ be an arbitrary graph from $\mathcal{T}$.

Let $k = qp + 1$ and let $X = A_0 \cup A_1 \cup \ldots \cup A_m$ and $Y = B_1 \cup \ldots \cup B_m \cup B_{m+1}$ be partitions satisfying the conditions of Lemma 7.2.10. Since G does not contain $H_q^{\circ\circ}$ as an induced subgraph, it holds that $m < q$.

We construct the labels for the vertices of G as follows. For a vertex $x \in X$ we define $\ell(x)$ as a label consisting of several tuples. The first tuple is $(0, i \mid -)$, where $i \in \{0, 1, \ldots, m\}$ is the unique index such that $x \in A_i$. This tuple follows by i tuples $(- \mid y_1^j, y_2^j, \ldots, y_{p_j}^j)$, $j \in [i]$, where $p_j < p$ and $\{y_1^j, y_2^j, \ldots, y_{p_j}^j\}$ are the non-neighbours of x in B_j. The last tuple of $\ell(x)$ is $(- \mid y_1^{i+1}, y_2^{i+1}, \ldots, y_{k'}^{i+1})$, where $k' < k$ and $y_1^{i+1}, y_2^{i+1}, \ldots, y_{k'}^{i+1}$ are the neighbours of x in $\bigcup_{i \geq j+1} B_i$. For a vertex $y \in Y$ we define $\ell(y) = (1, i \mid y)$, where $i \in [m+1]$ is the unique index such that $y \in B_i$.

Note that, in every label, the total length of prefixes is at most $1 + \lceil \log m \rceil \leq 1 + \lceil \log q \rceil$, and the total number of equality codes depends only on p, q, and k, which are constants. Therefore it remains to show that the labels can be used to define an equality-based adjacency decoder.

Given two vertices x, y in G the decoder operates as follows. First, it checks the first prefixes in the first tuples of $\ell(x)$ and $\ell(y)$. If they are the same, then x, y belong to the same part in G and the decoder outputs 0. Hence, we can assume that they are different. Without loss of generality, let $\ell(x) = (0, i \mid -)$ and $\ell(y) = (1, j \mid y)$, so $x \in A_i \subseteq X$ and $y \in B_j \subseteq Y$.

If $j \leq i$, then the decoder compares y with the equality codes $y_1^j, y_2^j, \ldots, y_{p_j}^j$ of the $(j+1)$-th tuple of $\ell(x)$. If y is equal to at least one of them, then y is among the non-neighbours of x in B_j and the decoder outputs 0; otherwise, x and y are adjacent and the decoder outputs 1. If $j > i$, then the decoder compares y with the equality codes $y_1^{i+1}, y_2^{i+1}, \ldots, y_{k'}^{i+1}$ of the last tuples of $\ell(x)$, and if y is equal to at least one of them, then y is among the neighbours of x in $\bigcup_{i \geq j+1} B_i$ and the decoder outputs 1; otherwise, x and y are not adjacent and the decoder outputs 0. □

Next, we develop an equality-based labeling scheme for stable classes of one-sided $F_{p,p}$-free bipartite graphs. A colored bipartite graph $G = (X, Y, E)$ is one-sided $F_{p,p}$-free if it does not contain $F_{p,p}$ as an induced subgraph such that the part of $F_{p,p}$ of size 2 is a subset of X.

Proposition 7.2.12. *Let $G = (X, Y, E)$ be any one-sided $F_{p,p}$-free bipartite graph and let $u, v \in X$ satisfy $\deg(u) \leq \deg(v)$. Then either $N(u) \cap N(v) = \emptyset$ or $|N(u) \cap N(v)| > |N(u)| - p$.*

Proof. Suppose that $N(u) \cap N(v) \neq \emptyset$, and for contradiction assume that $|N(u) \setminus N(v)| \geq p$. Since $\deg(u) \leq \deg(u)$, this means $|N(v) \setminus N(u)| \geq |N(u) \setminus N(v)| \geq p$. Let $w \in N(u) \cap N(v)$. Then $\{u, v\}$ with $\{w\} \cup (N(v) \setminus N(u)) \cup (N(u) \setminus N(v))$ induces a graph containing $F_{p,p}$, a contradiction. □

Proposition 7.2.13. *Let $G = (X, Y, E)$ be any one-sided $F_{p,p}$-free bipartite graph and let $x, y, z \in X$ satisfy $\deg(x) \geq \deg(y) \geq \deg(z) \geq 2p$. Suppose that $N(y) \cap N(z) \neq \emptyset$. Then*

$$N(x) \cap N(y) = \emptyset \iff N(x) \cap N(z) = \emptyset\,.$$

Proof. Since $N(y) \cap N(z) \neq \emptyset$, it holds that $|N(y) \cap N(z)| > |N(z)| - p \geq p$ by Proposition 7.2.12.

Suppose that $N(x) \cap N(y) \neq \emptyset$. For contradiction, assume that $N(x) \cap N(y) \cap N(z) = \emptyset$. Then $|N(y) \setminus N(x)| \geq |N(y) \cap N(z)| > |N(z)| - p \geq p$, which contradicts $|N(y) \cap N(x)| > |N(y)| - p$.

Now suppose that $N(x) \cap N(y) = \emptyset$. For contradiction, assume that $N(x) \cap N(z) \neq \emptyset$. Then $|N(x) \cap N(z)| \leq |N(z) \setminus N(y)| < p \leq |N(z)| - p < |N(x) \cap N(z)|$, a contradiction. □

We will say that a bipartite graph $G = (X, Y, E)$ is *left-disconnected* if there are two vertices $x, y \in X$ that are in different connected components of G. It is *left-connected* otherwise.

Proposition 7.2.14. *Let $G = (X, Y, E)$ be any one-sided $F_{p,p}$-free bipartite graph where every vertex in X has degree at least $2p$. Let $x \in X$ have maximum degree of all vertices in X. If G is left-connected, then for any $y \in X$ it holds that $|N(y) \cap N(x)| > |N(y)| - p$.*

Proof. Let $y \in X$. Since G is left-connected, there is a path from y to x. Let $y_0, y_1, \ldots, y_t$ be the path vertices in X, where $y = y_0, x = y_t$, and $N(y_{i-1}) \cap N(y_i) \neq \emptyset$ for each $i \in [t]$. By Propositions 7.2.12 and 7.2.13, it holds that if $N(y_i) \cap N(x) \neq \emptyset$ then $|N(y_i) \cap N(x)| > |N(y_i)| - p$ and $|N(y_{i-1}) \cap N(x)| > |N(y_{i-1})| - p$. Therefore the conclusion holds, because $N(y_{t-1}) \cap N(x) = N(y_{t-1}) \cap N(y_t) \neq \emptyset$. □

Lemma 7.2.15. *Fix any constants $p, q \geq 1$, let $k = (q+1)p$, and let $G = (X, Y, E)$ be any connected one-sided $F_{p,p}$-free bipartite graph. Then there exists a partition $X = X_0 \cup X_1 \cup X_2$ (where some of the sets can be empty) such that the following hold:*

(1) *X_0 is the set of vertices in X that have degree less than k.*

(2) *The induced subgraph $G[X_1, Y]$ is one-sided T_p-free.*

(3) *The induced subgraph $G[X_2, Y]$ is left-disconnected.*

(4) *For any r, s such that $r < q$ and $p < s \leq k$, if $X_1 \neq \emptyset$ and $Z_{r,s} \sqsubset G[X_2, Y]$, then $Z_{r+1,s-p} \sqsubset G$.*

Proof. Let X_0 be the set of vertices in X that have degree less then k, and let $X' = X \setminus X_0$. If $G[X', Y]$ is left-disconnected, then we define $X_1 = \emptyset$ and $X_2 = X'$.

Assume now that $G[X', Y]$ is left-connected. By Proposition 7.2.14, the highest-degree vertex $x \in X'$ satisfies $|N(x) \cap N(y)| > |N(y)| - p$ for every $y \in X'$. Define X_1 as follows: add the highest-degree vertex x to X_1, and repeat until $G[X' \setminus X_1, Y]$ is left-disconnected. Then set $X_2 = X' \setminus X_1$. Condition 3 holds by definition, so it remains to prove conditions 2 and 4.

For every $a, b \in X_1$, note that $N(a) \cap N(b) \neq \emptyset$. Suppose for contradiction that $T_p \sqsubset G[X_1, Y]$, then there are $a, b \in X_1$ such that T_p is contained in the subgraph induced by the vertices $\{a, b\}$ and $(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))$. But then adding any $c \in N(a) \cap N(b)$ results in a forbidden copy of induced $F_{p,p}$, a contradiction. This proves condition 2.

Now for any r, s such that $r < q$ and $p < s \leq k$, suppose that $X_1 \neq \emptyset$ and $Z_{r,s} \sqsubset G[X_2, Y]$. Then there are $u_1, \ldots, u_r \in X_2$ and pairwise disjoint sets $V_1 \subseteq N(u_1), \ldots, V_r \subseteq N(u_r)$ such that for each i, $|V_i| = s$, for every $1 \leq j \leq i$, v_i is adjacent to all vertices in V_j, and for every $i < j \leq r$, v_i is adjacent to no vertices in V_j.

Let x be the vertex in X_1 with least degree, so that x was the last vertex to be added to X_1. Then $G[X_2 \cup \{x\}, Y]$ is left-connected but $G[X_2, Y]$ is left-disconnected, and x is the highest-degree vertex of $G[X_2 \cup \{x\}, Y]$ in $X_2 \cup \{x\}$. Since $u_1, \ldots, u_r$ are in the same connected component of $G[X_2, Y]$, but the graph $G[X_2, Y]$ is disconnected, it must be that there is $z \in X_2$ such that $N(z) \cap N(u_i) = \emptyset$ for all u_i. It is also the case that $|N(x) \cap N(z)| > |N(z)| - p \geq k - p \geq s - p$ by Proposition 7.2.14, since x has the highest degree in $X_2 \cup \{x\}$.

Observe that for each $V_i \subseteq N(u_i)$ it holds that $|N(x) \cap V_i| \geq s - p$ also by Proposition 7.2.14. Set $V_i' = V_i \cap N(x)$ for each $i \in [r]$, and set $V_{r+1}' = N(x) \cap N(z)$. Clearly, the graph induced by $\{u_1, \ldots, u_r, x\} \cup V_1' \cup V_2' \cup \ldots \cup V_r' \cup V_{r+1}'$ contains $Z_{r+1,s-p}$ as an induced subgraph. □

We will now use the above structural result to construct a suitable decomposition scheme for stable one-sided $F_{p,p}$-free bipartite graphs. Let $p, q \geq 1$ be fixed constants, let $k = (q+1)p$, and let $\mathcal{F}_{p,q}$ be the class of one-sided $F_{p,p}$-free bipartite graphs that do not contain $H_q^{\circ\circ}$ as an induced subgraph. Let $G = (X, Y, E) \in \mathcal{F}_{p,q}$. Using Lemma 7.2.15, we define a decomposition tree $\mathcal{T}$ for G inductively as follows. Let G_v be the induced subgraph of G associated with node v of the decomposition tree and write $X' \subseteq X$, $Y' \subseteq Y$ for its sets of vertices, so $G_v = G[X', Y']$. Graph G is associated with the root node of $\mathcal{T}$.

- If G_v is one-sided T_k-free, terminate the decomposition, so v is a leaf node (L-node) of the decomposition tree.
- If G_v is disconnected (in particular, if it is left-disconnected), then v is a D-node such that the children are the connected components of G_v.
- If G_v is connected and not one-sided T_k-free, then X' admits a partition $X' = X'_0 \cup X'_1 \cup X'_2$ satisfying the condition of Lemma 7.2.15. Since G_v is connected, $X'_0 \cup X'_1 \neq \emptyset$. Furthermore, since G_v is not one-sided T_k-free, $X'_2 \neq \emptyset$. Hence, v is a P-node with exactly two children v_1 and v_2, where $G_{v_1} = G[X'_0 \cup X'_1, Y']$ and $G_{v_2} = G[X'_2, Y']$. Observe that
  (1) G_{v_1} is one-sided T_k-free, and therefore v_1 is a leaf;
  (2) G_{v_2} is left-disconnected, and therefore v_2 is a D-node; furthermore, every vertex $x \in X'_2$ has degree at least k in G_{v_2} (otherwise it would be included in the set X'_0).

Proposition 7.2.16. *Let $\mathcal{Q}$ be the class of one-sided T_k-free bipartite graphs. Then the graphs in $\mathcal{F}_{p,q}$ admit $(\mathcal{Q}, 2)$-decomposition trees of depth at most $2q$.*

Proof. By definition, the above decomposition scheme produces $(\mathcal{Q}, 2)$-decomposition trees. In the rest of the proof we will establish the claimed bound on the depth of any such tree. Suppose, towards a contradiction, that there exists a graph $G = (X, Y, E) \in \mathcal{F}_{p,q}$ such that the decomposition tree $\mathcal{T}$ for G has depth at least $2q+1$. Let $\mathcal{P} = (v_0, v_1, v_2, \ldots, v_s)$ be a leaf-to-root path in $\mathcal{T}$ of length $s \geq 2q+1$, where v_0 is a leaf and v_s is the root. Denote by $G_{v_i} = G[X^i, Y^i]$ the graph corresponding to a node v_i in $\mathcal{P}$. By construction, all internal nodes of $\mathcal{P}$ are either D-nodes or P-nodes. Clearly, the path cannot contain two consecutive D-nodes, as any child of a D-node is a connected graph. Furthermore, a unique non-leaf child v_i of a P-node is a D-node, and every $x \in X^i$ has degree at least k in G_{v_i}. Consequently, P-nodes and D-nodes alternated along (the internal part of) $\mathcal{P}$.

Let $v_{i-1}, v_i, v_{i+1}, v_{i+2}$ be four internal nodes of $\mathcal{P}$, where v_{i-1} and v_{i+1} are D-nodes, and v_i and v_{i+2} are P-nodes. Recall that, since the parent v_{i+2} of v_{i+1} is a P-node, every vertex in X_{i+1} has degree at least k in $G_{v_{i+1}}$. Hence, since G_{v_i} is a connected component of $G_{v_{i+1}}$, every vertex in $X^i \subseteq X^{i+1}$ also has degree at least k. Let $X^i = X^i_0 \cup X^i_1 \cup X^i_2$ be the partition of X^i according to the decomposition rules. Since $X^i_0 \cup X^i_1 \neq \emptyset$ and $X^i_0 = \emptyset$, we conclude that $X^i_1 \neq \emptyset$. Therefore, by Lemma 7.2.15, if the $G_{v_{i-1}} = G[X^i_2, Y^i]$ contains $Z_{r,s}$ for some $r < q$ and $p < s \leq k$, then G_{v_i} contains $Z_{r+1,s-p}$.

Let v_t be the first D-node in $\mathcal{P}$. Note that $t \leq 2$. Every vertex in X^t has degree at least k in G_{v_t}, and therefore $Z_{1,k} \sqsubset G_{v_t}$. By induction, the above discussion implies that for $1 \leq i \leq q-1$, the graph $Z_{1+i,k-ip}$ is an induced subgraph of $G_{v_{t+2i-1}}$. Hence, since the length of $\mathcal{P}$ is at least $2q+1$, we have $H^{\circ\circ}_q = Z_{q,1} \sqsubset Z_{q,k-(q-1)p} \sqsubset G_{v_{t+2q-3}} \sqsubset G$, a contradiction. □

Lemma 7.2.17. *Let $p \in \mathbb{N}$ and let $\mathcal{F}$ be a stable class of one-sided $F_{p,p}$-free bipartite graphs. Then $\mathcal{F}$ admits a constant-size equality-based adjacency labeling scheme.*

Proof. Since $\mathcal{F}$ is stable, it does not contain $\mathcal{C}^{\circ\circ}$ as a subclass. Let q be the minimum number such that $H^{\circ\circ}_q \notin \mathcal{F}$. Let $k = (q+1)p$ and let $\mathcal{Q}$ be the class of one-sided T_k-free bipartite graphs. We have that $\mathcal{F} \subseteq \mathcal{F}_{p,q}$, and therefore, by Proposition 7.2.16, the graphs in $\mathcal{F}$ admit $(\mathcal{Q}, 2)$-decomposition trees of depth at most $2q$. Hence, by Lemma 7.2.11 and Lemma 7.2.2, $\mathcal{F}$ admits a constant-size equality-based adjacency labeling scheme. □

We conclude this section by showing that stable classes of $F^*_{p,p'}$-free graphs admit constant-size equality-based adjacency labeling schemes. For this we will use the above result for one-sided $F_{p,p}$-free graphs and the following

Proposition 7.2.18 ([All09], Corollary 9)**.** *Let $G = (X, Y, E)$ be a $F^*_{p,p}$-free bipartite graph. Then there is a partition $X = X_1 \cup X_2$ and $Y = Y_1 \cup Y_2$, where $|Y_2| \leq 1$, such that both $G[X_1, Y_1]$ and $\overline{\overline{G[X_2, Y_1]}}$ are one-sided $F_{p,p}$-free.*

Theorem 7.2.19. *For any constants $p, p' \geq 1$, a stable class $\mathcal{F}$ of $F^*_{p,p'}$-free bipartite graphs admits a constant-size equality-based adjacency labeling scheme.*

Proof. As before, since $\mathcal{F}$ is stable, it does not contain $\mathcal{C}^{\circ\circ}$ as a subclass. Let q be the minimum number such that $H^{\circ\circ}_q \notin \mathcal{F}$, and assume without loss of generality that $p \geq p'$. It follows that $\mathcal{F}$ is a subclass of $(F^*_{p,p}, H^{\circ\circ}_q)$-free bipartite graphs. Let $G = (X, Y, E)$ be a member of this class. Let $X = X_1 \cup X_2, Y = Y_1 \cup Y_2$ be the partition given by Proposition 7.2.18. We assign labels as follows.

We start the label for each vertex with a one-bit prefix indicating whether it is in X or Y. We then append the following labels. For $x \in X$, we use another one-bit prefix that is equal to 1 if x is adjacent to the unique vertex Y_2, and 0 otherwise. Then, we use one more one-bit prefix to indicate whether $x \in X_1$ or $x \in X_2$. If $x \in X_1$, complete the label by using the labeling scheme of Lemma 7.2.17 for $G[X_1, Y_1]$. If $x \in X_2$, complete the label by using the labeling scheme of Lemma 7.2.17 for $\overline{\overline{G[X_2, Y_1]}}$.

For $y \in Y$, use a one-bit prefix to indicate whether $y \in Y_2$. If $y \in Y_1$ then concatenate the two labels for y obtained from the labeling scheme of Lemma 7.2.17 for $G[X_1, Y_1]$ and $\overline{\overline{G[X_2, Y_1]}}$.

The decoder first checks if x, y are in opposite parts. Now assume $x \in X, y \in Y$. The decoder checks if $y \in Y_2$ and outputs the appropriate value using the appropriate prefix from the label of x. Then if $x \in X_1$, it uses the labels of x and y in $G[X_1, Y_1]$; otherwise it uses the labels of x and y in $\overline{\overline{G[X_2, Y_1]}}$ and flips the output. □

P_7-Free Bipartite Graphs

In this section, we prove Theorem 1.3.26 for P_7-free bipartite graphs by developing a constant-size equality-based adjacency labeling scheme for stable classes of P_7-free bipartite graphs

In the below definition, for two disjoint sets of vertices A and B we say that A is *complete* to B if every vertex in A is adjacent to every vertex in B; we also say that A is *anticomplete* to B if there are no edges between A and B.

Definition 7.2.20 (Chain Decomposition)**.** See Figure 7.5 for an illustration of the chain decomposition. Let $G = (X, Y, E)$ be a bipartite graph and $k \in \mathbb{N}$. We say that G admits a *k-chain decomposition* if one of the parts, say X, can be partitioned into subsets $A_1, \ldots, A_k, C_1, \ldots, C_k$ and the other part Y can be partitioned into subsets $B_1, \ldots, B_k$, $D_1, \ldots, D_k$ in such a way that:

- For every $i \leq k-1$, the sets A_i, B_i, C_i, D_i are non-empty. For $i = k$, at least one of the sets A_i, B_i, C_i, D_i must be non-empty.
- For each $i = 1, \ldots, k$,
  - every vertex of B_i has a neighbour in A_i;
  - every vertex of D_i has a neighbour in C_i;

- For each $i = 2, \ldots, k-1$,
  - every vertex of A_i has a non-neighbour in B_{i-1};
  - every vertex of C_i has a non-neighbour in D_{i-1};
- For each $i = 1, \ldots, k$,
  - the set A_i is anticomplete to B_j for $j > i$ and is complete to B_j for $j < i-1$;
  - the set C_i is anticomplete to D_j for $j > i$ and is complete to D_j for $j < i-1$;
- For each $i = 1, \ldots, k$,
  - the set A_i is complete to D_j for $j < i$, and is anticomplete to D_j for $j \geq i$;
  - the set C_i is complete to B_j for $j < i$, and is anticomplete to B_j for $j \geq i$.

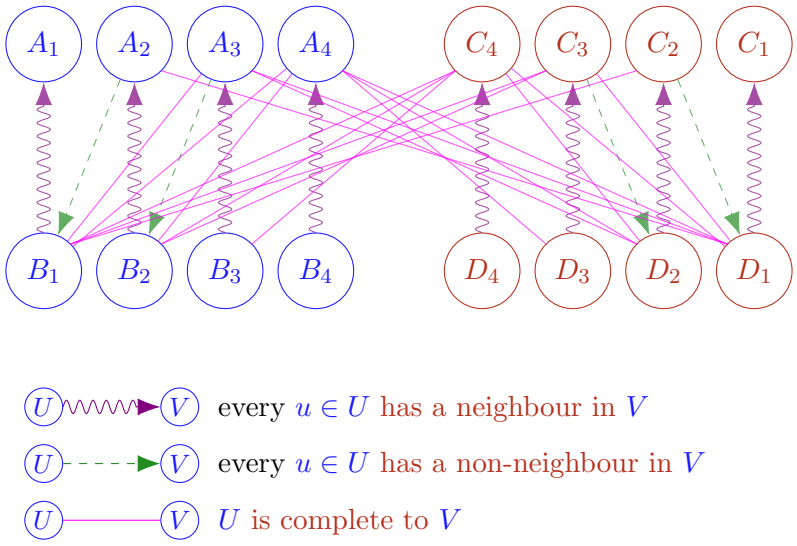


Figure 7.5: Example of a 4-chain decomposition.

Remark 7.2.21. In the case of a 2-chain decomposition of a connected P_7-free bipartite graphs, we will also need the fact that every vertex in A_2 and every vertex in A_1 have a neighbour in common; and every vertex in C_2 and every vertex in C_1 have a neighbour in common. This is not stated explicitly in [LZ17], but easily follows from a proof in [LZ17]. Since the neighbourhood of every vertex in A_1 lies entirely in B_1, the above fact implies that every vertex in A_2 has a neighbour in B_1. Similarly, the neighbourhood of every vertex in C_1 lies entirely in D_1, and therefore every vertex in C_2 has a neighbour in D_1.

Theorem 7.2.22 ([LZ17]). *Let $G = (X, Y, E)$ be a P_7-free bipartite graph such that both G and $\overline{\overline{G}}$ are connected. Then G or $\overline{\overline{G}}$ admits a k-chain decomposition for some $k \geq 2$.*

Lemma 7.2.23. *Let $G = G(X, Y, E)$ be a connected P_7-free bipartite graph of chain number c that admits a k-chain decomposition for some $k \geq 2$. Then there exists a partition of X into $p \leq 2(c+1)$ sets $X_1, X_2, \ldots, X_p$, and a partition of Y into $q \leq 2(c+1)$ sets $Y_1, Y_2, \ldots, Y_q$ such that, for any $i \in [p], j \in [q]$,*

$$\mathsf{ch}(G[X_i, Y_j]) < \mathsf{ch}(G)\,.$$

Proof. Assume, without loss of generality, that X is partitioned into subsets $A_1, \ldots, A_k$, $C_1, \ldots, C_k$ and Y into subsets $B_1, \ldots, B_k$, $D_1, \ldots, D_k$ satisfying Definition 7.2.20. Since at least one of the sets A_k, B_k, C_k, D_k is non-empty, and every vertex in B_k has a neighbour in A_k and every vertex in D_k has a neighbour in C_k, at least one of A_k and C_k is non-empty. Without loss of generality we assume that A_k is not empty. It is straightforward to check by definition that for any vertices $a_2 \in A_2, a_3 \in A_3, \ldots, a_k \in A_k$, and $d_1 \in D_1, d_2 \in D_2, \ldots, d_{k-1} \in D_{k-1}$ the subgraph of G induced by $\{a_2, a_3, \ldots, a_k, d_1, d_2, \ldots, d_{k-1}\}$ is isomorphic to $H^{\circ\circ}_{k-1}$, which implies that k is at most $c+1$. We also observe that any path from a vertex in A_1 to a vertex in D_1 contains at least 4 vertices, and hence G contains $H^{\circ\circ}_2$ and $\mathsf{ch}(G) \geq 2$. We split our analysis in two cases.

Case 1. $k \geq 3$. We will show that for any $X' \in \{A_1, \ldots, A_k, C_1, \ldots, C_k\}$ and $Y' \in \{B_1, \ldots, B_k, D_1, \ldots, D_k\}$, $\mathsf{ch}(G[X', Y']) < \mathsf{ch}(G)$. Since $\mathsf{ch}(G) \geq 2$, the chain number of a biclique is 1, and the chain number of a co-biclique is 0, we need only to verify pairs of sets that can induce a graph which is neither a biclique nor a co-biclique. By Definition 7.2.20, these are the pairs (A_i, B_i), (C_i, D_i) for $i \in [k]$ and (A_i, B_{i-1}), (C_i, D_{i-1}) for $i \in \{2, \ldots, k\}$.

We start with the pair (A_1, B_1). Since D_2 is anticomplete to A_1, and C_2 is complete to B_1, for any vertex $d_2 \in D_2$ and its neighbour $c_2 \in C_2$ we have that $\mathsf{ch}(G[A_1, B_1]) < \mathsf{ch}(G[A_1 \cup \{c_2\}, B_1 \cup \{d_2\}]) \leq \mathsf{ch}(G)$. Similarly, since D_1 is complete to all $A_2, A_3, \ldots, A_k$, and C_1 is anticomplete to all $B_1, B_2, \ldots, B_k$, addition of a vertex $d_1 \in D_1$ and its neighbour $c_1 \in C_1$ to any of the graphs $G[A_i, B_i]$ or $G[A_i, B_{i-1}]$ for $i \in \{2, \ldots, k\}$ strictly increases the chain number of that graph. Symmetric arguments establish the desired conclusion for the pairs of sets (C_i, D_i), $i \in [k]$, and (C_i, D_{i-1}), $i \in \{2, \ldots, k\}$.

In this case, $A_1, \ldots, A_k, C_1, \ldots, C_k$ and $B_1, \ldots, B_k, D_1, \ldots, D_k$ are the desired partitions of X and Y respectively.

Case 2. $k = 2$. Assume first that both A_2 and C_2 are non-empty. Let c_2 be a vertex in C_2, d_1 be a neighbour of c_2 in D_1 (which exists by Remark 7.2.21), and c_1 be a neighbour of d_1 in C_1. Since C_2 is complete to B_1 and D_1 is anticomplete to A_1,

$\mathsf{ch}(G[A_1, B_1]) < \mathsf{ch}(G[A_1 \cup \{c_2\}, B_1 \cup \{d_1\}]) \leq \mathsf{ch}(G)$. Similarly, because C_1 is anticomplete to both B_1 and B_2 and D_1 is complete to A_2, we have that $\mathsf{ch}(G[A_2, B_1]) < \mathsf{ch}(G[A_2 \cup \{c_1\}, B_1 \cup \{d_1\}]) \leq \mathsf{ch}(G)$ and $\mathsf{ch}(G[A_2, B_2]) < \mathsf{ch}(G[A_2 \cup \{c_1\}, B_2 \cup \{d_1\}]) \leq \mathsf{ch}(G)$. Using symmetric arguments we can show that the chain number of each of $G[C_1, D_1]$, $G[C_2, D_1]$, and $G[C_2, D_2]$ is strictly less than the chain number of G. All other pairs of sets (X', Y'), where $X' \in \{A_1, A_2, C_1, C_2\}$ and $Y' \in \{B_1, B_2, D_1, D_2\}$ induce either a biclique or a co-biclique, and therefore $\mathsf{ch}(G[X', Y']) < \mathsf{ch}(G)$. In this case, A_1, A_2, C_1, C_2 and B_1, B_2, D_1, D_2 are the desired partitions of X and Y respectively.

The case when one of A_2 and C_2 is empty requires a separate analysis. Assume that $A_2 \neq \emptyset$ and $C_2 = \emptyset$. The case when $A_2 = \emptyset$ and $C_2 \neq \emptyset$ is symmetric and we omit the details. Since C_2 is empty, D_2 is also empty and therefore A_1, A_2, C_1 is a partition of X, and B_1, B_2, D_1 is a partition of Y. Let a_2 be a vertex in A_2, d_1 be a vertex in D_1, and c_1 be a neighbour of d_1 in C_1. Let B_1' be the neighbourhood of a_2 in B_1 and let $B_1'' = B_1 \setminus B_1'$. We claim that A_1, A_2, C_1 and B_1', B_1'', B_2, D_1 are the desired partitions of X and Y respectively. All the pairs of sets, except (A_1, B_1') and (A_1, B_1''), can be treated as before and we skip the details. For (A_1, B_1'), we observe that a_2 is complete to B_1' and D_1 is anticomplete to A_1, and hence $\mathsf{ch}(G[A_1, B_1']) < \mathsf{ch}(G[A_1 \cup \{a_2\}, B_1' \cup \{d_1\}]) \leq \mathsf{ch}(G)$.

To establish the desired property for (A_1, B_1''), we first observe that by Remark 7.2.21 every vertex in A_1 has a neighbour in common with a_2, and therefore every vertex in A_1 has a neighbour in B_1'. If $G[A_1, B_1'']$ is edgeless the property holds trivially. Otherwise, let $P \subseteq A_1$ and $Q \subseteq B_1''$ be such that $P \cup Q$ induces a $H_s^{\circ\circ}$ in $G[A_1, B_1'']$, where $s \geq 1$ is the chain number of the latter graph. Let x be the vertex in P that has degree 1 in $G[P, Q]$, and let y be a neighbour of x in B_1'. We claim that y is complete to P. Indeed, if y is not adjacent to some $x' \in P$, then $x', z, x, y, a_2, d_1, c_1$ would induce a forbidden P_7, where z is the vertex in Q that is adjacent to every vertex in P. Consequently, $G[P \cup \{a_2\}, Q \cup \{y\}]$ is isomorphic to $H_{s+1}^{\circ\circ}$, and therefore $\mathsf{ch}(G[A_1, B_1'']) < \mathsf{ch}(G)$. □

For two pairs of numbers (a, b) and (c, d) we write $(a, b) \preceq (c, d)$ if $a \leq c$ and $b \leq d$, and we write $(a, b) \prec (c, d)$ if at least one of the inequalities is strict.

Lemma 7.2.24. *Let $G = G(X, Y, E)$ be a P_7-free bipartite graph such that both G and $\overline{\overline{G}}$ are connected, and let c be the chain number of G. Then there exists a partition of X into $p \leq 2(c+2)$ sets $X_1, X_2, \ldots, X_p$, and a partition of Y into $q \leq 2(c+2)$ sets $Y_1, Y_2, \ldots, Y_q$ such that for any $i \in [p]$, $j \in [q]$*

$$\left(\mathsf{ch}(G_{i,j}), \mathsf{ch}(\overline{\overline{G_{i,j}}})\right) \prec \left(\mathsf{ch}(G), \mathsf{ch}(\overline{\overline{G}})\right),$$

where $G_{i,j} = G[X_i, Y_j]$.

Proof. It is easy to verify that for any $k \geq 2$, the graph $\overline{\overline{H_k^{\circ\circ}}}$ contains the half graph $H_{k-1}^{\circ\circ}$, which implies that $\mathsf{ch}(\overline{\overline{G}}) \leq \mathsf{ch}(G) + 1 = c + 1$. Furthermore, the bipartite complement of a P_7 is again P_7, and hence the bipartite complement of any P_7-free bipartite graph is also P_7-free.

By Theorem 7.2.22, G or $\overline{\overline{G}}$ admits a k-chain decomposition for some $k \geq 2$. Therefore, by Lemma 7.2.23 applied to either G or $\overline{\overline{G}}$, there exist a partition of X into at most $p \leq 2(c+2)$ sets $X_1, X_2, \ldots, X_p$, and a partition of Y into at most $q \leq 2(c+2)$ sets $Y_1, Y_2, \ldots, Y_q$ such that either $\mathsf{ch}(G_{i,j}) < \mathsf{ch}(G)$ holds for any $i \in [p]$, $j \in [q]$, or $\mathsf{ch}(\overline{\overline{G_{i,j}}}) < \mathsf{ch}(\overline{\overline{G}})$ holds for any $i \in [p]$, $j \in [q]$. This together with the fact that the chain number of an induced subgraph of a graph is never larger than the chain number of the graph, implies the lemma. □

We are now ready to specify a decomposition scheme for P_7-free bipartite graphs. Let $G = (X, Y, E)$ be a P_7-free bipartite graph of chain number c. Let $\mathcal{Q}$ be the class consisting of bicliques and co-bicliques. We define a $(\mathcal{Q}, 2(c+2))$-decomposition tree $\mathcal{T}$ for G inductively as follows. Let G_v be the induced subgraph of G associated with node v of the decomposition tree and write $X' \subseteq X$, $Y' \subseteq Y$ for its sets of vertices, so $G_v = G[X', Y']$. Graph G is associated with the root node of $\mathcal{T}$.

- If G_v belongs to $\mathcal{Q}$, then terminate the decomposition, so v is a leaf node (L-node) of the decomposition tree.
- If G_v does not belong to $\mathcal{Q}$ and is disconnected, then v is a D-node such that the children are the connected components of G_v.
- If G_v does not belong to $\mathcal{Q}$, is connected, and $\overline{\overline{G_v}}$ is disconnected, then v is a $\overline{D}$-node. There are sets $X'_1, \ldots, X'_t \subseteq X'$ and $Y'_1, \ldots, Y'_t \subseteq Y'$ such that $\overline{\overline{G[X'_1, Y'_1]}}, \ldots, \overline{\overline{G[X'_t, Y'_t]}}$ are the connected components of $\overline{\overline{G_v}}$. The children of this node are $G[X'_1, Y'_1]$, $\ldots$, $G[X'_t, Y'_t]$.
- If G_v does not belong to $\mathcal{Q}$, and neither G_v, nor $\overline{\overline{G_v}}$ is disconnected, then v is a P-node. Let $X'_1, X_2, \ldots, X'_p$ be a partition of X' into $p \leq 2(c+2)$ sets, and $Y'_1, Y_2, \ldots, Y'_q$ be a partition of Y' into $q \leq 2(c+2)$ sets, as in Lemma 7.2.24. The children of this node are $G[X'_i, Y'_j]$, $i \in [p]$, $j \in [q]$.

Claim 7.2.25. *Let $\mathcal{T}$ be a decomposition tree as a above, and let $G_i = G[X_i, Y_i]$, $i = 1, 2, 3$, be internal nodes in $\mathcal{T}$ such that G_3 is the parent of G_2 which is in turn the parent of G_1. Then*

(1) *one of* G_3, G_2, *and* G_1 *is a* P*-node, or* G_i *is* $\overline{D}$*-node and* G_{i-1} *is a* D*-node for some* $i \in \{3, 2\}$*;*

(2) *if* G_3 *is a* $\overline{D}$*-node and* G_2 *is a* D*-node, then* $\mathsf{ch}(G_1) < \mathsf{ch}(G_3)$.

Proof. We start by proving the first statement. Observe that every child a D-node is a connected graph, and therefore it is not a D-node. Similarly, the bipartite complement of every child of a $\overline{D}$-node is a connected graph, and therefore a $\overline{D}$-node cannot have a $\overline{D}$-node as a child. Hence, if none of G_3, G_2, G_1 is a P-node, either G_3 is a $\overline{D}$-node and therefore G_2 is a D-node, or G_3 is a D-node, in which case G_2 is a $\overline{D}$-node and G_1 is a D-node. In both cases we have a pair of parent-child nodes, where the parent is a $\overline{D}$-node and the child is a D-node.

To prove the second statement, let now G_3 be a $\overline{D}$-node and G_2 be a D-node, i.e. $G[X_2, Y_2]$ is disconnected, while $G[X_3, Y_3]$ is connected, but its bipartite complement is disconnected. Then there are sets $X_1' \subseteq X_2 \setminus X_1$ and $Y_1' \subseteq Y_2 \setminus Y_1$ such that $G[X_1', Y_1']$ and $G[X_1, Y_1]$ are connected components of $G[X_2, Y_2]$ and at least one of X_1', Y_1' is non-empty. Also at least one of the sets $X_2' = X_3 \setminus X_2$ and $Y_2' = Y_3 \setminus Y_2$ is non-empty, and every vertex in X_2' is adjacent in G to every vertex in Y_2, and every vertex in Y_2' is adjacent in G to every vertex in X_2. If exactly one of the sets X_1' and Y_1' is non-empty, say Y_1', then X_2' is also non-empty, as otherwise $G[X_3, Y_3]$ would be disconnected. Hence, any vertices $x' \in X_2'$ and $y' \in Y_1'$ can augment any half graph $H_k^{\circ\circ}$ in $G[X_1, Y_1]$ into a half graph $H_{k+1}^{\circ\circ}$. Consequently, $\mathsf{ch}(G[X_1, Y_1]) < \mathsf{ch}(G[\{x'\} \cup X_1, \{y'\} \cup Y_1]) \leq \mathsf{ch}(G_3)$. If both sets X_1' and Y_1' are non-empty, the argument is similar and we omit the details. □

Theorem 7.2.26. *Let* $\mathcal{F}$ *be a stable class of* P_7*-free bipartite graphs. Then* $\mathcal{F}$ *admits a constant-size equality-based adjacency labeling scheme.*

Proof. Since $\mathcal{F}$ is stable, it does not contain $\mathcal{C}^{\circ\circ}$ as a subclass. Let c be the maximum number such that $H_c^{\circ\circ} \in \mathcal{F}$, and let $G = (X, Y, E)$ be an arbitrary graph from $\mathcal{F}$.

By the above discussion G admits a $(\mathcal{Q}, 2(c+2))$-decomposition tree, where $\mathcal{Q}$ is the class consisting of bicliques and co-bicliques, and every P-node $G' = G[X', Y']$ is specified by the partition $X_1', \ldots, X_p'$ of X' and the partition $Y_1', \ldots, Y_q'$ of Y' as in Lemma 7.2.24, where p and q are bounded from above by $2(c+2)$.

We claim that the depth of such a decomposition tree is at most $6c$. To show this we associate with every node G' the pair $\left(\mathsf{ch}(G'), \mathsf{ch}(\overline{\overline{G'}})\right)$ and we will prove that if the length of the path from the root G to a node G' is at least $6c$, then $\mathsf{ch}(G') \leq 1$ or $\mathsf{ch}(\overline{\overline{G'}}) \leq 1$, which means that G' is either a biclique or a co-biclique, and therefore is a leaf node.

Let $\mathcal{P}$ be the path from the root to the node G'. By Claim 7.2.25 (1), among any three consecutive nodes of the path, there exists a P-node, or a pair of nodes labeled with $\overline{D}$ and D respectively such that the $\overline{D}$-node is the parent of the D-node. In the former case, by Lemma 7.2.24, for the child node H' of the P-node H on the path $\mathcal{P}$, we have $\left(\mathsf{ch}(H'), \mathsf{ch}(\overline{\overline{H'}})\right) \prec \left(\mathsf{ch}(H), \mathsf{ch}(\overline{\overline{H}})\right)$. In the latter case, by Claim 7.2.25 (2), the child of the D-node on the path $\mathcal{P}$ has the chain number strictly less than that of the D-node. In other words, for every node H in the path $\mathcal{P}$ and its ancestor H' at distance 3 from H, we have $\left(\mathsf{ch}(H'), \mathsf{ch}(\overline{\overline{H'}})\right) \prec \left(\mathsf{ch}(H), \mathsf{ch}(\overline{\overline{H}})\right)$.

Now, since for the root node G we have $\left(\mathsf{ch}(G), \mathsf{ch}(\overline{\overline{G}})\right) \prec (c, c+1)$, if $\mathcal{P}$ has length at least $6c$, then $\mathsf{ch}(G') \leq 1$ or $\mathsf{ch}(\overline{\overline{G'}}) \leq 1$, as required. The result now follows from Lemma 7.2.2 and a simple observation that class $\mathcal{Q}$ admits a constant-size equality-based adjacency labeling scheme. □

References

[AAA+22] Bogdan Alecu, Vladimir E. Alekseev, Aistis Atminas, Vadim Lozin, and Viktor Zamaraev. Graph parameters, implicit representations and factorial properties. In submission, 2022.

[AAH+17] Mikkel Abrahamsen, Stephen Alstrup, Jacob Holm, Mathias Bæk Tejs Knudsen, and Morten Stöckel. Near-Optimal Induced Universal Graphs for Bounded Degree Graphs. In *44th International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 80 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 128:1–128:14, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[AAL21] Bogdan Alecu, Aistis Atminas, and Vadim Lozin. Graph functionality. *Journal of Combinatorial Theory, Series B*, 147:139–158, 2021.

[ABR05] Stephen Alstrup, Philip Bille, and Theis Rauhe. Labeling schemes for small distances in trees. *SIAM Journal on Discrete Mathematics*, 19(2):448–462, 2005.

[AC06] Nir Ailon and Bernard Chazelle. Information theory in property testing and monotonicity testing in higher dimension. *Information and Computation*, 204(11):1704–1717, 2006.

[ACFL16] Aistis Atminas, Andrew Collins, Jan Foniok, and Vadim V Lozin. Deciding the bell number for hereditary graph properties. *SIAM Journal on Discrete Mathematics*, 30(2):1015–1031, 2016.

[ACLZ15] Aistis Atminas, Andrew Collins, Vadim Lozin, and Victor Zamaraev. Implicit representations and factorial properties of graphs. *Discrete Mathematics*, 338(2):164–179, 2015.

[ADK17] Stephen Alstrup, Søren Dahlgaard, and Mathias Bæk Tejs Knudsen. Optimal induced universal graphs and adjacency labeling for trees. *Journal of the ACM (JACM)*, 64(4):1–22, 2017.

[ADPR03] Noga Alon, Seannie Dar, Michal Parnas, and Dana Ron. Testing of clustering. *SIAM Journal on Discrete Mathematics*, 16(3):393–417, 2003.

[AFZ19] Noga Alon, Jacob Fox, and Yufei Zhao. Efficient arithmetic regularity and removal lemmas for induced bipartite patterns. *Discrete Analysis*, 2019:14, 2019.

[AGG+19] Ittai Abraham, Cyril Gavoille, Anupam Gupta, Ofer Neiman, and Kunal Talwar. Cops, robbers, and threatening skeletons: Padded decomposition for minor-free graphs. *SIAM Journal on Computing*, 48(3):1120–1145, 2019.

[AGHP16a] Stephen Alstrup, Cyril Gavoille, Esben Bistrup Halvorsen, and Holger Petersen. Simpler, faster and shorter labels for distances in graphs. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 338–350. SIAM, 2016.

[AGHP16b] Stephen Alstrup, Inge Li Gørtz, Esben Bistrup Halvorsen, and Ely Porat. Distance labeling schemes for trees. In *43rd International Colloquium on Automata, Languages, and Programming (ICALP)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2016.

[AHW16] Noga Alon, Rani Hod, and Amit Weinstein. On active and passive testing. *Combinatorics, Probability and Computing*, 25:1–20, 2016.

[AK08] Alexandr Andoni and Robert Krauthgamer. Distance estimation protocols for general metrics. https://www.cs.columbia.edu/~andoni/papers/de.pdf, 2008.

[AKR18] Alexandr Andoni, Robert Krauthgamer, and Ilya P. Razenshteyn. Sketching and embedding are equivalent for norms. *SIAM Journal on Computing*, 47(3):890–916, 2018.

[Ale92] Vladimir Evgen'evich Alekseev. Range of values of entropy of hereditary classes of graphs. *Diskretnaya Matematika*, 4(2):148–157, 1992.

[Ale97] Vladimir Evgen'evich Alekseev. On lower layers of a lattice of hereditary classes of graphs. *Diskretnyi Analiz i Issledovanie Operatsii*, 4(1):3–12, 1997.

[All09] Peter Allen. Forbidden induced bipartite graphs. *Journal of Graph Theory*, 60(3):219–241, 2009.

[Alo86] Noga Alon. Covering graphs by the minimum number of equivalence relations. *Combinatorica*, 6(3):201–206, 1986.

[AMS99] Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. *Journal of Computer and System Sciences*, 58(1):137–147, 1999.

[AMY16] Noga Alon, Shay Moran, and Amir Yehudayoff. Sign rank versus VC dimension. In *Conference on Learning Theory (COLT)*, pages 47–80, 2016.

[AR02] Stephen Alstrup and Theis Rauhe. Small induced-universal graphs and compact implicit graph representations. In *Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS 2002)*, pages 53–62. IEEE, 2002.

[AR14] David Adjiashvili and Noy Rotbart. Labeling schemes for bounded degree graphs. In *International Colloquium on Automata, Languages, and Programming (ICALP)*, pages 375–386. Springer, 2014.

[ARSV06] Noga Alon, Radoš Radoičić, Benny Sudakov, and Jan Vondrák. A ramsey-type result for the hypercube. *Journal of Graph Theory*, 53(3):196–208, 2006.

[AS16] Noga Alon and Joel H Spencer. *The probabilistic method*. John Wiley & Sons, 2016.

[Ass82] Patrice Assouad. Sur la distance de nagata. *CR Acad. Paris*, 294:31–34, 1982.

[ASW15] Emmanuel Abbe, Amir Shpilka, and Avi Wigderson. Reed–Muller codes for random erasures and errors. *IEEE Transactions on Information Theory*, 61(10):5229–5252, 2015.

[ATYY17] Anurag Anshu, Dave Touchette, Penghui Yao, and Nengkun Yu. Exponential separation of quantum communication and classical information. In *Proceedings of the ACM SIGACT Symposium on Theory of Computing (STOC)*, pages 277–288, 2017.

[Bar19] Dalya Baron. Machine learning in astronomy: A practical overview. https://arXiv.org/abs/1904.07248, 2019.

[BB16] Aleksandrs Belovs and Eric Blais. A polynomial lower bound for testing monotonicity. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*, pages 1021–1032, 2016.

[BB20] Eric Blais and Abhinav Bommireddi. On testing and robust characterizations of convexity. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2020)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.

[BBBY12] Maria-Florina Balcan, Eric Blais, Avrim Blum, and Liu Yang. Active property testing. In *Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 21–30. IEEE, 2012.

[BBM12] Eric Blais, Joshua Brody, and Kevin Matulef. Property testing lower bounds via communication complexity. *Computational Complexity*, 21(2):311–358, 2012.

[BBM+20] Alexander R. Block, Simina Branzei, Hemanta K. Maji, Himanshi Mehta, Tamalika Mukherjee, and Hai H. Nguyen. P_4-free partition and cover numbers and application. Cryptology ePrint Archive, Report 2020/1605, 2020. https://ia.cr/2020/1605.

[BBW00] József Balogh, Béla Bollobás, and David Weinreich. The speed of hereditary properties of graphs. *Journal of Combinatorial Theory, Series B*, 79(2):131–156, 2000.

[BBW05] József Balogh, Béla Bollobás, and David Weinreich. A jump to the bell number for hereditary graph properties. *Journal of Combinatorial Theory, Series B*, 95(1):29–48, 2005.

[BCO+15] Eric Blais, Clément Canonne, Igor Oliveira, Rocco Servedio, and Li-Yang Tan. Learning circuits with few negations. *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, page 512, 2015.

[BCS20] Hadley Black, Deeparnab Chakrabarty, and C Seshadhri. Domain reduction for monotonicity testing: A $o(d)$ tester for boolean functions in d-dimensions. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1975–1994, 2020.

[BCS+21] Abdul Basit, Artem Chernikov, Sergei Starchenko, Terence Tao, and Chieu-Minh Tran. Zarankiewicz's problem for semilinear hypergraphs. In *Forum of Mathematics, Sigma*, volume 9. Cambridge University Press, 2021.

[BEHW89] Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K Warmuth. Learnability and the Vapnik-Chervonenkis dimension. *Journal of the ACM (JACM)*, 36(4):929–965, 1989.

[BFH21] Eric Blais, Renato Ferreira Pinto Jr., and Nathaniel Harms. VC dimension and distribution-free sample-based testing. In *Proceedings of the ACM SIGACT Symposium on Theory of Computing (STOC)*, pages 504–517, 2021.

[BGK+21] Édouard Bonnet, Colin Geniet, Eun Jung Kim, Stéphan Thomassé, and Rémi Watrigant. Twin-width II: small classes. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1977–1996. SIAM, 2021.

[BGP20] Marthe Bonamy, Cyril Gavoille, and Michał Pilipczuk. Shorter labeling schemes for planar graphs. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 446–462. SIAM, 2020.

[BH18] Avrim Blum and Lunjia Hu. Active tolerant testing. In *Conference On Learning Theory (COLT)*, 2018.

[BHMZ20] Olivier Bousquet, Steve Hanneke, Shay Moran, and Nikita Zhivotovskiy. Proper learning, Helly number, and an optimal SVM bound. In *Conference on Learning Theory*, pages 582–609. PMLR, 2020.

[BKTW20] Édouard Bonnet, Eun Jung Kim, Stéphan Thomassé, and Rémi Watrigant. Twin-width i: tractable fo model checking. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 601–612. IEEE, 2020.

[BL98] Shai Ben-David and Ami Litman. Combinatorial variability of Vapnik-Chervonenkis classes with applications to sample compression schemes. *Discrete Applied Mathematics*, 86(1):3–25, 1998.

[Bla09] Eric Blais. Testing juntas nearly optimally. In Michael Mitzenmacher, editor, *Proceedings of the ACM Symposium on Theory of Computing (STOC)*, pages 151–158. ACM, 2009.

[BLM13] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration inequalities: A nonasymptotic theory of independence*. Oxford university press, 2013.

[BMR16] Piotr Berman, Meiram Murzabulatov, and Sofya Raskhodnikova. Tolerant testers of image properties. In *43rd International Colloquium on Automata, Languages, and Programming (ICALP)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

[BMR19] Piotr Berman, Meiram Murzabulatov, and Sofya Raskhodnikova. The power and limitations of uniform samples in testing properties of figures. *Algorithmica*, 81(3):1247–1266, 2019.

[BOW10] Eric Blais, Ryan O'Donnell, and Karl Wimmer. Polynomial regression under arbitrary product distributions. *Machine Learning*, 80(2-3):273–294, 2010.

[BR92] Avrim L Blum and Ronald L Rivest. Training a 3-node neural network is NP-complete. *Neural Networks*, 5(1):117–127, 1992.

[BRY14] Eric Blais, Sofya Raskhodnikova, and Grigory Yaroslavtsev. Lower bounds for testing properties of functions over hypergrid domains. In *Proceedings of the IEEE 29th Conference on Computational Complexity (CCC)*, pages 309–320, 2014.

[Bsh19] Nader H. Bshouty. Almost Optimal Distribution-Free Junta Testing. In *34th Computational Complexity Conference (CCC)*, volume 137, pages 2:1–2:13. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2019.

[Bsh20] Nader H. Bshouty. Almost Optimal Testers for Concise Representations. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2020)*, volume 176, pages 5:1–5:20. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2020.

[BT95] Béla Bollobás and Andrew Thomason. Projections of bodies and hereditary properties of hypergraphs. *Bulletin of the London Mathematical Society*, 27(5):417–424, 1995.

[But09] Steve Butler. Induced-universal graphs for graphs with bounded maximum degree. *Graphs and Combinatorics*, 25(4):461–468, 2009.

[BY19] Eric Blais and Yuichi Yoshida. A characterization of constant-sample testable properties. *Random Structures & Algorithms*, 55(1):73–88, 2019.

[CCMW19] Jérémie Chalopin, Victor Chepoi, Shay Moran, and Manfred K Warmuth. Unlabeled sample compression schemes and corner peelings for ample and

maximum classes. In *46th International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 132, page 34. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2019.

[CDJS17] Deeparnab Chakrabarty, Kashyap Dixit, Madhav Jha, and C Seshadhri. Property testing on product distributions: Optimal testers for bounded derivative properties. *ACM Transactions on Algorithms (TALG)*, 13(2):1–30, 2017.

[CDST15] Xi Chen, Anindya De, Rocco A Servedio, and Li-Yang Tan. Boolean function monotonicity testing requires (almost) $n^{1/2}$ non-adaptive queries. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*, pages 519–528, 2015.

[CFSS17] Xi Chen, Adam Freilich, Rocco A Servedio, and Timothy Sun. Sample-based high-dimensional convexity testing. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2017)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017.

[CGG+19] Clément L. Canonne, Elena Grigorcscu, Siyao Guo, Akash Kumar, and Karl Wimmer. Testing k-monotonicity: The rise and fall of boolean functions. *Theory of Computing*, 15(1):1–55, 2019.

[CGM11] Sourav Chakraborty, David García-Soriano, and Arie Matsliah. Efficient sample extractors for juntas with applications. In *38th International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 6755 of *Lecture Notes in Computer Science*, pages 545–556. Springer, 2011.

[Cha18] Maurice Chandoo. A complexity theory for labeling schemes. *arXiv preprint arXiv:1802.02819*, 2018.

[Cha20] Amit Chakrabarti. One-way randomized communication complexity of greater-than. Theoretical Computer Science Stack Exchange, 2020, https://cstheory.stackexchange.com/q/48110. URL: https://cstheory.stackexchange.com/q/48110 (version: 2020-12-30).

[Chu90] Fan RK Chung. Universal graphs and induced-universal graphs. *Journal of Graph Theory*, 14(4):443–454, 1990.

[CHZZ22] TsunMing Cheung, Hamed Hatami, Rosie Zhao, and Itai Zilberstein. Boolean functions with small approximate spectral norm. *Electronic Colloquium on Computational Complexity (ECCC)*, TR22-041, 2022.

[CLR20] Victor Chepoi, Arnaud Labourel, and Sébastien Ratel. On density of subgraphs of Cartesian products. *Journal of Graph Theory*, 93(1):64–87, 2020.

[CLV19] Arkadev Chattopadhyay, Shachar Lovett, and Marc Vinyals. Equality alone does not simulate randomness. In *34th Computational Complexity Conference (CCC 2019)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2019.

[CMK19] Mónika Csikós, Nabil H Mustafa, and Andrey Kupavskii. Tight lower bounds on the VC-dimension of geometric set systems. *Journal of Machine Learning Research*, 20(81):1–8, 2019.

[CP22] Xi Chen and Shyamal Patel. Distribution-free testing for halfspaces (almost) requires PAC learning. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1715–1743. SIAM, 2022.

[CS16] Deeparnab Chakrabarty and Comandur Seshadhri. An $o(n)$ monotonicity tester for boolean functions over the hypercube. *SIAM Journal on Computing*, 45(2):461–472, 2016.

[CS18] Artem Chernikov and Sergei Starchenko. A note on the Erdös-Hajnal property for stable graphs. *Proceedings of the American Mathematical Society*, 146(2):785–790, 2018.

[CST14] Xi Chen, Rocco A Servedio, and Li-Yang Tan. New algorithms and lower bounds for monotonicity testing. In *Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 286–295. IEEE, 2014.

[CWX17] Xi Chen, Erik Waingarten, and Jinyu Xie. Beyond Talagrand functions: new lower bounds for testing monotonicity and unateness. In *Proceedings of the ACM SIGACT Symposium on Theory of Computing (STOC)*, pages 523–536, 2017.

[DEG+21] Vida Dujmović, Louis Esperet, Cyril Gavoille, Gwenaël Joret, Piotr Micek, and Pat Morin. Adjacency labelling for planar graphs (and beyond). *Journal of the ACM (JACM)*, 68(6):1–33, 2021.

[DHK+10] Ilias Diakonikolas, Prahladh Harsha, Adam Klivans, Raghu Meka, Prasad Raghavendra, Rocco A Servedio, and Li-Yang Tan. Bounding the average sensitivity and noise sensitivity of polynomial threshold functions. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pages 533–542, 2010.

[DKS18] Ilias Diakonikolas, Daniel M. Kane, and Alistair Stewart. Learning geometric concepts with nasty noise. In *Proceedings of the ACM SIGACT Symposium on Theory of Computing (STOC)*, pages 1061–1073, 2018.

[DMN19] Anindya De, Elchanan Mossel, and Joe Neeman. Is your function low dimensional? In *Conference on Learning Theory (COLT)*, volume 99 of *Proceedings of Machine Learning Research*, pages 979–993. PMLR, 2019.

[DN16] Zdeněk Dvorák and Sergey Norin. Strongly sublinear separators and polynomial expansion. *SIAM Journal on Discrete Mathematics*, 30(2):1095–1101, 2016.

[DOSW11] Ilias Diakonikolas, Ryan O'Donnell, Rocco A Servedio, and Yi Wu. Hardness results for agnostically learning low-degree polynomial threshold functions. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1590–1606. SIAM, 2011.

[DRR+14] Erik D Demaine, Felix Reidl, Peter Rossmanith, Fernando Sánchez Villaamil, Somnath Sikdar, and Blair D Sullivan. Structural sparsity of complex networks: Bounded expansion in random models and real-world graphs. *arXiv preprint arXiv:1406.2587*, 2014.

[Dvo21] Zdeněk Dvorák. A note on sublinear separators and expansion. *European J. Combin.*, 93:103273, 2021.

[EHK22] Louis Esperet, Nathaniel Harms, and Andrey Kupavskii. Sketching distances in monotone graph classes. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM), to appear*, 2022.

[EHZ22] Louis Esperet, Nathaniel Harms, and Viktor Zamaraev. Optimal adjacency labels for subgraphs of cartesian products. https://arxiv.org/abs/2206.02872, 2022. In submission.

[EIX22] Talya Eden, Piotr Indyk, and Haike Xu. Embeddings and labeling schemes for A^*. In *13th Innovations in Theoretical Computer Science Conference (ITCS 2022)*, volume 215, page 62. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2022.

[ELO08] Louis Esperet, Arnaud Labourel, and Pascal Ochem. On induced-universal graphs for the class of bounded-degree graphs. *Information Processing Letters*, 108(5):255–260, 2008.

[ES35] Paul Erdös and George Szekeres. A combinatorial problem in geometry. *Compositio mathematica*, 2:463–470, 1935.

[ES20] Rogers Epstein and Sandeep Silwal. Property testing of LP-type problems. In *47th International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 168, pages 98:1–98:18. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2020.

[FGNW17] Ofer Freedman, Paweł Gawrychowski, Patrick K Nicholson, and Oren Weimann. Optimal distance labeling schemes for trees. In *Proceedings of the ACM Symposium on Principles of Distributed Computing (PODC 2017)*, pages 185–194, 2017.

[Fil20] Arnold Filtser. Scattering and sparse partitions, and their applications. In *47th International Colloquium on Automata, Languages, and Programming (ICALP)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.

[FK09] Pierre Fraigniaud and Amos Korman. On randomized representations of graphs using short labels. In *Proceedings of the Symposium on Parallelism in Algorithms and Architectures (SPAA)*. ACM Press, 2009.

[FK10] Pierre Fraigniaud and Amos Korman. Compact ancestry labeling schemes for XML trees. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 458–466. SIAM, 2010.

[FLN+02] Eldar Fischer, Eric Lehman, Ilan Newman, Sofya Raskhodnikova, Ronitt Rubinfeld, and Alex Samorodnitsky. Monotonicity testing over general poset domains. In *Proceedings of the thiry-fourth annual ACM Symposium on Theory of Computing (STOC)*, pages 474–483, 2002.

[Flo89] Sally Floyd. *Space-bounded learning and the Vapnik-Chervonenkis dimension*. PhD thesis, University of California, Berkeley, 1989.

[FR10] Shahar Fattal and Dana Ron. Approximating the distance to monotonicity in high dimensions. *ACM Transactions on Algorithms (TALG)*, 6(3):1–37, 2010.

[FT03] Jittat Fakcharoenphol and Kunal Talwar. An improved decomposition theorem for graphs excluding a fixed minor. In *Approximation, Randomization, and Combinatorial Optimization: Algorithms and Techniques*, pages 36–46. Springer, 2003.

[FW95] Sally Floyd and Manfred Warmuth. Sample compression, learnability, and the Vapnik-Chervonenkis dimension. *Machine learning*, 21(3):269–304, 1995.

[FY20] Noah Fleming and Yuichi Yoshida. Distribution-free testing of linear functions on $\mathbb{R}^n$. In *11th Innovations in Theoretical Computer Science Conference (ITCS 2020)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.

[GGR98] Oded Goldreich, Shari Goldwasser, and Dana Ron. Property testing and its connection to learning and approximation. *Journal of the ACM (JACM)*, 45(4):653–750, 1998.

[GJ22] Paweł Gawrychowski and Wojciech Janczewski. Simpler adjacency labeling for planar graphs with B-trees. In *Symposium on Simplicity in Algorithms (SOSA)*, pages 24–36. SIAM, 2022.

[GKK+01] Cyril Gavoille, Michal Katz, Nir A Katz, Christophe Paul, and David Peleg. Approximate distance labeling schemes. In *European Symposium on Algorithms (ESA)*, pages 476–487. Springer, 2001.

[GKN+20] Jakub Gajarský, Stephan Kreutzer, Jaroslav Nešetřil, Patrice Ossona de Mendez, Michał Pilipczuk, Sebastian Siebertz, and Szymon Toruńczyk. First-order interpretations of bounded expansion classes. *ACM Transactions on Computational Logic*, 21(4), jul 2020.

[GKR+18] Martin Grohe, Stephan Kreutzer, Roman Rabinovich, Sebastian Siebertz, and Konstantinos Stavropoulos. Coloring and covering nowhere dense graphs. *SIAM Journal on Discrete Mathematics*, 32(4):2467–2481, 2018.

[GKS17] Martin Grohe, Stephan Kreutzer, and Sebastian Siebertz. Deciding first-order properties of nowhere dense graphs. *Journal of the ACM (JACM)*, 64(3):1–32, 2017.

[GL07a] Cyril Gavoille and Arnaud Labourel. On local representation of distances in trees. In *Proceedings of the twenty-sixth annual ACM Symposium on Principles of Distributed Computing (PODC 2007)*, pages 352–353, 2007.

[GL07b] Cyril Gavoille and Arnaud Labourel. Shorter implicit representation for planar graphs and bounded treewidth graphs. In *European Symposium on Algorithms (ESA 2007)*, pages 582–593. Springer, 2007.

[GP03] Cyril Gavoille and David Peleg. Compact and localized distributed data structures. *Distributed Computing*, 16(2-3):111–120, September 2003.

[GP08] Cyril Gavoille and Christophe Paul. Optimal distance labeling for interval graphs and related graph families. *SIAM Journal on Discrete Mathematics*, 22(3):1239–1258, January 2008.

[GPPR04] Cyril Gavoille, David Peleg, Stéphane Pérennes, and Ran Raz. Distance labeling in graphs. *Journal of Algorithms*, 53(1):85–112, 2004.

[GPT21] Jakub Gajarskỳ, Michał Pilipczuk, and Szymon Toruńczyk. Stable graphs of bounded twin-width. *arXiv preprint arXiv:2107.03711*, 2021.

[GR16] Oded Goldreich and Dana Ron. On sample-based testers. *ACM Transactions on Computation Theory*, 8(2):1–54, 2016.

[Gra70] Ron L Graham. On primitive graphs and optimal vertex assignments. *Annals of the New York academy of sciences*, 175(1):170–186, 1970.

[GS09] Dana Glasner and Rocco A Servedio. Distribution-free testing lower bound for basic boolean functions. *Theory of Computing*, 5(1):191–216, 2009.

[GW94] Bernd Gärtner and Emo Welzl. Vapnik-Chervonenkis dimension and (pseudo-) hyperplane arrangements. *Discrete & Computational Geometry*, 12(4):399–432, 1994.

[Har14] Sariel Har-Peled. Determining the number of clusters using property testing algorithm. Theoretical Computer Science Stack Exchange, 2014, https://cstheory.stackexchange.com/q/25655. (accessed January 2021).

[Har19] Nathaniel Harms. Testing halfspaces over rotation-invariant distributions. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, pages 694–713. SIAM, 2019.

[Har20] Nathaniel Harms. Universal communication, universal graphs, and graph labeling. In *11th Innovations in Theoretical Computer Science Conference (ITCS 2020)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.

[Hau95] David Haussler. Sphere packing numbers for subsets of the Boolean n-cube with bounded Vapnik-Chervonenkis dimension. *Journal of Combinatorial Theory, Series A*, 69(2):217–232, 1995.

[HH21] Hamed Hatami and Pooya Hatami. The implicit graph conjecture is false. https://arxiv.org/abs/2111.13198, 2021.

[HHH21a] Lianna Hambardzumyan, Hamed Hatami, and Pooya Hatami. A counter-example to the probabilistic universal graph conjecture via randomized communication complexity. https://arxiv.org/abs/2111.10436, 2021.

[HHH21b] Lianna Hambardzumyan, Hamed Hatami, and Pooya Hatami. Dimension-free bounds and structural results in communication complexity. *Electronic Colloquium on Computational Complexity (ECCC)*, TR21-0666, 2021.

[HHP+22] Hamed Hatami, Pooya Hatami, William Pires, Ran Tao, and Rosie Zhao. Lower bound methods for sign-rank and their limitations. *Electronic Colloquium on Computational Complexity (ECCC)*, TR22-079, 2022.

[HK07] Shirley Halevy and Eyal Kushilevitz. Distribution-free property-testing. *SIAM Journal on Computing*, 37(4):1107–1138, 2007.

[HS07] Lisa Hellerstein and Rocco A. Servedio. On PAC learning algorithms for rich Boolean function classes. *Theoretical Computer Science*, 384(1):66–76, 2007.

[HSZZ06] Wei Huang, Yaoyun Shi, Shengyu Zhang, and Yufan Zhu. The communication complexity of the Hamming distance problem. *Information Processing Letters*, 99(4):149–153, 2006.

[HT01] Torben Hagerup and Torsten Tholey. Efficient minimal perfect hashing in nearly minimal space. In *Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 317–326. Springer, 2001.

[HWZ22] Nathaniel Harms, Sebastian Wild, and Viktor Zamaraev. Randomized communication and implicit graph representations. In *Proceedings of the ACM SIGACT Symposium on Theory of Computing (STOC)*, 2022.

[HY21] Nathaniel Harms and Yuichi Yoshida. Downsampling for testing and learning in product distributions. https://arxiv.org/abs/2007.07449, 2021.

[IM98] Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*, pages 604–613, 1998.

[Ind06] Piotr Indyk. Stable distributions, pseudorandom generators, embeddings, and data stream computation. *Journal of the ACM (JACM)*, 53(3):307–323, 2006.

[Jay09] T.S. Jayram. Problem 25: Communication complexity and metric spaces. https://sublinear.info/25, 2009.

[Joh14] Hunter R Johnson. Some new maximum VC classes. *Information Processing Letters*, 114(6):294–298, 2014.

[KKMS08] Adam T. Kalai, Adam R. Klivans, Yishay Mansour, and Rocco A Servedio. Agnostically learning halfspaces. *SIAM Journal on Computing*, 37(6):1777–1805, 2008.

[KLST20] Matthew Kwan, Shoham Letzter, Benny Sudakov, and Tuan Tran. Dense induced bipartite subgraphs in triangle-free graphs. *Combinatorica*, 40(2):283–305, 2020.

[KMS18] Subhash Khot, Dor Minzer, and Muli Safra. On monotonicity testing and boolean isoperimetric-type theorems. *SIAM Journal on Computing*, 47(6):2238–2276, 2018.

[KNR92] Sampath Kannan, Moni Naor, and Steven Rudich. Implicit representation of graphs. *SIAM Journal on Discrete Mathematics*, 5(4):596–603, 1992.

[KNR99] Ilan Kremer, Noam Nisan, and Dana Ron. On randomized one-round communication complexity. *Computational Complexity*, 8(1):21–49, 1999.

[KO04] Daniela Kühn and Deryk Osthus. Every graph of sufficiently large average degree contains a C_4-free subgraph of large average degree. *Combinatorica*, 24(1):155–162, 2004.

[KOR00] Eyal Kushilevitz, Rafail Ostrovsky, and Yuval Rabani. Efficient search for approximate nearest neighbor in high dimensional spaces. *SIAM Journal on Computing*, 30(2):457–474, 2000.

[KOS04] Adam R Klivans, Ryan O'Donnell, and Rocco A Servedio. Learning intersections and thresholds of halfspaces. *Journal of Computer and System Sciences*, 68(4):808–840, 2004.

[KOS08] Adam R Klivans, Ryan O'Donnell, and Rocco A Servedio. Learning geometric concepts via gaussian surface area. In *Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 541–550, 2008.

[KPR93] Philip Klein, Serge A Plotkin, and Satish Rao. Excluded minors, network decomposition, and multicommodity flow. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*, pages 682–690, 1993.

[KR00] Michael Kearns and Dana Ron. Testing problems with sublearning sample complexity. *Journal of Computer and System Science*, 61(3):428–456, 2000.

[KS04] Adam R Klivans and Rocco A Servedio. Learning DNF in time $2^{\tilde{o}(n^{1/3})}$. *Journal of Computer and System Sciences*, 68(2):303–318, 2004.

[KW07] Dima Kuzmin and Manfred K Warmuth. Unlabeled compression schemes for maximum classes. *Journal of Machine Learning Research*, 8(Sep):2047–2081, 2007.

[LM20] Hong Liu and Richard Montgomery. A solution to Erdős and Hajnal's odd cycle problem. *arXiv preprint arXiv:2010.15802*, 2020.

[LNP80] László Lovász, J Nešetšil, and Ales Pultr. On a product dimension of graphs. *Journal of Combinatorial Theory, Series B*, 29(1):47–67, 1980.

[LS05] Urs Lang and Thilo Schlichenmaier. Nagata dimension, quasisymmetric embeddings, and lipschitz extensions. *International Mathematics Research Notices*, 2005(58):3625–3655, 2005.

[LS09] Nati Linial and Adi Shraibman. Learning complexity vs communication complexity. *Combinatorics, Probability and Computing*, 18(1-2):227–245, 2009.

[LS10] James R Lee and Anastasios Sidiropoulos. Genus and the geometry of the cut graph. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 193–201. SIAM, 2010.

[LV08] VV Lozin and J Volz. The clique-width of bipartite graphs in monogenic classes. *International Journal of Foundations of Computer Science*, 19(2):477–494, 2008.

[LZ17] Vadim Lozin and Viktor Zamaraev. The structure and the number of P_7-free bipartite graphs. *European Journal of Combinatorics*, 65:143–153, 2017.

[Man97] Yishay Mansour. Pessimistic decision tree pruning based on tree size. In *Machine Learning-International Workshop then Conference*, pages 195–201. Citeseer, 1997.

[Mat96] Jiří Matoušek. On the distortion required for embedding finite metric spaces into normed spaces. *Israel Journal of Mathematics*, 93(1):333–344, 1996.

[Mat13] Jiří Matoušek. *Lectures on discrete geometry*, volume 212. Springer Science & Business Media, 2013.

[McC21] Rose McCarty. Dense induced subgraphs of dense bipartite graphs. *SIAM J. Discret. Math.*, 35(2):661–667, 2021.

[Meh84] Kurt Mehlhorn. *Data Structures and Algorithms 1 Sorting and Searching*. Monographs in Theoretical Computer Science. An EATCS Series, 1. Springer Berlin Heidelberg, Berlin, Heidelberg, 1st ed. 1984. edition, 1984.

[MM13] Colin McDiarmid and Tobias Müller. Integer realizations of disk and segment graphs. *Journal of Combinatorial Theory, Series B*, 103(1):114–143, 2013.

[MMA+19] R Benton Metcalf, Massimo Meneghetti, Camille Avestruz, Fabio Bellagamba, Clécio R Bom, Emmanuel Bertin, Rémi Cabanac, F Courbin, Andrew Davies, Etienne Decencière, et al. The strong gravitational lens finding challenge. *Astronomy & Astrophysics*, 625:A119, 2019.

[MNSW98] Peter Bro Miltersen, Noam Nisan, Shmuel Safra, and Avi Wigderson. On data structures and asymmetric communication complexity. *Journal of Computer and System Sciences*, 57(1):37–49, 1998.

[Mor12] Shay Moran. *Shattering Extremal Systems*. PhD thesis, Universität des Saarlandes Saarbrücken, 2012.

[MORS10] Kevin Matulef, Ryan O'Donnell, Ronitt Rubinfeld, and Rocco A Servedio. Testing halfspaces. *SIAM Journal on Computing*, 39(5):2004–2047, 2010.

[MS14] Maryanthe Malliaris and Saharon Shelah. Regularity lemmas for stable graphs. *Transactions of the American Mathematical Society*, 366(3):1551–1585, 2014.

[Mul89] John H Muller. Local structure in graph classes. 1989.

[Mun77] AG Munford. A note on the uniformity assumption in the birthday problem. *The American Statistician*, 31(3):119–119, 1977.

[MW16] Shay Moran and Manfred K Warmuth. Labeled compression schemes for extremal classes. In *International Conference on Algorithmic Learning Theory*, pages 34–49. Springer, 2016.

[N+19] Michelle Ntampaka et al. The role of machine learning in the next decade of cosmology. https://arxiv.org/abs/1902.10159, 2019.

[Nee14] Joe Neeman. Testing surface area with arbitrary accuracy. In *Symposium on Theory of Computing (STOC)*, pages 393–397. ACM, 2014.

[Nik20] Sasho Nikolov. One-way randomized communication complexity of greater-than. Theoretical Computer Science Stack Exchange, 2020, https://cstheory.stackexchange.com/q/48108. (accessed 2020-12-29).

[NK96] Noam Nisan and Eyal Kushilevitz. *Communication Complexity*. Cambridge University Press, 1996.

[NMP+21] Jaroslav Nešetřil, Patrice Ossona de Mendez, Michał Pilipczuk, Roman Rabinovich, and Sebastian Siebertz. Rankwidth meets stability. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2014–2033. SIAM, 2021.

[NO08] Jaroslav Nešetřil and Patrice Ossona de Mendez. Grad and classes with bounded expansion II. algorithmic aspects. *European Journal of Combinatorics*, 29(3):777–791, 2008.

[NO12] Jaroslav Nešetřil and Patrice Ossona de Mendez. *Sparsity: graphs, structures, and algorithms*, volume 28. Springer-Verlag, 2012.

[NO15] Jaroslav Nešetřil and Patrice Ossona de Mendez. On low tree-depth decompositions. *Graphs and combinatorics*, 31(6):1941–1963, 2015.

[NOdMS22] Jaroslav Nešetřil, Patrice Ossona de Mendez, and Sebastian Siebertz. Structural properties of the first-order transduction quasiorder. In *EACSL Conference on Computer Science Logic (CSL 2022)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2022.

[NOW12] Jaroslav Nešetřil, Patrice Ossona de Mendez, and David R Wood. Characterisations and examples of graph classes with bounded expansion. *European Journal of Combinatorics*, 33(3):350–373, 2012.

[O'D14] Ryan O'Donnell. *Analysis of Boolean Functions*. Cambridge University Press, 2014.

[OS10] Ryan O'Donnell and Rocco A Servedio. New degree bounds for polynomial threshold functions. *Combinatorica*, 30(3):327–358, 2010.

[Pel00] David Peleg. Proximity-preserving labeling schemes. *Journal of Graph Theory*, 33(3):167–176, 2000.

[Pel05] David Peleg. Informative labeling schemes for graphs. *Theoretical Computer Science*, 340(3):577–593, 2005.

[PRR06] Michal Parnas, Dana Ron, and Ronitt Rubinfeld. Tolerant property testing and distance approximation. *Journal of Computer and System Sciences*, 72(6):1012–1042, 2006.

[PSW20] Toniann Pitassi, Morgan Shirley, and Thomas Watson. Nondeterministic and randomized boolean hierarchies in communication complexity. In *47th International Colloquium on Automata, Languages, and Programming (ICALP)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.

[Rad64] Richard Rado. Universal graphs and universal functions. *Acta Arithmetica*, 4(9):331–340, 1964.

[Ras03] Sofya Raskhodnikova. Approximate testing of visual properties. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 370–381. Springer, 2003.

[Raz17] Ilya Razenshteyn. *High-dimensional similarity search and sketching: algorithms and hardness*. PhD thesis, Massachusetts Institute of Technology, 2017.

[RR20] Dana Ron and Asaf Rosin. Almost Optimal Distribution-Free Sample-Based Testing of k-Modality. In Jarosław Byrka and Raghu Meka, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2020)*, volume 176 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 27:1–27:19, Dagstuhl, Germany, 2020. Schloss Dagstuhl–Leibniz-Zentrum für Informatik.

[RRSS09] Sofya Raskhodnikova, Dana Ron, Amir Shpilka, and Adam Smith. Strong lower bounds for approximating distribution support size and the distinct elements problem. *SIAM Journal on Computing*, 39(3):813–842, 2009.

[RS15] Sivaramakrishnan Natarajan Ramamoorthy and Makrand Sinha. On the communication complexity of greater-than. In *Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 442–444. IEEE, 2015.

[RV04] Luis Rademacher and Santosh Vempala. Testing geometric convexity. In *International Conference on Foundations of Software Technology and Theoretical Computer Science*, pages 469–480. Springer, 2004.

[RY20] Anup Rao and Amir Yehudayoff. *Communication Complexity and Applications*. Cambridge University Press, 2020.

[Sağ18] Mert Sağlam. Near log-convexity of measured heat in (discrete) time and consequences. In *Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 967–978. IEEE, 2018.

[SB14] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge University Press, 2014.

[Sch99] Edward R Scheinerman. Local representations using very short labels. *Discrete mathematics*, 203(1-3):287–290, 1999.

[Sil20] Sandeep Silwal. Personal communication, 2020.

[Smi88] D. V. Smirnov. Shannon's information methods for lower bounds for probabilistic communication complexity. Master's thesis, Moscow University, 1988.

[SNHM+18] Ignacio Sevilla-Noarbe, Ben Hoyle, MJ Marchã, MT Soumagnac, K Bechtol, A Drlica-Wagner, F Abdalla, J Aleksić, C Avestruz, E Balbinot, et al. Star–galaxy classification in the Dark Energy Survey Y1 data set. *Monthly Notices of the Royal Astronomical Society*, 481(4):5451–5469, 2018.

[Spi03] Jeremy P Spinrad. *Efficient graph representations*. American Mathematical Society, 2003.

[SS02] Michael Saks and Xiaodong Sun. Space lower bounds for distance approximation in the data stream model. In *Proceedings of the thirty-fourth annual ACM Symposium on Theory of Computing (STOC)*, pages 360–369, 2002.

[Sud10] Madhu Sudan. Invariance in property testing. In *Property testing*, pages 211–227. Springer, 2010.

[SZ94] Edward R Scheinerman and Jennifer Zito. On the size of hereditary classes of graphs. *Journal of Combinatorial Theory, Series B*, 61(1):16–39, 1994.

[Tan20] Li-Yang Tan. Personal communication, 2020.

[Tho83] Carsten Thomassen. Girth in graphs. *Journal of Combinatorial Theory, Series B*, 35(2):129–141, 1983.

[Tho04] Mikkel Thorup. Compact oracles for reachability and approximate distances in planar digraphs. *Journal of the ACM*, 51(6):993–1024, 2004.

[TZ05] Mikkel Thorup and Uri Zwick. Approximate distance oracles. *Journal of the ACM*, 52(1):1–24, 2005.

[Val84] Leslie G. Valiant. A theory of the learnable. *Commun. ACM*, 27(11):1134–1142, 1984.

[vdHOQ+17] Jan van den Heuvel, Patrice Ossona de Mendez, Daniel Quiroz, Roman Rabinovich, and Sebastian Siebertz. On the generalised colouring numbers of graphs that exclude a fixed minor. *European Journal of Combinatorics*, 66:129–144, 2017.

[Vem10a] Santosh Vempala. Learning convex concepts from gaussian distributions with PCA. In *Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 124–130, 2010.

[Vem10b] Santosh Vempala. A random-sampling-based algorithm for learning intersections of halfspaces. *Journal of the ACM*, 57(6):1–14, 2010.

[Vio15] Emanuele Viola. The communication complexity of addition. *Combinatorica*, 35(6):703–747, 2015.

[VV11a] Gregory Valiant and Paul Valiant. Estimating the unseen: an $n/\log(n)$-sample estimator for entropy and support size, shown optimal via new CLTs. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*, pages 685–694, 2011.

[VV11b] Gregory Valiant and Paul Valiant. The power of linear estimators. In *Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 403–412. IEEE, 2011.

[War68] Hugh E. Warren. Lower bounds for approximation by nonlinear manifolds. *Transactions of the American Mathematical Society*, 133(1):167–178, 1968.

[WD81] Roberta S Wenocur and Richard M Dudley. Some special Vapnik-Chervonenkis classes. *Discrete Mathematics*, 33(3):313–318, 1981.

[Wen62] James G. Wendel. A problem in geometric probability. *Mathematica Scandinavica*, 11:109–112, 1962.

[WP11] Oren Weimann and David Peleg. A note on exact distance labeling. *Information processing letters*, 111(14):671–673, 2011.

[WY19] Yihong Wu and Pengkun Yang. Chebyshev polynomials, moment matching, and optimal estimation of the unseen. *The Annals of Statistics*, 47(2):857–883, 2019.

APPENDICES

Appendix A

Lower Bound on Support Size Distinction

We provide an exposition of the proof of Theorem 2.2.5 in this section. The proof that follows is a very slight adaptation of the proof of Wu & Yang [WY19], with the only changes to their proof being the ones necessary to adapt the lower bound to be on the decision problem of support size distinction (SSD) instead of the support size estimation (SSE) problem.

We begin by defining a decision problem for distributions-of-distributions over $\mathbb{N}$.

Definition A.0.1 (Meta-Distribution Decision Problem)**.** Let $\mathcal{P}, \mathcal{Q}$ be two distributions over probability distributions on $\mathbb{N}$. $\mathsf{DEC}(\mathcal{P}, \mathcal{Q})$ is the minimum number m such that there exists an algorithm A that draws a set S of m independent samples from its input distribution, and its output $A(S)$ satisfies the following:

- $\mathop{\mathbb{P}}_{p\sim\mathcal{P}, S\sim p^m}[A(S) = 1] \geq 2/3$; and,
- $\mathop{\mathbb{P}}_{q\sim\mathcal{Q}, S\sim q^m}[A(S) = 0] \geq 2/3$.

It is clear that Theorem 2.2.5 follows from the fact that, for any $0 < \alpha < \beta \leq 1$ such that $\alpha \geq \delta$ and $\beta \leq 1 - \delta$,

$$\mathsf{SSD}(n, \alpha, \beta) \geq \sup_{\mathcal{P},\mathcal{Q}} \mathsf{DEC}(\mathcal{P}, \mathcal{Q}),$$

where the supremum is taken over all distributions $\mathcal{P}, \mathcal{Q}$ over distributions on $[n]$ such that any $p \in \operatorname{supp}(\mathcal{P})$ has $|\operatorname{supp}(p)| \le \delta n \le \alpha n$, any $q \in \operatorname{supp}(\mathcal{Q})$ has $|\operatorname{supp}(q)| \ge (1-\delta)n \ge \beta n$, and any $p \in \operatorname{supp}(\mathcal{P}) \cup \operatorname{supp}(\mathcal{Q})$ has $p_i \ge 1/n$ for each $i \in \operatorname{supp}(p)$. Therefore, to establish Theorem 2.2.5, it suffices to prove the following theorem.

Theorem A.0.2 ([WY19])**.** *There is a constant $C > 0$ such that, for every $n \in \mathbb{N}$ and every $C\frac{\sqrt{\log n}}{n^{1/4}} < \delta < \frac{1}{2}$, there exist distributions $\mathcal{P}, \mathcal{Q}$ over the space of probability distributions on $[n]$, such that $\mathsf{DEC}(\mathcal{P}, \mathcal{Q}) = \Omega\left(\frac{n}{\log n} \log^2 \frac{1}{1-\delta}\right)$, and where:*

- *Every $p \in \mathcal{P}$ has support size at most δn;*
- *Every $p \in \mathcal{Q}$ has support size at least $(1-\delta)n$;*
- *Every $p \in \mathcal{P} \cup \mathcal{Q}$ has $p(x) \ge 1/n$ for all $x \in \operatorname{supp}(p)$.*

What follows is adapted from the proofs of Wu & Yang [WY19].

Definition A.0.3. For any $\nu \ge 0$, let $\mathcal{D}_n(\nu)$ be the set of vectors $p \in \mathbb{R}^n$ such that each p_i satisfies $p_i \in \{0\} \cup \left[\frac{1+\nu}{n}, 1\right]$, and $|1 - \sum_i p_i| \le \nu$. For $p \in \mathcal{D}_n(\nu)$, we will write $\operatorname{supp}(p) = \{i \in [n] : p(i) > 0\}$. Note that $\mathcal{D}_n(0)$ is the set of probability distributions over $[n]$ with densities at least $1/n$ on the support.

Definition A.0.4 (Poisson Sampling Model)**.** Let $\mathcal{P}, \mathcal{Q}$ be distributions over $\mathcal{D}_n(\nu)$. Define $\widetilde{\mathsf{DEC}}(\mathcal{P}, \mathcal{Q})$ as the smallest number m such that there is an algorithm A that does the following on input $p \in \mathcal{D}_n(\nu)$: For each $i \in \mathbb{N}$, A receives a vector $s : \mathbb{N} \to \mathbb{N}$ such that $s(i) \sim \mathsf{Poi}(mp(i))$. $A(s)$ outputs 0 or 1, and satisfies:

- $\mathop{\mathbb{P}}_{p\sim\mathcal{P},s}[A(s) = 1] \ge 2/3$;
- $\mathop{\mathbb{P}}_{q\sim\mathcal{Q},s}[A(s) = 0] \ge 2/3$.

Lemma A.0.5. *For any n, ν, suppose that $\mathcal{P}, \mathcal{Q}$ are distributions over $\mathcal{D}_n(\nu)$. Then for distributions $\mathcal{P}', \mathcal{Q}'$ over $\mathcal{D}_n(0)$ defined by choosing $p \sim \mathcal{P}$ and taking $p/\sum_i p(i)$, or by choosing $q \sim \mathcal{Q}$ and taking $q/\sum_i q(i)$, respectively,*

$$\mathsf{DEC}(\mathcal{P}', \mathcal{Q}') \ge \Omega((1-\nu) \cdot \widetilde{\mathsf{DEC}}(\mathcal{P}, \mathcal{Q}))\,.$$

Proof. First we observe that for any t, conditioned on the event $\sum_i s_i = t$, the vector s has the same distribution as the vector s' obtained by drawing t points $i \in \mathbb{N}$ independently from $p/\sum_i p(i)$ and letting s'_i be the number of times item i is observed.

For any k, let A_k be the algorithm that, receiving k independent samples from the input distribution, has the highest probability of correctly distinguishing $\mathcal{P}'$ from $\mathcal{Q}'$. Let m be the minimum number such that A_m has success probability at least $9/10$, so that

$$\mathop{\mathbb{P}}_{p\sim\mathcal{P}',s'}[A_m(s')=1] \geq 9/10 \qquad \mathop{\mathbb{P}}_{q\sim\mathcal{Q}',s'}[A_m(s')=0] \geq 9/10\,.$$

Observe that (by standard boosting techniques), $m = \Theta(\mathsf{DEC}(\mathcal{P}',\mathcal{Q}'))$. For some $m' = \rho m$ (with $\rho > 1$ to be chosen later), we construct an algorithm in the Poisson testing model where $s_i \sim \mathsf{Poi}(m'p(i))$ and upon receiving a vector s with $\sum_i s_i = t$, runs $A_t(s)$.

$$\begin{aligned}
\mathop{\mathbb{P}}_{p\sim\mathcal{P},s}[A(s)=0] &= \sum_{k=0}^{\infty} \mathbb{P}[t=k] \mathop{\mathbb{P}}_{p,s}\left[A_k(s)=0 \;\middle|\; \sum_i s_i = k\right] \\
&= \sum_{k=0}^{\infty} \mathbb{P}[t=k] \mathop{\mathbb{P}}_{p',s'}\left[A_k(s')=0 \;\middle|\; \sum_i s_i = k\right] \\
&\leq \frac{1}{10}\mathbb{P}[t\geq m] + \mathbb{P}[t<m] = \frac{1}{10} + \frac{9}{10}\mathbb{P}[t<m]\,.
\end{aligned}$$

The same argument shows that for $q \sim \mathcal{Q}$,

$$\mathop{\mathbb{P}}_{q\sim\mathcal{Q},s}[A(S)=1] \leq \frac{1}{10} + \frac{9}{10}\mathbb{P}[t<m]\,,$$

so what remains is to bound m'. t is a sum of independent Poisson random variables $\mathsf{Poi}(m'p_i)$, so $t \sim \mathsf{Poi}(m'\sum_i p(i))$, which has mean $m'\sum_i p(i) \geq m'(1-\nu) = (1-\nu)\rho m$. For $X \sim \mathsf{Poi}(\lambda)$ and $z < \lambda$ we use the inequality:

$$\mathbb{P}[X<z] \leq \frac{(e\lambda)^z e^{-\lambda}}{z^z}\,,$$

which implies

$$\begin{aligned}
\mathbb{P}[t<m] &\leq \frac{(e(1-\nu)\rho m)^m e^{-m\rho(1-\nu)}}{m^m} \\
&= (e(1-\nu)\rho)^m e^{-m\rho(1-\nu)} = \exp\left(m(\ln(e(1-\nu)\rho) - \rho(1-\nu))\right)\,.
\end{aligned}$$

For any constant $C > 0$ there is C' such that for $\rho > C'/(1-\nu)$, this probability is at most $\exp m(1 + \ln(C') - C') = \exp -Cm$, so we can choose $\exp -C < 1/100$ to obtain a total failure probability of at most $1/10 + 9/100 < 1/3$, with $m' = \rho m = O(m/(1-\nu))$. Thus

$$\widetilde{\mathsf{DEC}}(\mathcal{P}, \mathcal{Q}) \le m' = \frac{m}{1-\nu} = O\left(\frac{1}{1-\nu} \cdot \mathsf{DEC}(\mathcal{P}', \mathcal{Q}')\right). \qquad \square$$

Lemma A.0.6. *Let $\nu, \lambda > 0$, and Suppose P, Q are random variables taking values in $\{0\} \cup [1+\nu, \lambda]$, such that $\mathbb{E}[P] = \mathbb{E}[Q] = 1, \mathbb{E}[P^j] = \mathbb{E}[Q^j]$ for all $j \in [L]$, and $|\mathbb{P}[P > 0] - \mathbb{P}[Q > 0]| = \delta$. Then for any $\alpha < 1/2$, if*

$$\frac{2\lambda}{n\nu^2} + \frac{2}{n\alpha^2\delta^2} + n\left(\frac{em\lambda}{2nL}\right)^L < 1/3,$$

then there exist distributions $\mathcal{P}, \mathcal{Q}$ over $\mathcal{D}_n(\nu)$ such that $\widetilde{\mathsf{DEC}}(\mathcal{P}, \mathcal{Q}) \ge m$, and for each $p \in \operatorname{supp}(\mathcal{P}), q \in \operatorname{supp}(\mathcal{Q})$, $\#\operatorname{supp}(q) - \#\operatorname{supp}(p) \ge (1-2\alpha)\delta n$ and $p(x), q(x) > (1+\nu)/n$ for each $x \in \operatorname{supp}(p), \operatorname{supp}(q)$ respectively.

Proof. Let $\mathcal{P}'$ be the distribution over vectors $\mathbb{R}^n$ obtained by drawing $p \sim \frac{1}{n}(P_1, \dots, P_n)$ where each P_i is an independent copy of P, and let $\mathcal{Q}'$ be the distribution obtained by drawing $q \sim \frac{1}{n}(Q_1, \dots, Q_n)$ in the same way. Let $\rho = \mathbb{P}[P > 0], \gamma = \mathbb{P}[Q > 0]$. Write S for the set of vectors p such that $|1 - \sum_i p_i| \le \nu$ and $|\#\operatorname{supp}(p) - n\rho| < \alpha\delta n$, and write T for the set of vectors q such that $|1 - \sum_i q_i| \le \nu$ and $|\#\operatorname{supp}(q) - n\gamma| \le \alpha\delta n$.

We will define $\mathcal{P}$ to be the distribution $\mathcal{P}'$ conditioned on the event S, while $\mathcal{Q}$ is the distribution $\mathcal{Q}'$ conditioned on T. Wu & Yang [WY19] show that these events occur with high probability (in particular, $\mathcal{P}, \mathcal{Q}$ are well-defined). It is clear that for each $p \in S, q \in T$, we will have

$$\#\operatorname{supp}(p) - \#\operatorname{supp}(q) \ge n\rho - n\gamma - 2\alpha\delta n = n\delta - 2\alpha\delta n = (1-2\alpha)\delta n,$$

as desired, and $p(x), q(x) \ge (1+\nu)/n$ for all $x \in \operatorname{supp}(p), \operatorname{supp}(q)$ respectively. So it remains to show the bound on $\widetilde{\mathsf{DEC}}(\mathcal{P}, \mathcal{Q})$.

Let the random variable $s(\mathcal{P})$ be the vector of values seen from a random $p \sim \mathcal{P}$ by a Poisson sampling algorithm with parameter m, i.e. for $s = s(\mathcal{P})$ and $p \sim \mathcal{P}, s_i \sim \mathsf{Poi}(mp_i)$. Wu & Yang [WY19] prove that

$$\|s(\mathcal{P}) - s(\mathcal{Q})\|_{\mathsf{TV}} \le \frac{2\lambda}{n\nu^2} + \frac{2}{n\alpha^2\delta^2} + n\left(\frac{em\lambda}{2nL}\right)^L,$$

which by assumption is less than $1/3$. Therefore, if a Poisson sampling algorithm A outputs 1 with probability at least $2/3$ over the random variable $s(\mathcal{P})$, it will output 1 with probability greater than $2/3 - 1/3 = 1/3$. Therefore $\widetilde{\mathsf{DEC}}(\mathcal{P}, \mathcal{Q}) \geq m$ as desired. □

Lemma A.0.7 ([WY19], Lemma 7)**.** *For any $L \in \mathbb{N}$, $\nu > 0, \lambda > 1 + \nu$, there exist random variables P, Q such that:*

1. *P, Q are supported on $\{0\} \cup [1 + \nu, \lambda]$;*
2. *$\mathbb{E}[P] = \mathbb{E}[Q] = 1$ and $\forall j \in [L], \mathbb{E}[P^j] = \mathbb{E}[Q^j]$; and,*
3. *For $t = \sqrt{\frac{1+\nu}{\lambda}}$,*

$$\mathbb{P}[P > 0] - \mathbb{P}[Q > 0] = \frac{(1+t)^2}{1+\nu} \cdot \left(1 - \frac{2t}{1+t}\right)^L .$$

Proof of Theorem A.0.2. For any parameters $L \in \mathbb{N}, \nu > 0, \lambda > 1 + \nu$, we obtain from Lemma A.0.7 random variables P, Q taking values in $\{0\} \cup [1 + \nu, \lambda]$ such that $\mathbb{E}[P] = \mathbb{E}[Q] = 1, \mathbb{E}[P^j] = \mathbb{E}[Q^j]$ for all $j \in [L]$, and

$$\mathbb{P}[P > 0] - \mathbb{P}[Q > 0] = \frac{(1+t)^2}{1+\nu} \cdot \left(1 - \frac{2t}{1+t}\right)^L =: \epsilon ,$$

where $t = \sqrt{\frac{1+\nu}{\lambda}}$. Then Lemma A.0.6 implies that for any $m, n \in \mathbb{N}$ and $\alpha > 0$, we get distributions $\mathcal{P}, \mathcal{Q}$ over $\mathcal{D}_n(\nu)$ such that $\widetilde{\mathsf{DEC}}(\mathcal{P}, \mathcal{Q}) \geq m$ and $\#\operatorname{supp}(p) - \#\operatorname{supp}(q) \geq (1 - 2\alpha)\epsilon n$ for all $p \in \operatorname{supp}(\mathcal{P}), q \in \operatorname{supp}(\mathcal{Q})$, as long as

$$\frac{2\lambda}{n\nu^2} + \frac{2}{n\alpha^2\epsilon^2} + n\left(\frac{em\lambda}{2nL}\right)^L < 1/3 . \tag{A.1}$$

Suppose that $(1 - 2\alpha)\epsilon = 1 - \delta$. Then for all $p \in \operatorname{supp}(\mathcal{P})$ we will have $\#\operatorname{supp}(p) \geq (1 - 2\alpha)\epsilon n = (1 - \delta)n$, as desired, and for all $q \in \operatorname{supp}(\mathcal{Q})$ we will have $\#\operatorname{supp}(q) \leq n - (1 - 2\alpha)\epsilon n = \delta n$. For any p we also have $p_i \geq (1 + \nu)/n$ so the normalized distribution $\mathcal{P}'$ defined in Lemma A.0.5 will have densities $p_i \geq \frac{1+\nu}{n\sum_i p_i} \geq \frac{1+\nu}{n(1+\nu)} = 1/n$, and the same for $\mathcal{Q}'$. Then for any $\nu = o(1)$, we will obtain a lower bound of

$$\mathsf{DEC}(\mathcal{P}', \mathcal{Q}') = \Omega(\widetilde{\mathsf{DEC}}(\mathcal{P}', \mathcal{Q}')) = \Omega(m) .$$

Therefore, what remains is to prove Equation (A.1) with parameter $\epsilon = \frac{1-\delta}{1-2\alpha}$.

Wu & Yang ([WY19], equation 34) show that for sufficiently large constant C, if $\delta = 1 - (1-2\alpha)\epsilon > C\frac{\sqrt{\log n}}{n^{1/4}}$ and $(1-2\alpha)\epsilon \geq n^{-o(1)}$ (where the latter holds trivially in our case because $(1-2\alpha)\epsilon \geq 1 - \delta > 1/2$), then there are parameters such that $\nu = o(1)$ and Equation (A.1) holds with

$$m = \Omega\left(\frac{n}{\log n}\log^2\frac{1}{(1-2\alpha)\epsilon}\right) = \Omega\left(\frac{n}{\log n}\log^2\frac{1}{1-\delta}\right),$$

which proves the theorem. □

Appendix B

Bibliographic Remark on Greater-Than

Recall the lower bound for Greater-Than:

Theorem B.0.1. *Any public-coin randomized SMP communication protocol for* Greater-Than *on domain* $[n]$ *requires* $\Omega(\log n)$ *bits of communication.*

Lower bounds for the Greater-Than problem in various models appear in [KNR99, MNSW98, Vio15, RS15, ATYY17]. The above theorem is stated in [KNR99] and [MNSW98]; in the latter it is also credited to [Smi88]. In [KNR99] the theorem is stated for one-way *private-coin* communication; the result for public-coin SMP communication follows from the fact that public-coin protocols for problems with domain size n can save at most $O(\log \log n)$ bits of communication over the private-coin protocol due to Newman's theorem.

However, as noted in a CSTheory StackExchange question of Sasho Nikolov [Nik20], the complete proof is not provided in either of [KNR99, MNSW98]. The same lower bound for *quantum* communication complexity is proved in [ATYY17], which implies the above result. A direct proof for classical communication complexity was suggested as an answer to [Nik20] by Amit Chakrabarti [Cha20]; we state this direct proof here for completeness and we thank Eric Blais for communicating this reference to us. We require the Augmented-Index communication problem and its lower bound from [MNSW98].

Definition B.0.2 (Augmented-Index)**.** In the Augmented-Index communication problem, Alice receives input $x \in \{0,1\}^k$ and Bob receives an integer $i \in [k]$ along with the values x_j for all $j > i$. Bob should output the value x_i.

Theorem B.0.3 ([MNSW98])**.** *Any public-coin randomized one-way communication protocol for* Augmented-Index *requires* $\Omega(k)$ *bits of communication.*

Proof of Theorem B.0.1, [Cha20]. Given inputs $x \in \{0,1\}^k$ and $i \in [k]$ to the Augmented Index problem, Bob constructs the string $y \in \{0,1\}^k$ where $y_j = x_j$ for all $j > i$ and $y_i = 0$, and $y_j = 1$ for all $j < i$. Consider the numbers $a, b \in [2^k]$ where the binary representation of a is x, with bit k being the most significant and bit 1 the least significant, and the binary representation of b is y, with the bits in the same order. If $x_i = 1$, then since $y_i = 0$ and $y_j = x_j$ for $j > i$, it holds that $b < a$. If $x_i = 0$, then since $y_j = x_j$ for $j \geq i$ and $y_j = 1$ for $j < i$ it holds that $b \geq a$. Therefore, computing Greater-Than on inputs a, b will solve Augmented Index. By Theorem B.0.3, the communication cost of Greater-Than for $n = 2^k$ is at least $\Omega(k) = \Omega(\log n)$. □