

Discovering Patterns from Sequences with Applications to Protein-Protein and Protein-DNA Interaction

by

Ho Yin Sze-To

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2018

© Ho Yin Sze-To 2018

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	Luis Rueda Professor
Supervisor(s)	Andrew K.C. Wong Distinguished Professor Emeritus Daniel Stashuk Professor
Internal Member(s)	Paul Fieguth Professor Department Chair Alexander Wong Associate Professor, P.Eng. Canada Research Chair in Medical Imaging Systems
Internal-external Member	Bin Ma Professor University Research Chair

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

Content from 5 papers are used in this thesis. I was the co-author with major contributions on designing the methods, implementation and writing the papers:

- **Sze-To, A.**, & Wong, A. K. (2017). Pattern-Directed Aligned Pattern Clustering. In Bioinformatics and Biomedicine (BIBM), 2017 IEEE International Conference on. IEEE. Acceptance Rate: 19%. **Won IEEE TCCLS Student Award**. Incorporated in Chapter 3 of this thesis. [126]
- **Sze-To, A.**, & Wong, A. K. (2018). Discovering Patterns from Sequences Using Pattern-Directed Aligned Pattern Clustering. NanoBioScience, IEEE Transactions on. Incorporated in Chapter 3 of this thesis. [127]
- Lee, E. S. A., **Sze-To, H. Y. A.**, Wong, M. H., Leung, K. S., Lau, T. C. K., & Wong, A. K. (2017). Discovering protein-dna binding cores by aligned pattern clustering. Computational biology and bioinformatics, IEEE/ACM transactions on, 14(2), 254-263. Incorporated in Chapter 4 of this thesis. [75]
- **Sze-To, A.**, Fung, S., Lee, E. S. A., & Wong, A. K. (2015, November). Predicting Protein-protein interaction using co-occurring Aligned Pattern Clusters. In Bioinformatics and Biomedicine (BIBM), 2015 IEEE International Conference on (pp. 55-60). IEEE. Acceptance Rate: 19%. Incorporated in Chapter 5 of this thesis. [124]
- **Sze-To, A.**, Fung, S., Lee, E. S. A., & Wong, A. K. (2016). Prediction of Protein-Protein Interaction via Co-Occurring Aligned Pattern Clusters. Methods. Incorporated in Chapter 5 of this thesis. [125]

Abstract

Understanding Protein-Protein and Protein-DNA interaction is of fundamental importance in deciphering gene regulation and other biological processes in living cells. Traditionally, new interaction knowledge is discovered through biochemical experiments that are often labor-intensive, expensive and time-consuming. Thus, computational approaches are preferred. Due to the abundance of sequence data available today, sequence-based interaction analysis becomes one of the most readily applicable and cost-effective methods.

One important problem in sequence-based analysis is to identify the functional regions from a set of sequences within the same family or demonstrating similar biological functions in experiments. The rationale is that throughout evolution the functional regions normally remain conserved (intact), allowing them to be identified as patterns from a set of sequences. However, there are also mutations such as substitution, insertion, deletion in these functional regions. Existing methods, such as those based on position weight matrices, assume that the functional regions have a fixed width and thus cannot identify functional regions with mutations, particularly those with insertion or deletion mutations. Recently, Aligned Pattern Clustering (APCn) was introduced to identify functional regions as Aligned Pattern Clusters (APCs) by grouping and aligning patterns with variable width. Nevertheless, APCn cannot discover functional regions with substitution, insertion and/or deletion mutations, since their frequencies of occurrences are too low to be considered as patterns.

To overcome such an impasse, this thesis proposes a new APC discovery algorithm known as Pattern-Directed Aligned Pattern Clustering (PD-APCn). By first discovering seed patterns from the input sequence data, with their sequence positions located and recorded on an address table, PD-APCn can use the seed patterns to direct the incremental extension of functional regions with minor mutations. By grouping the aligned extended patterns, PD-APCn can recruit patterns adaptively and efficiently with variable width without relying on exhaustive optimal search. Experiments on synthetic datasets with different sizes and noise levels showed that PD-APCn can identify the implanted pattern with mutations, outperforming the popular existing motif-finding software MEME with much higher recall and Fmeasure over a computational speed-up of up to 665 times. When applying PD-APCn on datasets from Cytochrome C and Ubiquitin protein families, all key binding sites conserved in the families were captured in the APC outputs.

In sequence-based interaction analysis, there is also a lack of a model for co-occurring functional regions with mutations, where co-occurring functional regions between interaction sequences are indicative of binding sites. This thesis proposes a new representation

model Co-Occurrence APCs to capture co-occurring functional regions with mutations from interaction sequences in database transaction format. Applications on Protein-DNA and Protein-Protein interaction validated the capability of Co-Occurrence APCs.

In Protein-DNA interaction, a new representation model, Protein-DNA Co-Occurrence APC, was developed for modeling Protein-DNA binding cores. The new model is more compact than the traditional one-to-one pattern associations, as it packs many-to-many associations in one model, yet it is detailed enough to allow site-specific variants. An algorithm, based on Co-Support Score, was also developed to discover Protein-DNA Co-Occurrence APCs from Protein-DNA interaction sequences. This algorithm is 1600x faster in run-time than its contemporaries. New Protein-DNA binding cores indicated by Protein-DNA Co-Occurrence APCs were also discovered via homology modeling as a proof-of-concept. In Protein-Protein interaction, a new representation model, Protein-Protein Co-Occurrence APC, was developed for modeling the co-occurring sequence patterns in Protein-Protein Interaction between two protein sequences. A new algorithm, WeMine-P2P, was developed for sequence-based Protein-Protein Interaction machine learning prediction by constructing feature vectors leveraging Protein-Protein Co-Occurrence APCs, based on novel scores such as Match Score, MaxMatch Score and APC-PPI score. Through 40 independent experiments, it outperformed the well-known algorithm, PIPE2, which also uses co-occurring functional regions while not allowing variable widths and mutations. Both applications on Protein-Protein and Protein-DNA interaction have indicated the potential use of Co-Occurrence APC for exploring other types of biosequence interaction in the future.

Acknowledgements

I would like to thank all the important people who made this thesis possible. Firstly, I would like to express my gratitude to my supervisors, Professor Andrew K. C. Wong and Professor Daniel Stashuk who guide and support me in my research. Professor Wong, thank you very much for providing me with opportunities in studying in University of Waterloo as an international student. I would not be able to study here without your support and assistance. Thank you for your patience and understanding. Your advice helps me to improve, and guides me throughout the entire Ph.D. process. Professor Daniel Stashuk, thank you very much for your consistent encouragement. Your words always boost my confidence, particularly when I need to face difficulties. I also appreciate your help in providing me with high-end computational equipment to run experiments.

Secondly, I would like to thank my Ph.D. committee members Professor Paul Fieguth, from Department of Systems Design Engineering, Prof. Alexander Wong, from Department of Systems Design Engineering, Professor Bin Ma, from Cheriton School of Computer Science, for their time and commitment contributed in reviewing this thesis. I would also like to thank Professor Luis Rueda, from School of Computer Science in University of Windsor, for accepting reviewing this thesis, and sparing his vacation time to participate in my PhD defense.

Thirdly, I would like to thank all my colleague whom I have worked and studied with at University of Waterloo, especially Dr. En-Shiun Annie Lee and Sanderz Fung for their outstanding collaborative work, as well as Dr. Dennis Zhuang, for his support in providing me with the relevant toolboxes. I would also like to specially thank Shawn Zhexuan Wang, who supported me tirelessly in my Ph.D study.

Last but not least, I would like to thank Professor Fakhri Karray, the director of Centre for Pattern Analysis and Machine Intelligence, who granted me access to the laboratory, allowing me to conduct research in the laboratory.

Dedication

To my beloved parents,
My beloved younger sister,
My mentors and supporters throughout my life.

Table of Contents

Author's Declaration	iii
Abstract	v
Acknowledgements	vii
Dedication	viii
List of Tables	xiii
List of Figures	xv
1 Introduction	1
1.1 Problem Definition	3
1.2 Challenges and Objectives	5
1.3 Contributions	11
1.4 Thesis structure	14
2 Background and Related Work	15
2.1 Protein, DNA and the Central Dogma	15
2.2 Functional Regions in Protein and DNA	16
2.2.1 Multiple Sequence Alignment (MSA)	17

2.2.2	Motif Discovery	17
2.2.3	Aligned Pattern Clustering (APCn)	17
2.3	Protein-DNA interaction	20
2.3.1	Protein-DNA binding core	22
2.3.2	Experimental approaches on studying Protein-DNA interaction	22
2.3.3	Computational approaches on studying Protein-DNA interaction	24
2.4	Protein-Protein interaction	25
2.4.1	Experimental methods on predicting Protein-Protein interaction	25
2.4.2	Computational methods on predicting Protein-Protein interaction	26
2.4.3	Sequence-based Protein-Protein interaction Prediction	26
2.4.4	Short Linear Motifs mediating Protein-Protein interaction	28
3	Pattern-Directed Aligned Pattern Clustering	29
3.1	Introduction	29
3.2	Method	30
3.2.1	Problem Definition	34
3.2.2	Input Sequence Data	34
3.2.3	Step 1: Seed Pattern Discovery	34
3.2.4	Step 2: Seed Pattern Extension	36
3.2.5	APC Growing	38
3.3	Experiments and Results	39
3.3.1	Design of Experiments	39
3.3.2	Synthetic Dataset Preparation	40
3.3.3	Evaluation of Experiments on Synthetic Datasets	40
3.3.4	Experimental Results Analysis on Dataset 1 (500 sequences)	42
3.3.5	Experimental Results Analysis on Dataset 2 (1000 sequences)	43
3.3.6	Experimental Results Analysis on Dataset 3 (2000 sequences)	45
3.3.7	Combined Analysis on Datasets 1, 2 and 3	46

3.3.8	Real Dataset Preparation and Parameter Setting	49
3.3.9	Experimental Results Analysis on Dataset Cytochrome C	49
3.3.10	Experimental Results Analysis on Dataset Ubiquitin	49
3.4	Summary	52
4	Discovering Binding Cores from Protein-DNA interaction sequences using Protein-DNA Co-Occurrence APC	56
4.1	Introduction	56
4.2	Method	57
4.2.1	Problem Definition	57
4.2.2	Biological Database	57
4.2.3	APC Discovery	57
4.2.4	Protein-DNA Co-Occurring APCs	60
4.2.5	Verification	61
4.3	Experiments and Results	63
4.3.1	Materials	63
4.3.2	Experimental Procedure	64
4.3.3	Comparative Schemes	66
4.3.4	Top 10 Protein-DNA Co-Occurring APCs	66
4.4	Discussion	67
4.4.1	Performance Comparison	67
4.4.2	Run-time Comparison	68
4.4.3	Homology Modeling	70
4.5	Summary	70
5	Predicting Protein-Protein Interaction Using Protein-Protein Co-occurrence APC	75
5.1	Introduction	75
5.2	Methods	76

5.3	Experiments and Results	82
5.3.1	Materials	82
5.3.2	Experimental design and parameter setting	83
5.3.3	Investigating the number of trees in the Random Forest	83
5.3.4	Investigating the effectiveness of APC-PPI	83
5.3.5	Comparison to PIPE2	85
5.3.6	Comparison to SVM-based Methods	85
5.3.7	Analysis of the features with high <i>hscore</i>	86
5.4	Summary	90
6	Conclusion and Future Work	91
6.1	Contributions and Novelty	91
6.2	Limitations and Future Work	92
6.2.1	Comprehensive analysis of the parameter setting of breakpoint gap and seed width in PD-APCn	92
6.2.2	Discovering Protein-DNA Binding Cores from a new Protein-DNA interaction sequence database	93
6.2.3	Improving the Prediction Performance of WeMine-P2P	93
6.2.4	Extending the representation of protein in other bioinformatics applications	93
6.3	Conclusion	95
	References	96

List of Tables

3.1	Performance evaluation of PD-APCn on Dataset 1 (500 sequences)	42
3.2	Parameter investigation of PD-APCn on Dataset 1 (500 sequences)	42
3.3	Performance evaluation of PD-APCn on Dataset 2 (1000 sequences)	44
3.4	Parameter investigation of PD-APCn on Dataset 2 (1000 sequences)	44
3.5	Performance evaluation of PD-APCn on Dataset 3 (2000 sequences)	45
3.6	Parameter investigation of PD-APCn on Dataset 3 (2000 sequences)	46
3.7	Runtime comparison of PD-APCn on Datasets 1, 2 and 3	47
4.1	A simplified example of TRANSFAC database on Protein-DNA interaction sequences	59
4.2	A summary of notations with examples on Protein-DNA interaction sequences	73
4.3	Runtime comparison between WeMine, USM-Nor and USM-Sum	74
5.1	Performance comparison of WeMine-P2P with different trees on the average Area Under Curve (AUC) among 40 independent datasets \pm the standard deviation	86
5.2	Performance comparison of WeMine-P2P with APC-PPI and Random-APC-PPI on the average Area Under Curve (AUC) among 40 independent datasets \pm the standard deviation	86
5.3	Performance comparison of PIPE2 and WeMine-P2P on the average Area Under Curve (AUC) among 40 independent datasets \pm the standard deviation	86
5.4	Performance comparison of SVM-based methods and WeMine-P2P on the average Area Under Curve (AUC) among 40 independent datasets \pm the standard deviation	87

5.5	Top 10 cAPC pairs in <i>hscore</i>	88
5.6	APCs in the top 10 cAPC pairs	89

List of Figures

1.1	An illustration of a functional region in a sequence	2
1.2	Problem 1: Identification and alignment of functional regions with mutations	4
1.3	Problem 2: Identification of co-occurrence functional regions with mutations between interaction sequences	6
1.4	An illustration of Protein-DNA interaction sequences from TRANSFAC . .	7
1.5	Problem 2b: Identification of co-occurrence functional regions with mutations between interaction sequences with interaction labels	8
1.6	An illustration of a sequence set with mutations and demonstration of outputs by existing software	10
1.7	An illustration of Protein-DNA Co-Occurrence Aligned Pattern Cluster .	11
1.8	An illustration of Protein-Protein Co-Occurrence Aligned Pattern Cluster .	12
2.1	An illustration of Central Dogma of Molecular Biology	21
2.2	An illustration of Protein-DNA (TF-TFBS) Binding Core	23
3.1	An overview of Pattern-Directed Aligned Pattern Clustering (PD-APCn) algorithm	31
3.2	An illustration of the concept of pattern breakpoint	32
3.3	An illustration of the extension of seed patterns	33
3.4	An illustration of results obtained from MEME and PD-APCn	35
3.5	An illustration of the definition of True Positive (TP), False Positive (FP) and False Negative (FN) for quantitative evaluation of the predicted conserved regions	41

3.6	A comparison of the outputs by MEME and PD-APCn on Dataset 3 (2000 sequences)	48
3.7	An illustration of the APCs outputted by PD-APCn on the Dataset Cytochrome C	50
3.8	A three-dimensional (3D) structure of Cytochrome C obtained from Protein Data Bank (PDB) (ID: 1HRC)	51
3.9	An illustration of the APCs outputted by PD-APCn on the Dataset Ubiquitin	53
3.10	A three-dimensional (3D) structure of Ubiquitin obtained from Protein Data Bank (PDB) (ID: 1AAR)	54
4.1	An overview of Protein-DNA (TF-TFBS) Binding Core discovery process .	58
4.2	An illustration of the Top 10 Protein-DNA Co-Occurring APCs (DNA variation = 1)	65
4.3	An illustration of the performance comparison between WeMine, USM and Random on Extended PDB verification	68
4.4	An illustration of how Protein-DNA Co-Occurrence APCs enabling Homology Modeling	69
4.5	A homology modeling of 1CQT with a mutation of T222A on Chain N to model the Protein-DNA Binding (WFCNRRQ, TTAATTG)	71
5.1	An illustration of a Protein-Protein Interaction Predictor WeMine-P2P . .	77
5.2	An example on how Match Score is calculated for a sequence segment . . .	79
5.3	A simplified Protein-Protein Interaction sequence dataset example with a training set and a testing set with three distinct classes	84
6.1	An illustration of Protein Binding Microarray (PBM) data	94
6.2	An illustration of encoding a protein sequence into a feature vector via APC and MaxMatchScore	95

Chapter 1

Introduction

Protein and DNA play vital roles in our human body [96]. A protein is made up of a chain of amino acids, i.e. represented by a string of alphabets, where an amino acid is denoted by an English alphabet [98]. Protein is important as it regulates biological processes and functions for virtually every biochemical reaction in living cells [98]. It carries out its function via a functional region. For example, as shown in Fig. 1.1(a), MGDVEKGKKI-FIMKCSQCHGGTVEKGGKHK is a protein sequence of Cytochrome C, where the highlighted region is the binding site binding to the heme molecule [13, 154]. On the other hand, DNA is made up of a chain of nucleotides, where a nucleotide is also represented by an English alphabet [98]. DNA is important as it does not only encode the genetic information of organisms as a living archive of instructions to accomplish the functions of life [98], but also encode the genetic switches (e.g. transcription factor binding sites (TFBSs) [133]) that regulate the expression of such information. It also carries out its function via a functional region. For example, as shown in Fig. 1.1(b), ACTTTATTTGCAATAGAAAATC is a DNA sequence taken from the TRANSFAC database [89], where the highlighted region encompasses a TFBS, verified by Protein Data Bank (PDB) [12] record (ID: 1CQT). Due to evolutionary pressure, these functional regions normally remain conserved [66, 67, 68]. However, over many generations, there are still possibly mutations such as insertion, substitution and deletion [66, 67, 68] within these functionally conserved regions.

For protein and DNA to actually exert their biological functions, they have to participate in biological mechanisms, in which they interact with others via their functional regions. Protein-Protein Interaction (PPI) [40] is one major mechanism, allowing a protein to function via interacting with another protein. PPI is important for various biological mechanisms such as metabolic cycles [49] and muscle contraction [87]. In PPI, two proteins A and B are brought into direct physical contact [104, 47]. In such a process, certain re-

MGDVEKGKKIFIMK**CSQCHGG**TVEKGGKHK

ACT**TTATTG**CAATAGAAAATC

a) A protein sequence

b) A DNA sequence

Figure 1.1: An illustration of a functional region in a sequence. a) MGDVEKGKKI-FIMK**CSQCHGG**TVEKGGKHK is a protein sequence of Cytochrome C, where the highlighted region is a functional region which is the binding site for binding to a heme molecule [13, 154]. b) ACT**TTATTG**CAATAGAAAATC is a DNA sequence taken from the TRANSFAC database [89], where the highlighted region is a functional region, which encompasses a transcription factor binding site (TFBS), verified by Protein Data Bank (PDB) [12] record (ID: 1CQT).

regions of a protein are in close contact with certain regions of another protein. Protein-DNA interaction (or TF-TFBS Binding) [84] is another major mechanism, allowing DNA to initiate a transcription of a gene expression. For the genetic information encoded in a DNA sequence, i.e. a gene, to be expressed, a special type of protein, known as a Transcription Factor (TF), has to bind to a segment of DNA called the transcription factor binding site (TFBS) located around the target gene [84]. This is essential in regulating genetic activities [84]. It is also observed that binding is sequence-specific and subtle changes in the binding sites may affect binding specificity. For example, *PKVVIL* binds *CACGTG* and *PKVEIL* binds *CAGCTG* but not vice versa [17], where the V/E variation is indicative for distinguishing Myc from MRF families [17]. These site-specific variants are important for understanding the underlying interaction mechanisms between interacting sequences.

Therefore, the identification of functional regions with mutations from Protein and DNA is vitally important in bioinformatics. Such knowledge if spotted effectively can reveal the crucial mutation hotspots [134], not only enabling us to have a better scientific understanding but also help the design of new drugs [142, 134]. Traditionally, these functional regions are identified in resolved high-resolution 3D structures obtained by X-ray crystallography [149] or nuclear magnetic resonance spectroscopic experiments [6]. However, these experiments are expensive, labor-intensive and time-consuming. Thus, sequence-based functional region identification methods are motivated by such great need of development.

1.1 Problem Definition

This thesis formulates two problems in sequence-based functional region identification. The first problem is the identification of functional regions with mutations from a set of sequences. Given a set of sequences within the same family or demonstrating similar biological functions, the outputs are (1) the starting and ending address locations of functional regions on the sequences if they exist; (2) an alignment of the functional regions. Alignment [138] here refers to a process that inserts gaps into a set of sequences such that the vertical similarity is maximized. Fig. 1.2 provides a simplified illustration of the problem. As shown in Fig. 1.2(a), the input data is a set of sequences (S0 to S8). It is a simplified dataset as it only has 9 sequences (S0-S8). As shown in Fig. 1.2(b), the functional regions in the input data are highlighted in color for illustration. As shown in Fig. 1.2(c), the output data is the aligned functional regions of a set of sequences, with their sequence ids, starting and ending address locations.

Extending from the first problem, the second problem is to identify co-occurring functional regions with mutations between interaction sequences such as Protein-Protein and Protein-DNA interaction sequences.

Here we first describe the concept of co-occurrence of functional regions, from a more general perspective. By co-occurrence, we perceive it as the co-existence of conserved regions within a functional domain governed by certain underlying biological functionality. These functional regions may co-occur on the same sequence [72, 73] to induce folding, interaction, direct binding or binding to another molecular complex. They might also be found within observed or conjectured functional domains consisting of a pair or a group of interaction sequences, implying their involvement in co-operative mechanisms.

In this thesis, we focus on the identification of functional regions co-occurring between two interaction sequences, associating particularly with Protein-DNA interaction and Protein-Protein interaction sequences. Due to mutations, certain functional regions may cease to function or alter their function. Because of this, when sequences are found within an interaction environment, where the actual interaction regions are not pinpointed because of the expensive cost associated, additional measures or methods have to be introduced to sort out which regions are corresponding to the actual interaction. A solution is proposed in this dissertation work.

Specifically, Fig. 1.3(a) provides an illustration of a simplified transaction database using Protein-DNA interaction sequence data as an example. Each transaction (T_0 to T_3) records an experimental record of protein sequences binding a set of DNA sequences but the exact binding sites have not been identified. Fig. 1.3(b) provides an illustration of

S0 MGDVEKGKKIFIMKCSQCHGGPEGLTAS
 S1 IVAFKTGLSCNEPDRWCSQCHGGEGTPALS
 S2 RACSPGLKNWDVFETICSQCHGGSTEPGLA
 S3 DASKNVFIGCELRWTPCSGMCHGGGSPALTE
 S4 ICGTAEPNRLDFSQVWCSGMCHGGGLPTEGAS
 S5 LPNCRATEWIKFSDGVCSGMCHGGAGTLPSE
 S6 SKNWGVFLCRADPIETCSACHGGPALSGETE
 S7 PKAGNEILVSTRFDWCCSQMMCHGGSPATELG
 S8 PKNFSGIRCVLTWADECSQHGGKTESPLAG

S0 MGDVEKGKKIFIMK**CSQCHGG**PEGLTAS
 S1 IVAFKTGLSCNEPDRW**CSQCHGG**EGTPALS
 S2 RACSPGLKNWDVFETI**CSQCHGG**STEPGLA
 S3 DASKNVFIGCELRWTP**CSGMCHGGG**SPALTE
 S4 ICGTAEPNRLDFSQVW**CSGMCHGGGL**PTEGAS
 S5 LPNCRATEWIKFSDGV**CSGMCHGG**AGTLPSE
 S6 SKNWGVFLCRADPIET**CSACHGG**PALSGETE
 S7 PKAGNEILVSTRFDWCC**SQMMCHGG**SPATELG
 S8 PKNFSGIRCVLTWADE**CSQHGG**KTESPLAG

a) Input data (Simplified): Sequence Set Alpha

b) Input data (Simplified) with highlighted functional region for illustration

C	S	Q	-	-	C	H	G	G	S0: [(14,20)]; s1:[(16,22)]; s2:[(16,22)];
C	S	G	M	-	C	H	G	G	S3: [(16,23)]; s4: [(16,23)]; s5: [(16,23)]
C	S	A	-	-	C	H	G	G	S6:[(16,22)]
C	S	Q	M	M	C	H	G	G	S7:[(16,24)]
C	S	Q	-	-	-	H	G	G	S8:[(16,21)]

c) Output data

Figure 1.2: Problem 1: Identification and alignment of functional regions with mutations. This figure illustrates the problem of identification and alignment of functional regions with mutations from a set of sequences. Given a set of sequences within the same family or demonstrating similar biological functions, the outputs are the starting and ending address locations of functional regions on the sequences if they exist, as well as an alignment of the functional regions. a) Input data, a set of sequences. This is a simplified dataset as it only has 9 sequences (S0-S8). b) Input data, with the functional regions highlighted in color for illustration. c) Output data, aligned functional regions of a set of sequences, with their sequence ids, starting and ending address locations.

a transaction database with the co-occurring functional regions highlighted. Here, a co-occurring functional region refers to a pair of functional regions occurring in both sides of the sequence pair in a transaction. We observe that 1) FDERRMFR and ACTTCCG co-occur in transactions T_0 and T_1 ; 2) FDERMMFR and ACTCCCG co-occur in transaction T_2 . As shown in Fig. 1.3(c), the output data is Protein-DNA Co-Occurrence Aligned Pattern Cluster, capturing co-occurring functional regions with mutations in both Protein-side and DNA-side. Fig. 1.4 provides an illustration of a real transaction database of Protein-DNA interaction sequence data from TRANSFAC [89]. An illustration of Protein-DNA interaction sequences from TRANSFAC [89]. Each transaction records an experimental record of protein sequences binding a set of DNA sequences but the exact binding sites have not been identified. In each transaction, the protein sequence can be as long as 500 amino acids (residues) on average. Also, in each transaction, there can be on average 22 DNA sequences, where each DNA sequence can have on average 25 nucleotides. There can be more than 700 transactions in TRANSFAC [89].

There is also a slight variant of the second problem. It is to identify co-occurring functional regions with mutations between interaction sequences such as Protein-Protein interaction sequences, while an interaction label is introduced to each transaction. A ‘+’ interaction label indicates that this is an experimentally-proven record of interaction. A ‘-’ interaction label indicates that this is an experimentally-proven record of non-interaction. It should be noted that for both cases the binding sites are not indicated in the transaction. Fig. 1.5(a) provides an illustration of a transaction database using Protein-Protein interaction sequence data as an example. Fig. 1.5(b) provides an illustration of a transaction database with the co-occurring functional regions highlighted. As shown in Fig. 1.5(c), the output data is Protein-Protein Co-Occurrence Aligned Pattern Cluster, capturing the co-occurring functional regions with mutations in both protein sides.

1.2 Challenges and Objectives

One major challenge in the identification of functional regions from a set of sequences is to capture the functional regions with mutations. As shown in Fig. 1.6(a), there is a set of sequence, i.e. Sequence Set Alpha, which has 9 sequences (s0-s8), containing two functional regions, CSQCHGG and CSGMCHGG, as well as the functional region with substitution mutation CSACHGG in s6, the functional region with insertion mutation CSQMMCHGG in s7, and the functional region with deletion mutation CSQHGG in s8.

Due to technological limitations, existing methods, such as those based on position weight matrices (PWMs) [151], have to constrain the functional regions to having a fixed

	Protein (TF) Sequence	DNA (TFBS) Sequence(s)
T_0	AQQQFDERRMFROPOP	{GACTTCCGG}
T_1	MNNNFDERRMFRKIKI	{ACTTCCGA}
T_2	WSDEFDERMMFRJJC	{GACTCCCGTTC}
T_3	HKWEVHMRHVHKJV	{GCACTT; AAGTAC}

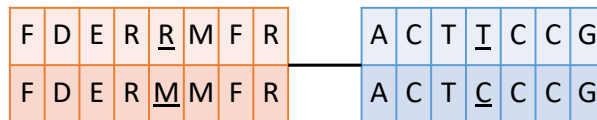
Transaction Database (Simplified): Each transaction (T_0 to T_3) records an **experimental** record of protein sequence binding a set of DNA sequences but the **exact binding sites** have not been identified

a) Input Data (Simplified): a Protein-DNA interaction sequence database in transaction format

	Protein (TF) Sequence	DNA (TFBS) Sequence(s)
T_0	AQQQ <u>FDERRM</u> FROPOP	{ <u>GACTTCCGG</u> }
T_1	MNNN <u>FDERRM</u> FRKIKI	{ <u>ACTTCCGA</u> }
T_2	WSDE <u>FDERRM</u> FRJJC	{ <u>GACTCCCGTTC</u> }
T_3	HKWEVHMRHVHKJV	{GCACTT; AAGTAC}

Co-occurrence: 1) FDERRM and ACTTCCG co-occur in transactions T_0 and T_1 ;
 2) FDERRM and ACTCCG co-occur in transaction T_2

b) Input Data (Simplified) with highlighted functional regions for illustration



c) Output Data: Protein-DNA Co-Occurrence Aligned Pattern Cluster, capturing co-occurring functional regions with mutations

Figure 1.3: Problem 2: Identification of co-occurrence functional regions with mutations between interaction sequences, using Protein-DNA interaction sequences as an example. Given a transaction database of interaction sequences, the output is to find out the co-occurring functional regions with mutations. a) Input data, a simplified transaction database of protein-DNA interaction sequences. Each transaction (T_0 to T_3) records an experimental record of protein sequences binding a set of DNA sequences but the exact binding sites have not been identified. This is a simplified dataset as it only contains 4 transactions and the protein sequences only have less than 20 amino acids. b) Input data, with the co-occurring functional regions highlighted for illustration. We observe that 1) FDERRM and ACTTCCG co-occur in transactions T_0 and T_1 ; 2) FDERRM and ACTCCG co-occur in transaction T_2 . c) Output data, Protein-DNA Co-Occurrence Aligned Pattern Cluster, capturing co-occurring functional regions with mutations in both Protein-side and DNA-side.

An Illustration of Protein-DNA interaction sequences from TRANSFAC

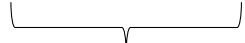
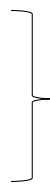

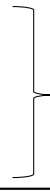
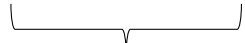

ID	Protein	DNA (On average: 25 nucleotides long)
0	RNDCE...QQQG  On average: ~500 residues long	ATACC...AC CCGTTAA ... AC...GTT  On average: 22 sequences
1	GQERR...NNNG  On average: ~500 residues long	GTATC...TT ATGCCGG ... TTACCCT  On average: 22 sequences
...		
N	RREQQ...MNGT  On average: ~500 residues long	CCGAA...GT GACTTAA ... CGGGTCC  On average: 22 sequences

Figure 1.4: An illustration of Protein-DNA interaction sequences from TRANSFAC [89]. Each transaction records an experimental record of protein sequences binding a set of DNA sequences but the exact binding sites have not been identified. In each transaction, the protein sequence can be as long as 500 amino acids (residues) on average. Also, in each transaction, there could be on average 22 DNA sequences, where each of them can have on average 25 nucleotides. There could be more than 700 transactions in TRANSFAC [89].

	Protein Sequence 1	Protein Sequence 2	Interaction Label
T ₀	AQQQFDERRMFROPOP	DAEVMPGQYNTHGALHSN	+
T ₁	MNNNFDERRMFRKIKI	CPCPGQYNTHGQNP	+
T ₂	WSDEFDERMMFRJJC	KHPGQONTKGKEF	+
T ₃	HKWEVHMRHVHKJV	REVFQKMAAECTQGT	-

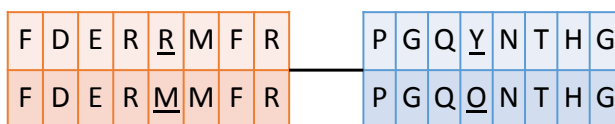
Transaction Database: Each transaction (T₀ to T₃) records an **experimental** record of protein sequence binding (+) or not binding (-) another protein sequence indicated by the **interaction label**, but the **exact binding sites** have not been identified

a) Input Data, a Protein-Protein interaction sequence database in transaction format with interaction labels

	Protein Sequence 1	Protein Sequence 2	Interaction Label
T ₀	AQQQ <u>FDERRM</u> FRROPOP	DAEVM <u>PGQYNTHG</u> ALHSN	+
T ₁	MNNN <u>FDERRM</u> FRKIKI	CPC <u>PGQYNTHG</u> QNP	+
T ₂	WSDE <u>FDERRM</u> FRJJC	KH <u>PGQYNTHG</u> KEF	+
T ₃	HKWEVHMRHVHKJV	REVFQKMAAECTQGT	-

Co-occurrence: 1) FDERRMFR and PGQYNTHG co-occur in transactions T₀ and T₁;
 2) FDERRMFR and PGQYNTHG co-occur in transaction T₂

b) Input Data with highlighted functional regions



c) Output Data, Protein-Protein Co-Occurrence Aligned Pattern Cluster, capturing co-occurring functional regions with mutations

Figure 1.5: Problem 2b: Identification of co-occurrence functional regions with mutations between interaction sequences with interaction labels. Given a transaction database of interaction sequences, while an interaction label is introduced to each transaction, the output is to find out the co-occurring functional regions with mutations. a) input data, a transaction database of protein-protein interaction sequences with interaction label; '+' indicates that this is an experimentally-proven record of interaction. '-' indicates that this is an experimentally-proven record of non-interaction. b) input data, with the co-occurring functional regions highlighted. c) output data, Protein-Protein Co-Occurrence Aligned Pattern Cluster, capturing co-occurring functional regions with mutations in both protein sides.

width and thus will fail to identify functional regions with insertion mutation and will take in noises when identifying functional regions with deletion mutation. MEME is the most popular existing software among existing methods that based on PWMs [151]. As shown in Fig. 1.6(b), the output of MEME [7] on the Sequence Set Alpha is demonstrated. We observe that MEME assumes the functional region has a fixed width of 7 and outputs a PWM with a width of 7. Thus, MEME [7] fails to identify the functional regions with insertion mutation (e.g. CSQMMCHGG in s7) and took in noises when identifying functional regions with deletion mutation (i.e. CSQHGG in s8). Furthermore, these methods are not efficient since their determination of the optimal width parameter has to rely on exhaustive search.

Recently, Aligned Pattern Clustering (APCn) [77, 143] was introduced to identify from a set of sequences functional regions as Aligned Pattern Clusters (APCs) [77, 143] by grouping and aligning patterns with variable width. As shown in Fig. 1.6(c), the output of the existing software WeMine running APCn [77, 143] on the Sequence Set Alpha is demonstrated. With this method, first, variable-width patterns with high frequencies of occurrence and sufficient statistical significance are discovered [77, 143]. Then, these patterns are clustered based on column similarity, where gaps can be introduced. The outputs are referred to as APCs [77, 143]. Nevertheless, a drawback of APCn is that it cannot identify functional regions with mutations if their frequency of occurrences is too low to be considered as a pattern. Hence, a new algorithm was developed to overcome these challenges.

In addition, in the problem of identifying co-occurring functional regions with mutations between interaction sequences in Protein-DNA interaction [79] and Protein-Protein interaction [111, 109, 112], existing algorithms do not have representation models allowing mutations. Given a database of protein-DNA interaction sequences in transaction format as shown in Fig. 1.3(a), the traditional representation model, one-to-one mapped associated patterns [79] as shown in Fig. 1.7(a), cannot capture the mutations (or site variations). Hence, a new representation model for capturing co-occurring functional regions with mutations, and a discovery algorithm for discovering such representation model from interaction sequence data, have been developed.

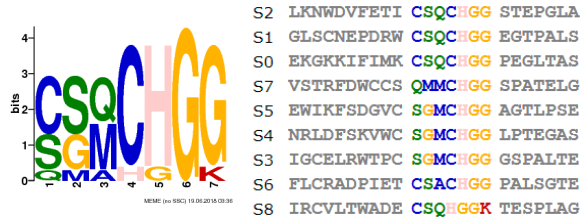
Based on this new representation model, it is also interesting to investigate if it can help to build better sequence-based Protein-Protein Interaction machine learning prediction model. The challenge here is how to construct a feature vector using the new representation model. Fig. 1.8 provides an illustration.

Therefore, the objectives of this thesis are to:

- discover functional regions with substitution, insertion and deletion mutations

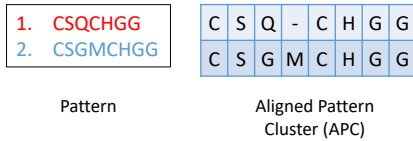
S0 MGDVEKGGKIFIMK**CSQCHGG**PEGLTAS
 S1 IVAFKTLGLSCNEPDRW**CSQCHGG**EGTPALS
 S2 RACSPGLKNWDFETI**CSQCHGG**STEPGLA
 S3 DASKNVFIGELRWTP**CSGMCHGG**GSPALTE
 S4 ICGTAEPNRLDFSKVW**CSGMCHGG**LPTEGAS
 S5 LPNCRATEWIKFSDGV**CSGMCHGG**AGTLPSE
 S6 SKNWGVFLCRADPIET**CSACHGG**PALSGTE
 S7 PKAGNEILVSTRFDWCC**CSQMMCHGG**SPATELG
 S8 PKNFSGIRCVLTWAE**CSQHGG**KTESPLAG

a) Sequence Set Alpha



Position Weight Matrix (PWM) Functional Regions Identified by MEME

b) Output of the most popular existing software MEME on the Sequence Set Alpha



c) Output of the existing software WeMine running Aligned Pattern Clustering (APCn) on the Sequence Set Alpha

C S Q - - C H G G	S0: [(14,20)]; s1: [(16,22)]; s2: [(16,22)];
C S G M - C H G G	S3: [(16,23)]; s4: [(16,23)]; s5: [(16,23)]
C S A - - C H G G	S6: [(16,22)]
C S Q M M C H G G	S7: [(16,24)]
C S Q - - - H G G	S8: [(16,21)]

d) Output of the proposed algorithm Pattern-Directed Aligned Pattern Clustering (PD-APCn)

Figure 1.6: An illustration of a sequence set with mutations and demonstration of outputs by existing software. a) Sequence Set A is composed of 9 sequences (s0-s8), containing two functional regions, CSQCHGG and CSGMCHGG, as well as the functional region with substitution mutation CSACHGG in s6, the functional region with insertion mutation CSQMMCHGG in s7, and the functional region with deletion mutation CSQHGGG in s8. b) The output of the most popular existing software MEME [7] on the Sequence Set Alpha is demonstrated. We observe that MEME assumed the functional region having a fixed width of 7 and outputted a position weight matrix (PWM) [151] with a width of 7. c) The output of the existing software WeMine running APCn [77, 143] on the Sequence Set Alpha is demonstrated. We observe that APCn cannot identify functional regions with mutations, since their frequencies of occurrence are too low to be considered as patterns. d) The output of the proposed algorithm Pattern-Directed Aligned Pattern Clustering (PD-APCn) [126, 127] is demonstrated. PD-APCn [126, 127] can identify all functional regions with mutations, indicating their starting and ending address locations with sequence ids, in an efficient manner.

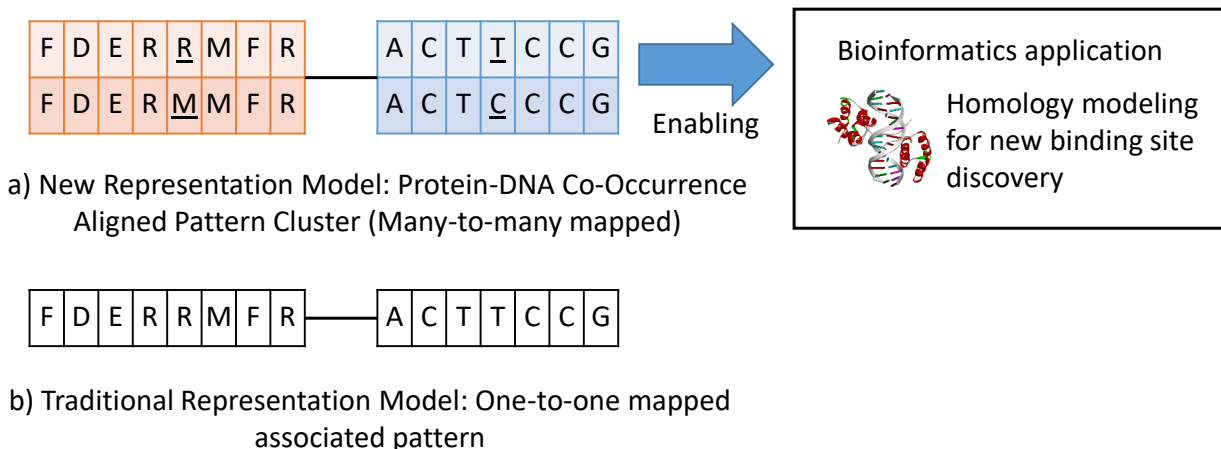


Figure 1.7: An illustration of Protein-DNA Co-Occurrence Aligned Pattern Cluster. a) New representation model: Protein-DNA Co-Occurrence Aligned Pattern Cluster, which is many-to-many mapped, enabling the use of homology modeling [116] to discover new binding sites by considering all pair-wise combinations between both sides. b) Traditional representation model: One-to-one mapped associated pattern [79], where mutations are not captured.

- use an efficient process to determine the model width adaptively from data without exhaustive search
- develop a new representation model for capturing co-occurring functional regions with mutations
- develop a new algorithm to discover such a representation model from interaction sequence data in the format of a transaction database
- develop applications of the new representation model in real Protein-DNA and Protein-Protein interaction sequence data

1.3 Contributions

The contribution of this thesis can be stated as follows:

- This thesis proposes a new algorithm known as Pattern-Directed Aligned Pattern Clustering (PD-APCn) [126, 127] which can identify functional regions with mu-

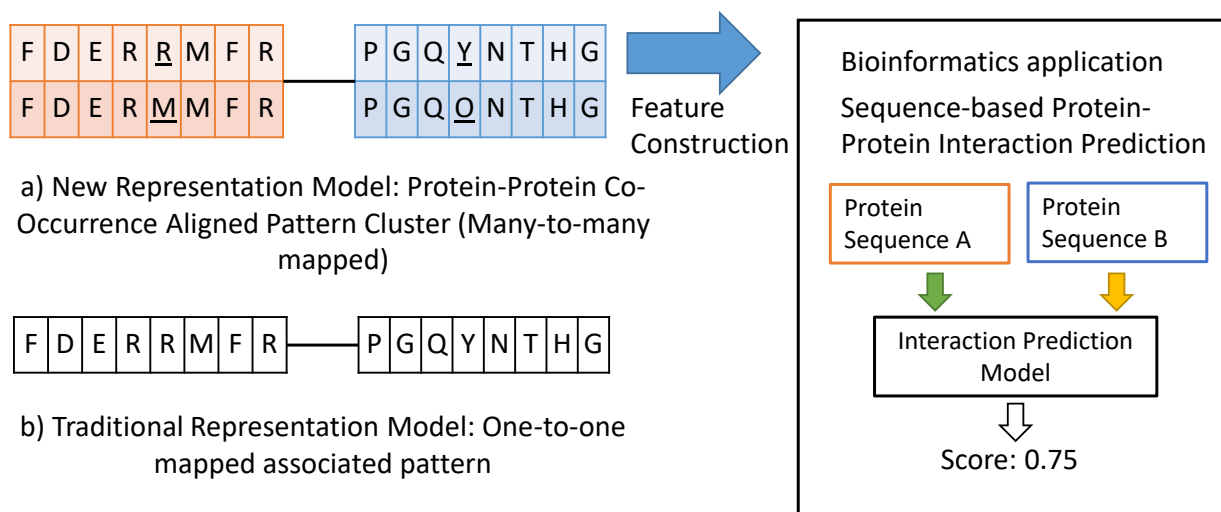


Figure 1.8: An illustration of Protein-Protein Co-Occurrence Aligned Pattern Cluster. a) New representation model: Protein-Protein Co-Occurrence Aligned Pattern Cluster, which is many-to-many mapped. Leveraging the proposed *APC – PPI* score [124, 125] in this thesis, feature vectors can be constructed using the Protein-Protein Co-Occurrence APC to train a sequence-based protein-protein interaction prediction model, using a transaction database of protein-protein interaction sequences with interaction label as training data. b) Traditional representation model: One-to-one mapped associated pattern [109, 112]. Our experimental results [124, 125] on 40 independent datasets demonstrated our prediction model are better than the ones based on the traditional representation model [109, 112].

tations from sequence data. Compared to the existing algorithm Aligned Pattern Clustering (APCn) [77, 143], it does not require users to specify the representation model width parameter, and can identify mutated functional regions which have low frequencies of occurrence. Compared to the most popular existing software MEME [7], in our experiments on synthetic datasets, PD-APCn could identify the implanted functional regions and outperform with higher recall and Fmeasure over a computational speed-up of up to 665 times [126, 127]. When applying PD-APCn on real datasets from Cytochrome C and Ubiquitin protein families, all key binding sites in the families were captured in the APC outputs. [127]. As shown in Fig. 1.6(d), the output of the proposed algorithm Pattern-Directed Aligned Pattern Clustering (PD-APCn) [126, 127] is demonstrated. PD-APCn [126, 127] can identify the functional regions with mutations, indicating their starting and ending address locations with sequence ids, while APCn and MEME cannot.

- This thesis proposes a new representation model known as Co-Occurrence APC [74, 124, 75, 125], and the discovery algorithm for discovering co-occurring functional regions between interaction sequences in transaction database format via the Co-Support score [74, 124, 75, 125].
- This thesis presents an application of Co-Occurrence APC to Protein-DNA interaction. We developed Protein-DNA Co-Occurrence APC [74, 75], as shown in 1.7(b), for the discovery of protein-DNA binding cores with higher precision (up to 20% more precise) with a 1600 times faster run-time than those of its contemporaries. The significant of the speed-up is attributed to replacing the combinatorial search of one-to-one co-occurrence in the entire transaction dataset to the many-to-many search of co-occurrences between patterns within each high-ranking Protein-DNA Co-Occurrence APC (Fig. 4.1). New Protein-DNA binding cores revealed by Protein-DNA Co-Occurrence APC are also discovered via homology modeling [116] as a proof-of-concept [74, 75].
- This thesis presents an application of Co-Occurrence APC to Protein-Protein interaction [124, 125]. We developed Protein-Protein Co-Occurrence APCs [124, 125], as shown in 1.8(b), to construct feature vectors by the proposed *Match* score [124, 125], *MaxMatch* score [124, 125] and *APC – PPI* score [124, 125]. Based on such feature vectors, we built machine learning prediction models for sequence-based Protein-Protein interaction prediction which outperformed its nearest counterpart PIPE2 [109, 112] among 40 independent datasets. The significance of APC [77, 143] enables WeMine-P2P to have pattern variants and flexible width in the features, leading

to stronger interpretability. A list of interpretable biological features discovered via WeMine-P2P has been rendered in Tables 5.5 and 5.6.

1.4 Thesis structure

This thesis is presented with the following structure. Chapter 2 provides a summary of the background knowledge and related work. Chapter 3 introduces the new Pattern-Directed Aligned Pattern Clustering (PD-APCn) and the experimental results. Chapter 4 introduces the new representation model known as Co-Occurrence APC and its applications to Protein-DNA interaction. Chapter 5 introduces the application of Co-Occurrence APC to Protein-Protein interaction. Chapter 6 concludes the thesis and indicates potential future work.

Chapter 2

Background and Related Work

In this chapter, the basic biological background knowledge is introduced. For completeness, the concept of Protein, DNA and the Central Dogma of Molecular Biology is briefly presented, followed by a review on Protein-DNA interaction and Protein-Protein interaction.

2.1 Protein, DNA and the Central Dogma

Amino acid (Residue) is a fundamental organic compound with amine ($-\text{NH}_2$), carboxylic ($-\text{COOH}$) functional groups and a specific side chain. Different amino acids are differentiated by the side chains attached. Human body has 20 standard amino acids, which are {A, R, N, D, C, E, Q, G, H, I, L, K, M, F, P, S, T, W, Y, V}. An amino acid, also called a residue, can form a peptide bond with another amino acid.

Protein can be simply interpreted as a linear sequence of amino acids, i.e. a string of alphabets, where each alphabet represents one amino acid. For example, RAWYVFMP is a protein sequence. Protein structure can be divided into 4 distinct levels, from primary to quaternary. The primary structure of a protein refers to the linear sequence of amino acids linked by peptide bonds. The secondary, tertiary and quaternary refer to the 3D arrangement of the bonding. Hence, a 3D structure of a protein reveals 4 levels of information.

Nucleotides are (organic) molecules where each of them has a five-carbon sugar, a specific nitrogenous base and at least one phosphate group. There are five types of nitrogenous bases including Adenine (A), Uracil (U), Guanine (G), Thymine (T), and Cytosine (C). This is called complementary base pairing.

DNA (deoxyribonucleic acid) is interpreted as a sequence of nucleotides. Only 4 types of nucleotide (A, C, G, T) are present in DNA. For example, ACAGATTT is a DNA sequence.

The Central Dogma of Molecular Biology The central dogma of molecular biology normally refers to how a protein is made from DNA. As shown in Fig. 2.1, DNA is first transcribed into messenger Ribonucleic Acid (mRNA) and mRNA is then translated into protein. mRNA serves as the template of DNA to convey the genetic information. After the transcription, mRNA is then translated to protein. A gene is a segment of DNA encoding a protein. The process which turns a gene into a protein is called gene expression.

2.2 Functional Regions in Protein and DNA

As mentioned in Chapter 1, for protein and DNA to actually exert their biological functions, they have to participate in biological mechanisms, in which they interact with others via their functional regions. Therefore, the identification of functional regions with mutations from Protein and DNA is vitally important in bioinformatics. Such knowledge if spotted effectively could reveal the crucial mutation hotspots [134], not only enabling us to have a better scientific understanding but also help the design of new drugs [142, 134]. Fig. 1.1 provides an illustration of functional regions.

Traditionally, functional regions in Protein and DNA are identified in resolved high-resolution 3D structures obtained by X-ray crystallography [149] or nuclear magnetic resonance spectroscopic experiments [6]. However, these experiments are expensive, labor-intensive and time-consuming.

Under evolutionary pressure, these functional regions normally remain conserved [81]. Thus, sequence-based identification methods are feasible. To identify them, domain annotation [33] leverages existing databases (such as PFam [34]) or profile hidden markov models [35]. Nevertheless, functional regions not recorded in existing databases or too distinct from the recorded ones cannot be identified.

2.2.1 Multiple Sequence Alignment (MSA)

For de novo discovery of functional regions, Multiple Sequence Alignment (MSA) [132] is one approach, but it is suitable only for globally homologous sequences with a high level of similarity [132]. Even within the same protein family, this “homologous” assumption may not hold. For example, in the class A Scavenger Receptor [140] with five subclasses, the width of collagenous domains varies in subclasses from 75 to 250 amino acids [62].

2.2.2 Motif Discovery

Motif Discovery is an approach to locate and align locally homologous subsequences [37]. Throughout the years, many algorithms [95, 82, 7, 139], have been developed to find unknown patterns (*de novo* motifs) only from a set of protein or DNA sequences.

MEME [7] is the most popular method to represent such homologous sub-sequences by a position weight matrix (PWM) [151] which is fixed-width, but the span of protein functional regions, with frameshifts (insertion and deletion mutations) varies in width. A graphical illustration of the output of the motif discovery algorithm MEME [7] showing a PWM is provided in Fig. 1.6(b).

Furthermore, to identify the width parameter of a PWM requires exhaustive computational intensive search. In MEME [7], the search range of the default PWM width parameter varies from 8 to 50. This is a default option of motif discovery software for a bioinformatics scientist. In addition, GLAM2 [38] is also a popular motif discovery algorithm, with its specialty in identifying motifs with gaps, and it is often used as a benchmark algorithm [113] or integrated into a bioinformatics pipeline [69]. Both of them were used as benchmark algorithms in Chapter 3.

2.2.3 Aligned Pattern Clustering (APCn)

Aligned Pattern Clustering (APCn) [77, 143] was introduced to discover functional regions with variable width from protein family sequences as Aligned Pattern Clusters (APCs) [77, 143]. In this section, we briefly describe Aligned Pattern Clusters (APCs) [77, 143], and their discovery algorithm Aligned Pattern Clustering (APCn) [77, 143].

Definition of Aligned Pattern Cluster (APC)

An APC [77, 143] is a group of sequence patterns augmented by inserting gaps - and wildcards *, such that the augmented sequence patterns share a high column similarity and each of them has the same length.

Sequence

Let Σ be a set of alphabets. Let s_k be a sequence comprising of alphabets in Σ , i.e. $s_k = s_k^1 s_k^2 \dots s_k^{|s_k|}$, where $s_k^j \in \Sigma, \forall j = 1, 2, \dots, |s_k|$.

Sequence Set

Let S be a set of sequences, i.e. $S = \{s_k | k = 1, 2, \dots, |S|\}$.

Pattern

A pattern \bar{p} is defined as an ordered sequence of interdependent symbols from Σ , i.e. $\bar{p} = \bar{p}_1 \bar{p}_2 \dots \bar{p}_{|\bar{p}|}$, where $\bar{p}_j \in \Sigma, \forall j = 1, 2, \dots, |\bar{p}|$, that passes the requirements [144], such as minimum width min_{width} , maximum width max_{width} , minimum occurrence $min_{occurrence}$, confidence interval $conf_{interv}$, redundancy pruning (delta-closed) threshold $delta_{closed}$.

Pattern Discovery Algorithm

Given a sequence set S , minimum width min_{width} , maximum width max_{width} , minimum occurrence $min_{occurrence}$, confidence interval $conf_{interv}$, redundancy pruning (delta-closed) threshold $delta_{closed}$, a set of patterns \bar{P} by the pattern discovery algorithm [144], i.e. $\bar{P} = \{\bar{p}^i | i = 1, \dots, |\bar{P}|\} = \{\bar{p}^1, \bar{p}^2, \dots, \bar{p}^{|\bar{P}|}\}$. A graphical illustration of the output of the pattern discovery algorithm [144] is provided in 1.6(c).

Aligned Pattern Cluster (APC)

Given a set of patterns $\bar{P}^l = \{\bar{p}^{l,1}, \bar{p}^{l,2}, \dots, \bar{p}^{l,m_l}\}$, an APC C^l is defined as

$$C^l = \text{ALIGN}(\bar{P}^l) \quad (2.1)$$

$$= \text{ALIGN} \begin{pmatrix} \bar{p}^{l,1} \\ \bar{p}^{l,2} \\ \vdots \\ \bar{p}^{l,m_l} \end{pmatrix} = \begin{pmatrix} p^{l,1} \\ p^{l,2} \\ \vdots \\ p^{l,m_l} \end{pmatrix} = (P^l) \quad (2.2)$$

$$= \begin{pmatrix} \sigma_1^{l,1} & \sigma_2^{l,1} & \dots & \sigma_{n_l}^{l,1} \\ \sigma_1^{l,2} & \sigma_2^{l,2} & \dots & \sigma_{n_l}^{l,2} \\ \vdots & \vdots & \vdots & \vdots \\ \sigma_1^{l,m_l} & \sigma_2^{l,m_l} & \dots & \sigma_{n_l}^{l,m_l} \end{pmatrix}_{m_l \times n_l}, \quad (2.3)$$

where $\sigma_j^{l,i} \in \Sigma \cup \{-\} \cup \{*\}$, $\forall i = 1, 2, \dots, m_l, \forall j = 1, 2, \dots, n_l$, and ALIGN [77, 143] is a process to maximize the column similarity in \bar{P}^l , by inserting gaps and wildcards, to obtain a set of aligned patterns $P^l = \{p^{l,1}, p^{l,2}, \dots, p^{l,m_l}\}$ with the same length n_l .

Aligned Pattern Clustering (APCn) algorithm

Given a set of patterns $\bar{P} = \{\bar{p}^1, \bar{p}^2, \dots, \bar{p}^{|\bar{P}|}\}$, a set of APCs $C = \{C^1, C^2, \dots, C^{|\bar{P}|}\}$, can be obtained by the Aligned Pattern Clustering (APCn) algorithm [77, 143]. The algorithm is illustrated in Algorithm 1.

Thus, an APC is formed by clustering and aligning the patterns discovered by the pattern discovery algorithm [144]. A graphical illustration of the output of the Aligned Pattern Clustering (APCn) algorithm [77, 143] is provided in 1.6(c). It is a crucial observation as we find out that if a pattern is missed by the Pattern Discovery algorithm [144], it will not be discovered by the Aligned Pattern Clustering (APCn) algorithm [77, 143]. As shown in Fig. 1.6(a), Sequence Set A is composed of 9 sequences (s0-s8), containing two functional regions, CSQCHGG and CSGMCHGG, as well as the functional region with substitution mutation CSACHGG in s6; the functional region with insertion mutation CSQMMCHGG in s7; and the functional region with deletion mutation CSQHGGG in s8. As shown in Fig. 1.6(c), we observe that APCn cannot identify functional regions with rare mutations, such as CSACHGG on S6, CSQMMCHGG on S7 and CSQHGGG on S8, since their frequencies of occurrence are too low to be considered as patterns.

Algorithm 1 Aligned Pattern Clustering (APCn) algorithm [77, 143]

Input: a set of patterns $\bar{P} = \{\bar{p}^1, \bar{p}^2, \dots, \bar{p}^{|\bar{P}|}\}$, a minimum similarity threshold, $min_{Similarity}$.

Output: a set of APCs $C = \{C^1, C^2, \dots, C^{|\bar{P}|}\}$

Set all $\bar{p}^i \in \bar{P}$ as $C^i \in C, \forall i = 1, 2, \dots, |\bar{P}|$

for all pairs of (C^i, C^j) **do**

 compute $Similarity(C^i, C^j)$

end for

while True **do**

$s = \text{select max } Similarity(C^{max_i}, C^{max_j})$

if $s < min_{Similarity}$ **then**

 break

end if

$C^{new} = \text{merge}(C^{max_i}, C^{max_j})$

 remove C^{max_i}, C^{max_j} from C

 insert C^{new} into C

for all pairs of (C^{new}, C^i) **do**

 compute $Similarity(C^{new}, C^i)$

end for

end while

return C

2.3 Protein-DNA interaction

A Protein-DNA interaction is referred to as a protein binding a molecule of DNA [123], when they are in close contact ($<3.5\text{\AA}$ [1]), usually via chemical bonds such as hydrogen bonds [117]. Hence, a Protein-DNA interaction is also considered as a Protein-DNA binding. Protein-DNA interactions play essential roles in DNA transcription [85, 84]. A transcription factor (TF) is a special type of protein. A TF regulates (activates or inhibits) the expression of a gene by binding itself in a sequence-specific manner in most cases to a segment of DNA located around the target gene called the transcription factor binding site (TFBS) [85, 84]. Sequence-specific binding is referred to as the ability of a TF to distinguish different DNA sequences.

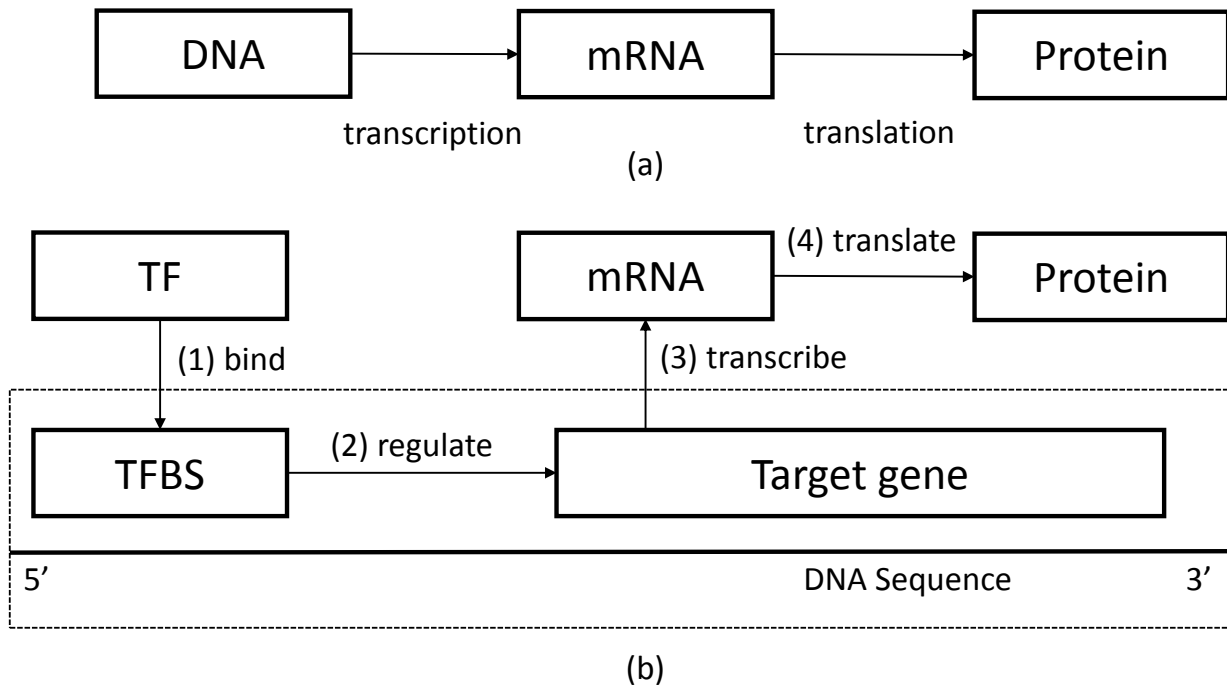


Figure 2.1: This figure illustrates the Central Dogma of Molecular Biology and how Protein-DNA binding regulates (activates or inhibits) transcription. (a) The Central Dogma of Molecular Biology simply refers to how protein is made from DNA. Transcription turns DNA to mRNA and translation turns mRNA to Protein. (b) How Protein-DNA binding regulates transcription is illustrated. (1) A transcription factor (TF), a special type of protein, binds itself to a segment of DNA called transcription factor binding site (TFBS), which is located around the target gene which is a DNA segment coded for protein. (2) Gene transcription is initiated. (3) The target gene is transcribed to mRNA. (4) The mRNA is translated to a protein.

2.3.1 Protein-DNA binding core

A DNA binding domain is the binding region of a TF. It can recognize a collection of similar TFBSs. A domain annotation indicates both the starting position and the ending position of that domain. DNA binding domain annotations are conventionally long, varying from 25 to 500 amino acids. However, according to high-resolution 3D Protein-DNA binding structures, only short regions (<10) of TF and TFBS actually form critical chemical bonds. The regions between a TF and a TFBS in close contact ($<3.5\text{\AA}$ [1, 99]) are referred to as Protein-DNA binding cores [18, 79] (see Figure 2.2). The argument is that such a close contact is unlikely to be a random happening. It is because that in the binding cores critical chemical bonds [71] exist to pull together the residue and the nucleotide. Existing work has shown that this close contact is energetically important [91, 63], causing differential binding when mutated [17], and is important for establishing regulation across model organisms and databases such as SwissRegulon [101]. While Protein-DNA binding cores are relatively short regions (<10), TFBS can be as long as 20 bp [18, 79]. It is observed that subtle changes in binding cores may affect its binding specificity. For example, the V/E variation between *PKVVIL* – *CACGTG* and *PKVEIL* – *CAGCTG* (in bold fonts) is indicative for distinguishing Myc from MRF families [17]. Hence, analyzing and affirming binding cores will not only consolidate our knowledge of gene regulation but also potentially provide additional insights on the binding specificity of transcription factors.

2.3.2 Experimental approaches on studying Protein-DNA interaction

Traditionally, experiments such as DNA footprinting [48] or gel-shift assays [52] were used for studying Protein-DNA interactions. DNA footprinting [48] provides a binary binding signal between the protein and the target DNA sequence (50 - 200 base pairs). Gel-shift assays [52] provide more or less the same signal except that the target DNA sequence can be as short as 5 base pairs [114]. They are good for verification but impractical for discovery due to the large search space required to identify the right sites. Also, they do not provide any binding site information on the protein side. Expensive experiments such as X-ray crystallography or Nuclear Magnetic Resonance (NMR) are conducted to obtain high-resolution 3D Protein-DNA binding structure to identify the Protein-DNA binding cores (<10 residues/ base pairs on both sides). Due to the high cost of the high-resolution 3D structures, the available Protein-DNA 3D structures are limited and far from being complete [142]. Furthermore, as these experiments are labor-intensive and time-consuming, they are unable to be conducted on scale. Therefore, the recent trend

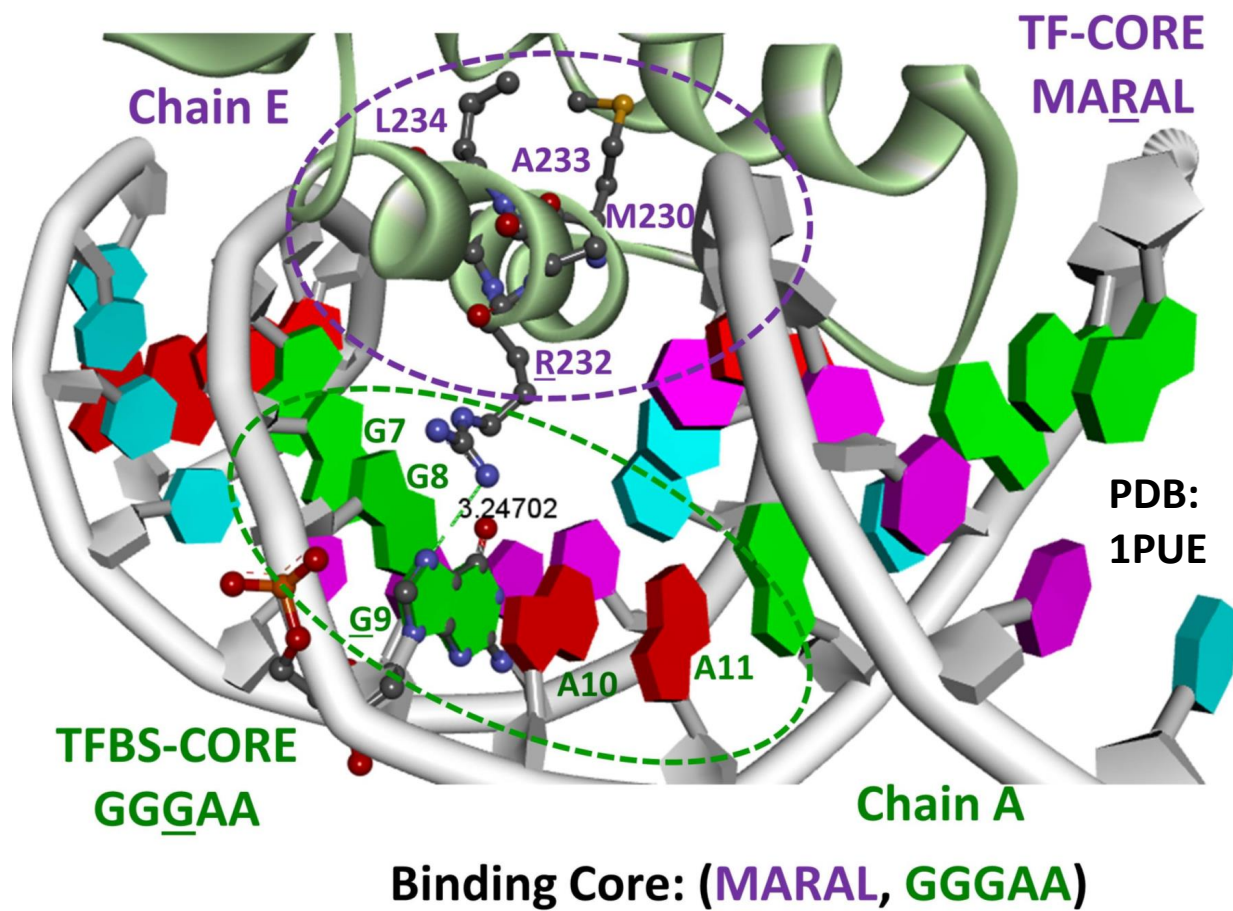


Figure 2.2: [75] A binding core [79] is denoted as a region between a TF and a TFBS in close contact ($<3.5\text{\AA}$ [1, 99]). For example, (MARAL,GGGAA) is a binding core, where a hydrogen bond exists between R and G with a length of 3.25\AA . MARAL and GGGAA are called as a TF-core and a TFBS-core respectively.

to study Protein-DNA interactions is to develop high-throughput technologies [123] such as chromatin immunoprecipitation (ChIP) followed by microarray or sequencing (ChIP-Chip or ChIP-Seq) [16, 102] and protein binding microarray (PBM) [11]. ChIP-Chip or ChIP-Seq [16, 102] sequencing technologies measure the binding occupancy of a particular TF to the nucleotide sequences on a genome-wide basis in vivo (within cells), but at low resolution. These sequencing technologies are only able to indicate a region of 100 or more base pairs [123], and are available in the Encyclopedia of DNA Elements (ENCODE) database [22]. Another emerging sequencing technology is PBM [11, 10] which enables the measuring of the binding preference of a TF to all possible nucleotide sequences with 8 base pairs in vitro (in test tubes) [11, 10] as available at the UniProbe [58]. Nevertheless, it does not provide any binding site information on the TF side.

2.3.3 Computational approaches on studying Protein-DNA interaction

Different approaches have been developed, but most of them are one-sided.

Association Rule Mining. Among the computational techniques, association rule mining is the only few systematic two-sided approach which was recently applied in [79, 148] to mine TF-TFBS associated patterns, e.g. (MARAL, GGGAA), from sequence data only. Despite the satisfactory results, the algorithms [79] represent a TF-TFBS binding by a TF-TFBS associated pattern which is a one-to-one pattern mapping of a protein sequence and a DNA sequence only without considering any variation. One algorithm [146] suggests that it can discover many-to-many mapped TF-TFBS associated patterns using evolutionary algorithms. However, it is still based on one-to-one mapped TF-TFBS associated patterns [79]. It leads to information loss and involves combinatorial trials. Another algorithm in [148] does consider variations by allowing at most 1 mismatch between every TF-TFBS associated pattern, but each of them is represented as a pair of position weight matrices, which is not biologically intuitive for interpretation, and the column-wise associations are also not reserved for analysis.

Unified Score Model. Besides systematic methods, ad-hoc models [20] have been developed to discover TF-TFBS associated patterns with variations but are not totally based on sequences. Recently, a new method called Unified Score Model (USM) [19] was proposed to mine TF-TFBS associated patterns with variations from sequence data only. USM performs motif discovery (allowing at most 1 mismatch) on TF-side and the TFBS-side of each corresponding TF. It is thus computationally intensive as motif discovery is performed for $N + 1$ times, given N transcription factors and their DNA binding sequences.

USM then links up both sides by summing the TF-side and its corresponding TFBS-side motif scores as a unified score to rank TF-TFBS associated patterns. Although variations are considered, the representation of TF-TFBS associated patterns is still one-to-one, giving no site-specific information on variations. For example, assume that the TF-TFBS associated pattern (FQNRRMK, TTATTTG) is discovered by USM, it does not include the information about whether its variants, such as (FQNRRAK, TTAATTG), are possible for binding.

2.4 Protein-Protein interaction

Protein-protein interaction (PPI) is important for biological processes and functions in living cells such as metabolic cycles, DNA transcription and replication, and signaling cascades [40]. Hence, studying PPI is critical for better understanding the molecular mechanisms inside the cell [40]. Following [104, 47], we refer a PPI as an interaction that brings two different proteins A and B into direct and close physical contact ($<6\text{\AA}$ [100]), i.e. heterodimeric interaction. In contrast, most homodimeric interaction, in which proteins A and B are the same, are for maintaining the stability of the interacting complex (as proteins A and B cannot exist independently [97]) but not for regulating cellular processes [97].

2.4.1 Experimental methods on predicting Protein-Protein interaction

Predicting Protein-Protein interaction is a process to predict if one protein will interact with another protein. It is particularly useful for discovering the unknown functions of a target protein [56]. A number of experimental methods have been developed for studying PPI. Low-throughput methods such as crystallography [70] can provide an accurate understanding of the target PPI. However, these methods are expensive, labor-intensive and time-consuming [152, 110], and hence are not suitable for large-scale discovery. Although there are attempts [122, 57] to enhance the throughput, their efficiency remains to be demonstrated. Recently, high-throughput methods such as the yeast two-hybrid (Y2H) systems [60], and tandem affinity purification (TAP) [40] with mass spectrometry [53], have been developed for large-scale PPI detection. Nevertheless, these experimental methods usually suffer from high rates of both false positive and false negative predictions [105, 83]. Hence, developing effective and reliable computational methods to facilitate more accurate prediction of PPI is of fundamental importance [121].

2.4.2 Computational methods on predicting Protein-Protein interaction

Predicting Protein-Protein interaction using computational methods is a process to predict if one protein will interact with another protein based on digital computation but not physical experiments. Existing computational methods for PPI prediction can be divided into four types depending on the input data. The first type such as Computational docking [108] requires three-dimensional structures of the target proteins. It can be applied to the target proteins to simulate if they can interact based on physiochemical properties such as shape complementarity, electrostatics, and biochemical information [39]. The second type requires genomic information of the target proteins, e.g. gene fusion events [31], the conservation of gene-order [23], and the calculation of prior probabilities of genomic features between interacting proteins [61]. The third type requires prior biological knowledge of the target proteins, e.g. phylogenetic profiles [106], domain knowledge of proteins [21, 64, 41] and topological properties of proteins in PPI networks [153]. All these methods do not have general applicability because the required data/information is not always available. The last type of methods require only sequence data. It uses the coded information inherent in sequences to predict if a protein pair interacts. For this reason, sequence-based methods are becoming popular, since sequence data is more readily available nowadays [56].

2.4.3 Sequence-based Protein-Protein interaction Prediction

Sequence-based Protein-Protein interaction Prediction is a process to predict if one protein will interact with another protein using only their sequences as input to a computer program.

PIPE [111] / PIPE2 [109, 112] is a well-established sequence-based method. Given a protein A, a protein B and a database of positive PPIs, PIPE simply counts how frequently all fixed-length protein sequence segments in Proteins A and B found co-occurring in the database. To achieve such a task, all combinations of 20-mers between Protein A and Protein B are first enumerated using a sliding window with a width of 20. Then, the co-occurrence of each combination, e.g. MGIRRLVSVITRPIINKVNS from Protein A and GPEAII LTGTFDDWKGTLP M from Protein B, is searched in the database, and the frequency of their co-occurrence is counted. The sum of all counts is then computed. If the sum is larger than or equal to a threshold, the algorithm then predicts that protein A and B would interact. PIPE2 is a much faster version of PIPE. However, in spite of the satisfactory prediction performance, we observe that there is room for improvement. The key drawback of PIPE/PIPE2 is their use of a fixed-window of 20 amino acids. This

is biologically unrealistic since functional regions such as the Short Linear Motifs (SLiMs [30]) have variable length from 3 to 15 amino acids [30]). Most of them are less than 10 amino acids [90]. Recently, a similar algorithm called VLASPD [56] that allows variable lengths of protein sequence segments is proposed. Nevertheless, it still uses exact patterns, which are neither realistic nor useful for biological analysis since it does not accept variants. Furthermore, it adopts a threshold-based prediction model, which does not allow nonlinear relationships between features and class outputs. Nevertheless, since PIPE2 is well benchmarked [104], we compared our newly proposed algorithm with it.

Another well-established sequence-based method involves the use of a Support Vector Machine (SVM) with a Pairwise String Kernel [88, 51, 136, 121, 43, 9]. They encode a PPI pair into a feature vector composed by the co-occurrence of k-mer (a sequence of k residues) and train the SVM to predict if a protein pair can interact. For example, assume $k = 3$, a selected feature could be the number of counts of how often the 3-mers, say WTG and LGA co-occur in a protein pair along the entire sequence. Since all possible 3-mers are considered, the feature space could be as large as $20^3 \times 20^3$ (i.e. 64 millions) [47]. With a SVM, even with such a high dimensionality, by using the kernel trick, neither computing nor storing the feature vector is needed. As no feature vectors are computed, in spite of achieving satisfactory prediction performance, it is hard to use SVM results to reveal or interpret why the feature space leads to its good performance. Thus, since the feature space is hardly interpretable, not much biological knowledge can be gained. Hence, to overcome this hurdle encountered when using a SVM is another key motivation of our proposed method. It should be noted that it is possible to generalize k-mer counting strategies allowing for gaps and mismatches [78]. However, these methods still do not allow a variable length. For example, if k is set to be 5, these methods would still consider all the 5-mers, while in WeMine-P2P, there could be 5-mers, 6-mers and 7-mers. In WeMine-P2P, we utilize the locally conserved sequence pattern clusters [143, 77] and their co-occurrence [74] to obtain biologically realistic and interpretable features that are flexible in pattern length while allowing variants. Experiments showed that our prediction results based on these features are comparable to those achieved by the SVM with Pairwise String Kernel approaches. In addition, the presence of concrete feature values makes the feature analysis of our models (and the subsequent biological interpretation) easier for biologists, compared to the SVM with Pairwise String Kernel approaches, which have no concrete features and thus make feature analysis (and the subsequent biological interpretation) of the models difficult.

2.4.4 Short Linear Motifs mediating Protein-Protein interaction

Short Linear Motifs (SLiMs) [29, 93], also known as Linear Motifs (LMs) or Eukaryotic Linear Motifs (ELMs) [26] or minimotifs, are conserved [107] and short protein sequences (generally 2-8 residues in length [107], <10 residues [90]) that mediate Protein-Protein interaction via interacting with (the globular domains of) the same and/or other proteins [24]. The key property of SLiMs is their linearity, which means three-dimensional arrangement is not required to bring distant amino acids together to make the recognizable unit [59]. The conservation of SLiMs varies, where some are highly conserved while others are not [59]. The SLiMs that are not conserved can easily evolve to cater for different types of Protein-Protein interaction [90].

Chapter 3

Pattern-Directed Aligned Pattern Clustering

3.1 Introduction

As described in Chapter 2, for protein and DNA to actually exert their biological functions, they have to participate in biological mechanisms, in which they interact with others via their functional regions. Therefore, the identification of functional regions with mutations from Protein and DNA is vitally important in bioinformatics. Such knowledge if spotted effectively can reveal the crucial mutation hotspots [134], not only enabling us to have a better scientific understanding but also to help the design of new drugs [142, 134].

Up-to-date, domain annotation [33] is one approach to identify functional regions from sequences but it needs to leverage existing databases (such as PFam [34]) or profile hidden markov models [35]. Nevertheless, functional regions not recorded in existing databases or too distinct from the recorded ones cannot be identified. For de novo discovery, motif discovery [37] such as the most popular software MEME [7] locates and aligns locally homologous sub-sequences to obtain a position weight matrix (PWM) [151] which is a fixed-length representation model whereas protein functional region size varies. It thus requires computational expensive exhaustive search to obtain a PWM [151] with width of optimal range.

To overcome such an impasse, this thesis proposes a new algorithm, Pattern-Directed Aligned Pattern Clustering (PD-APCn) [127, 126], which can: a) use a systematic process to determine the representation model width adaptively from data without exhaustive

search; b) discover functional regions with mutations. Experiments on synthetic datasets with different sizes and noise levels showed that PD-APCn [127, 126] outperforms MEME [7] with much higher recall and Fmeasure and computational speed 665 times faster than MEME. When applied to the Cytochrome C and Ubiquitin families, PD-APCn found all key binding sites within the APCs.

3.2 Method

There are two phases in PD-APCn. Given a set of sequences, Phase I is for the discovery of seed patterns leveraging the pattern discovery algorithm [144] based on a suffix tree [5]. An address table is then constructed from the seed patterns. The seed patterns are then extended via the address table to obtain a set of extended seed patterns. Given a set of seed patterns, Phase II of PD-APCn is to initiate and expand the APCs [77, 143] via a new procedure known as APC growing. Figure 3.1 provides a system overview.

PD-APCn is based on two important concepts. The first is the introduction of the breakpoint (Fig. 3.2). We should keep in mind that some mutated patterns (when fragmented) could not be discovered by the pattern discovery algorithm (PDA) [144] since the frequency of occurrences of the entire mutational pattern is too low. In Fig. 2(a) the data space, a pattern ACGGTT in the data space occurs 3 times over 5 sequences. However, its mutated variants ACGCTT and ACGATT, with a single substitution mutation, occur only once and thus cannot be discovered statistically as patterns. Nevertheless, the sub-patterns ACG and TT may still have high frequency of occurrences (if functional), and thus they can still be discovered as patterns. Hence, if we have the address location of the sub-patterns ACG and TT, we consider the mutation spot between them (say C and A) as a breakpoint. By jumping over it the mutated variants ACGCTT and ACGATT can be discovered. In a like manner, Fig. 3.2(b) and (c) illustrate the finding of the insertion and deletion mutations through the breakpoints respectively.

The second concept of PD-APCn is the seed pattern extension introduced to increase the coverage of the growing APC. We observed that the width of seed patterns is inherent in data and should not be affected by the algorithmic process and/or the width parameters. As shown in Fig. 3.3(a), with seed width = 3, we apply the same procedure of jumping over a breakpoint and obtain a full coverage. When the seed width is changed to 4 (Fig. 3.3(b)), the same full coverage is obtained, showing pattern width adaptation without exhaustive search.

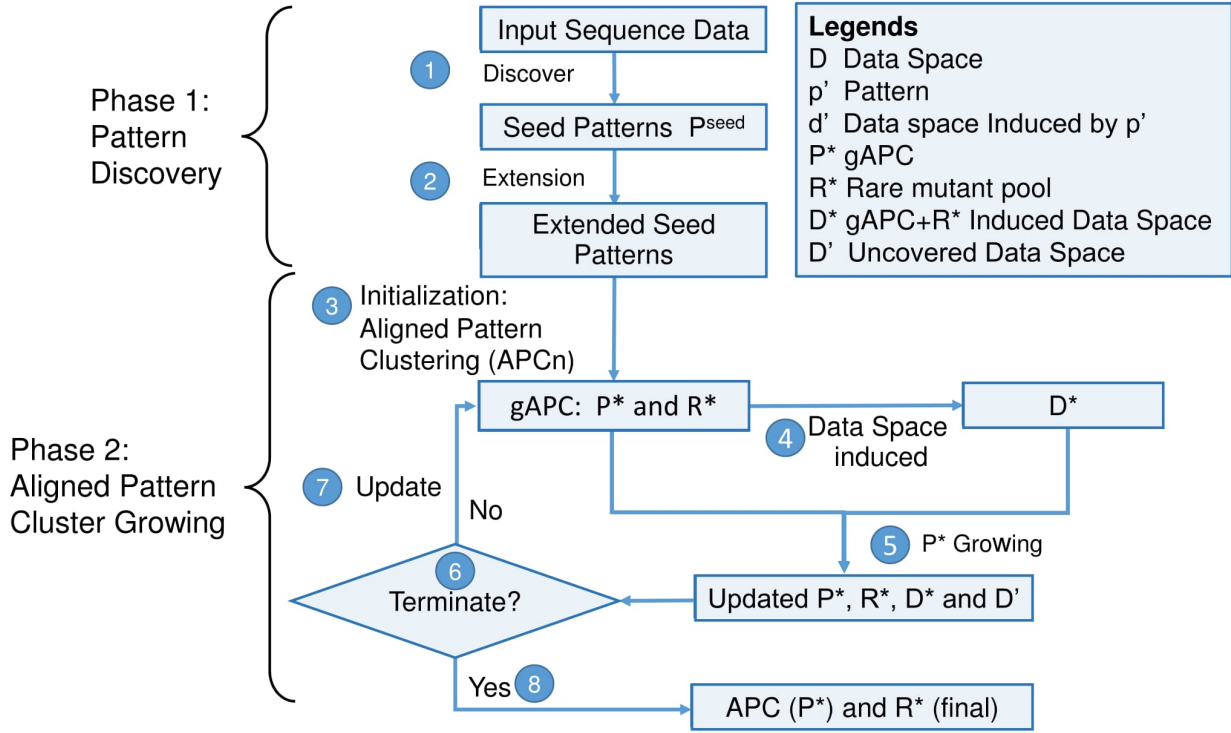


Figure 3.1: An overview of the PD-APCn with the workflow given in circled steps. Phase I: Pattern Discovery. (1) Obtain a set of seed patterns with seed width (preferably small) via the PDA [144] based on a suffix tree [5]. (2) Extend the seed patterns to their superpatterns over the breakpoint gaps to obtain a set of extended seed patterns. Phase II: Growing of gAPCs. (3) Obtain a seed APC (gAPC) from the extended seed patterns. Specifically, the top extended seed pattern is initially considered as a gAPC with only one pattern. Within each gAPC C^* , we denote the patterns (whose support no smaller than $min_{Support}$) as P^* and the rare mutational patterns (whose support smaller than $min_{Support}$) as R^* . (4) Induce data space D from P^* and R^* via the suffix tree (5) For the next extended seed pattern p' , if p' is found significantly similar to the patterns in a gAPC C^* , and its support is no smaller than $min_{Support}$, include it in P^* , update P^* , D^* and D . if p' is found significantly similar to the patterns in a gAPC C^* , and its support is smaller than $min_{Support}$, include it in R^* , update R^* , D^* and D . Otherwise, p' is considered as a new gAPC with only one pattern. (6) Check terminating condition (if a specified amount of extended seed patterns are reached). (7) If not terminated, conduct next run. (8) Consider a gAPC as a final model, which is composed of APC (P^*) and R^* . Rank all final models based on their support. Output the final models with high ranking.

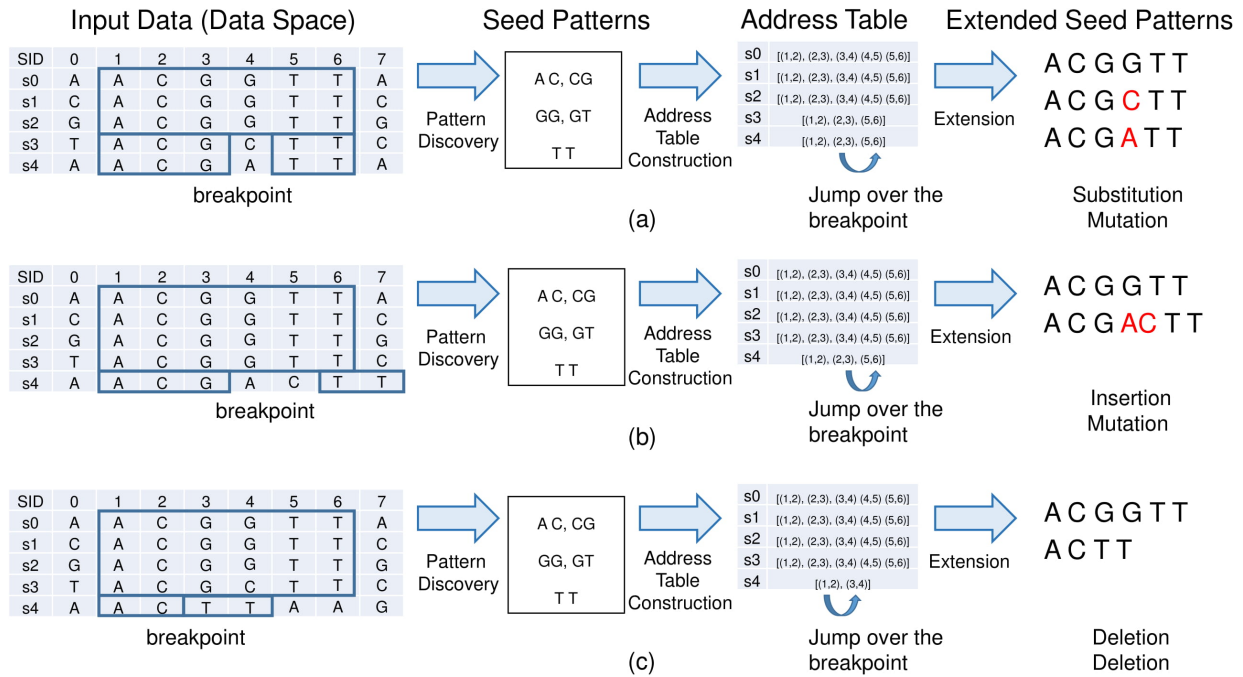


Figure 3.2: This figure illustrates how the concept of pattern breakpoint is used for discovering patterns with 3 types of mutations: (a) substitution, (b) insertion and (c) deletion. Seed Patterns (with seed width=2, $min_{Support} = 5$) are first discovered from the input data (data space). An address table is then constructed from the occurrence of the discovered seed patterns. By jumping over the breakpoints between the subpatterns, a set of extended seed patterns, encompassing the rare mutational patterns, can be discovered, via the process called extension.

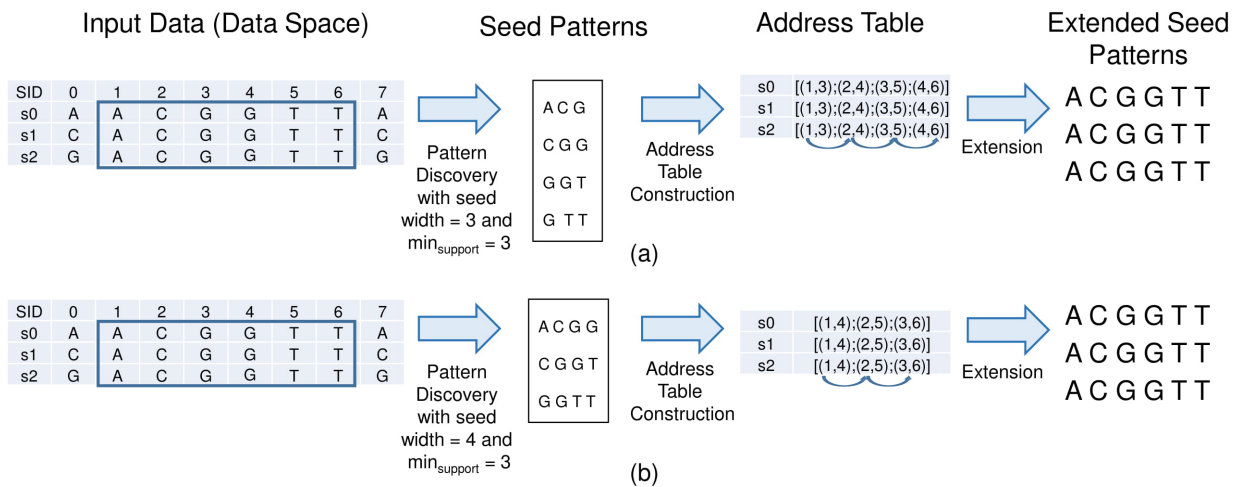


Figure 3.3: Extension of seed patterns to adaptively determine the representation model width. Seed patterns are first discovered from the input data (data space), with (a) seed width = 3, $\min_{Support} = 3$; and (b) seed width = 4, $\min_{Support} = 3$. An address table is then constructed from the occurrence of the discovered patterns. By jumping over the breakpoints between the pattern occurrence, a set of extended seed patterns is discovered. We observe that the set of extended seed patterns obtained in (a) and (b) respectively are the same, showing that the representation model width could be obtained from data adaptively without exhaustive search.

3.2.1 Problem Definition

Give a set of sequences $S = \{s_k | k = 1, 2, \dots, |S|\}$, a positive integer $w_{seed} \in \mathbb{Z}_+$ to determine the width of seed patterns, a positive integer $gap_{break} \in \mathbb{Z}_+$ to control the breakpoint gap, a real-valued similarity threshold $min_{similarity} \in \mathbb{R}$ to cluster patterns, we would like to find a set of aligned pattern clusters (APCs) $\mathbb{C} = \{C^l | l = 1, \dots, |\mathbb{C}|\} = \{C^1, C^2, \dots, C^{|\mathbb{C}|-1}, C^{|\mathbb{C}|}\}$. For details about APCs, please refer to section 2.2.3.

3.2.2 Input Sequence Data

Referring to section 2.2.3, let Σ be a set of alphabets $\{e_1, e_2, \dots, e_{|\Sigma|}\}$. A sequence $s = s_1 s_2 \dots s_{|s|-1} s_{|s|}$, where each $s_i \in \Sigma$ and s is of length $|s|$. Let S be a set of sequences, i.e. $S = \{s_k | k = 1, 2, \dots, |S|\}$. A set of input protein sequences is shown in Fig. 3.4(a).

3.2.3 Step 1: Seed Pattern Discovery

Leveraging the pattern discovery algorithm (PDA) [144] based on a suffix tree [5] (section 2.2.3), we can discover patterns with any width specified, locate the pattern occurrence, and count the pattern support. Hence, we can obtain a set of patterns to serve as seeds efficiently. The seed patterns discovered are then ranked according to their support from highest to lowest. Such crucial information can later assist in finding the breakpoints where mutated patterns can be identified. It should be noted that when we use the pattern discovery algorithm (PDA) [144] in this chapter, we turn off the delta-close redundancy and statistical non-induce pruning. Here we provide more definitions.

Occurrence

A sequence \bar{s} occurs in a sequence s if and only if \bar{s} is a subsequence of s , i.e. $\exists i$ such that $\bar{s} = s[i, i + |\bar{s}| - 1]$, where $1 \leq i \leq |s| - |\bar{s}| + 1$. It is also equivalent to saying that \bar{s} occurs at the position i in s . Hence, given a sequence segment \bar{s} and a sequence s , the occurrence of \bar{s} in s is defined as:

$$Occurrence(\bar{s}, s) = \begin{cases} 1, & \text{if } \bar{s} \text{ occurs in } s \\ 0, & \text{otherwise} \end{cases} \quad (3.1)$$

SID	0	1	2	3	4	5	6	7	8	9	10	11	12
s0	V	H	W	C	A	Q	H	G	C	C	A	G	H
s1	I	P	C	A	Q	H	G	C	G	G	C	A	G
s2	V	C	A	Q	C	H	G	C	M	C	A	G	Y
s3	E	C	A	Q	H	G	T	T	C	C	A	G	E
s4	V	A	C	A	Q	A	A	H	G	C	C	A	G
s5	K	C	A	Q	H	G	C	C	A	G	R	A	G
s6	A	Q	R	C	A	Q	H	G	C	A	C	A	G
s7	F	V	C	A	Q	H	G	C	T	C	A	G	F
s8	Y	S	F	G	C	A	Q	H	G	C	C	A	G
s9	H	J	C	A	Q	H	G	C	C	A	G	E	R

(a) Input Data

PID	0	1	2	3	4	5	6	7	8	9	10	11
p0	C	A	Q	H	G	C	C	A	G			
p1	C	A	Q	H	G	C	C	C	C	A	G	
p2	C	A	Q	C	H	G	C	M	C	A	G	
p3	C	A	Q	H	G	T	T	C	C	A	G	
p4	C	A	Q	A	A	H	G	C	C	A	G	
p5	C	A	Q	H	G	C	C	A	G			
p6	C	A	Q	H	G	C	A	C	A	G		
p7	C	A	Q	H	G	C	T	C	A	G		
p8	C	A	Q	H	G	C	C	A	G			
p9	C	A	Q	H	G	C	C	A	G			

(c) PD-APCn Phase I Output:
A set of Extended Seed Patterns



(b) Top Position Weight Matrix (PWM) discovered by MEME (with optimum width found to be 7)

AP: Aligned Pattern														
C	A	Q	-	-	H	G	-	-	C	-	-	C	A	G
C	A	Q	C	-	H	G	-	-	C	M	-	C	A	G
C	A	Q	-	-	H	G	-	-	C	G	G	C	A	G
C	A	Q	-	-	H	G	-	-	C	-	T	C	A	G
C	A	Q	-	-	H	G	-	-	C	A	-	C	A	G
C	A	Q	-	-	H	G	T	T	C	-	-	C	A	G
C	A	Q	A	A	H	G	-	-	C	-	-	C	A	G

RMP: Rare Mutant Pattern

(d) PD-APCn Phase II Output : An APC, composed of Aligned Patterns and Rare Mutant Patterns

Figure 3.4: Illustrative examples of results obtained from MEME [7] and PD-APCn. (a) Input data, a set of sequences, with the pattern “CAQHGCCA” highlighted in orange, with the mutations highlighted in green. (b) The top Position Weight Matrix (PWM) by MEME [7] on this dataset. (c) A set of extended seed patterns obtained from this dataset (with seed width = 3 and breakpoint gap = 3) by PD-APCn Phase I. Sequence in yellow shade are patterns, i.e. patterns whose support being no smaller than $\min_{Support}$, while those in blue shade are mutants with one occurrence. (d) An Aligned Pattern Cluster (APC) obtained by PD-APCn Phase II, where the APC is composed of an aligned pattern and several rare mutant patterns.

Support

Given a sequence \bar{s} , and a set of sequences S , the support of \bar{s} over S is defined as the number of sequences in S in which \bar{s} occurs. Formally, we write

$$Support(\bar{s}, S) = \sum_{s_k \in S} Occurrence(\bar{s}, s_k) \quad (3.2)$$

Pattern

Given a set of sequences S , we consider a sequence p as pattern if its support is larger than or equal to a minimum threshold $min_{Support}$, i.e. $Support(p, S) \geq min_{Support}$. It should be noted that as it is a new algorithm, the definition here is not the same as the one in section 2.2.3.

Seed Pattern

We then define a seed pattern \bar{p} as a pattern with a particular width w_{seed} , i.e. $|\bar{p}| = w_{seed}$. Given a set of sequences S , a set of seed patterns P^{seed} could then be discovered from S by the pattern discovery algorithm [144] via setting w_{seed} and $min_{Support}$, i.e. $P^{seed} = \{\bar{p}^i | i = 1, \dots, |\bar{P}|\} = \{\bar{p}^1, \bar{p}^2, \dots, \bar{p}^{|\bar{P}|}\}$.

Rare Mutant Pattern

Given a set of sequences S and a set of Patterns P , we consider a sequence r as a rare mutant pattern if its support being lower than a minimum threshold $min_{Support}$, i.e. $Support(p, S) < min_{Support}$ and is found to be significantly similar to the patterns in P , i.e. $ALIGN(P, r) [77, 143] \geq min_{Similarity}$.

3.2.4 Step 2: Seed Pattern Extension

Leveraging the PDA [144] based on a suffix tree [5], given a seed pattern \bar{p}^j , we can retrieve the sequences in which \bar{p}^j occurs and its occurrence positions. For example, as shown in Fig. 3.2(a), the occurrence of ACGGTT over s1 is (1,6). Hence, an address table mapping a sequence s_k to the occurrence of seed patterns on itself can be constructed.

Address Table

Given a sequence s_k , and a set of seed patterns P^{seed} , a function H is defined as follows.

$$H(s_k, P^{seed}) = \{(o_1^k, t_1^k), (o_2^k, t_2^k), \dots, (o_{n_k}^k, t_{n_k}^k)\} \quad (3.3)$$

where o_j^k is the position that a seed pattern $\bar{p}^j \in P^{seed}$ occurs in s_k , t_j^k is the ending position, $\forall j = 1, 2, \dots, n_k$, and n_k is the number of seed patterns occurring in s_k . For example, as shown in Fig. 3.2(a), $H(s_3, \{AC, CG, GG, GT, TT\}) = \{(1,2), (2,3), (3,6)\}$. An address table is constructed by applying function H to every $s_k \in S$.

Breakpoint Gap

Given two pattern occurrences, (o_i^k, t_i^k) and (o_{i+1}^k, t_{i+1}^k) , we define the gap between them as

$$gap_{(o_i^k, t_i^k), (o_{i+1}^k, t_{i+1}^k)} = o_{i+1}^k - t_i^k - 1 \quad (3.4)$$

Note that two pattern occurrences, (o_i^k, t_i^k) and (o_{i+1}^k, t_{i+1}^k) could be merged into one pattern occurrence (o_i^k, t_{i+1}^k) , if $gap_{(o_i^k, t_i^k), (o_{i+1}^k, t_{i+1}^k)} \leq gap_{break}$, where gap_{break} is a non-negative integer defined by users. Hence, $gap_{(o_i^k, t_i^k), (o_{i+1}^k, t_{i+1}^k)}$ is a breakpoint gap if $gap_{(o_i^k, t_i^k), (o_{i+1}^k, t_{i+1}^k)} \leq gap_{break}$.

Extended Seed Pattern

By merging pattern occurrences, the seed patterns are extended to their superpatterns, allowing the identification of rare mutant patterns such as those with frameshifts. For example, as illustrated in 3.4(c), “CAQHGC” has a width of 6 occurring at position 2 on s1, i.e. (2,7), and “CAG” has a width of 3 occurring at position 10 on s1, i.e. (10,12). With $gap_{break} = 2$, these two occurrences would be grouped into one occurrence, i.e. (2,12), allowing the identification of the rare mutant pattern “CAQHGC²CAG”. As mentioned, such rare mutant patterns if spotted effectively can reveal the crucial mutation hotspots [134], not only enabling us to have a better scientific understanding but also to help the design of new drugs [142, 134]. We applied such operation on the address table constructed to obtain a set of extended seed patterns P_{ext}^{seed} . Afterwards, all the extended seed patterns are ranked according to their statistical significance [144].

3.2.5 APC Growing

After the discovery of a set of extended seed patterns P^{seed} in Phase 1 (Fig. 5.1), an iterative APC growing process (steps (3) to (8)) directed by the extended seed patterns follows in Phase 2 as below. Here we first define APC. For details, please refer to section 2.2.3. Let a set of APC be defined as:

$$\mathbb{C} = \{C^l | l = 1, \dots, |\mathbb{C}|\} = \{C^1, C^2, \dots, C^{|\mathbb{C}|-1}, C^{|\mathbb{C}|}\}$$

and let an APC be defined as,

$$C^l = \text{ALIGN}(\mathbb{P}^l), \tag{3.5}$$

$$= \begin{pmatrix} s_1^1 & s_2^1 & \dots & s_n^1 \\ s_1^2 & s_2^2 & \dots & s_n^2 \\ \vdots & \vdots & \vdots & \vdots \\ s_1^m & s_2^m & \dots & s_n^m \end{pmatrix}_{m \times n} = \begin{pmatrix} p^1 \\ p^2 \\ \vdots \\ p^m \end{pmatrix}, \tag{3.6}$$

$$= (p^1 \quad p^2 \quad \dots \quad p^m). \tag{3.7}$$

where $s_j^i \in \Sigma \cup \{-\}$ is a pattern p^i with a new column index j . Each of the $|\mathbb{P}^l| = m$ patterns in the rows of C^l is of length $|C^l| = n$.

Step 3: Initialization of gAPC

Obtain a seed APC (gAPC) from the extended seed patterns. Specifically, the top extended seed pattern is initially considered as a gAPC with only one pattern. Within each gAPC, we denote the patterns (with support no smaller than $min_{Support}$) as P^* and the rare mutant patterns (with support smaller than $min_{Support}$) as R^* . It should be noted that initialization of gAPC is conducted only in the first run of this step.

Step 4: Induce data space D^* from P^* and R^*

We denote data space D^* as a set of sequences containing the patterns in P^* and R^* , as well as data space D' as a set of sequences not containing any patterns in P^* , i.e. the data space uncovered yet. Via the suffix tree constructed by PDA [144], such an operation is efficient.

Step 5: P* and R* growing

For the next extended seed pattern p' , if p' is found significantly similar to the patterns in a gAPC C^* , and its support is no smaller than $min_{Support}$, include it in P^* , update P^* , D^* and D . if p' is found significantly similar to the patterns in a gAPC C^* , and its support is smaller than $min_{Support}$, include it in R^* , update R^* , D^* and D . Otherwise, p' is considered as a new gAPC with only one pattern. It should be noted that the similarity between p' and the patterns in a gAPC C^* is computed by ALIGN ($P^* \cup R^* \cup p'$) [77, 143].

Step 6: Check terminating condition

We check if we have reached the terminating condition, i.e. if a specified amount of extended seed patterns are reached.

Step 7: Continue or terminate

If termination condition is reached, we end the growing process, i.e. jump to step 8. Otherwise, we continue the algorithm, i.e. back to step 3 but skip the initialization.

Step 8: Output the final models

At termination, each gAPC C^* will be composed of P^* and R^* and is considered as the final model. Rank all the final models by their support and output those with high ranking.

3.3 Experiments and Results

3.3.1 Design of Experiments

To demonstrate the effectiveness of PD-APCn, we designed and conducted synthetic experiments to evaluate its performance with respect to how effective it is at discovering and locating the conserved functional regions scattered in a dataset with various conserved and mutational patterns synthetically generated. Three sets of synthetic data of different number of sequences subjecting to different mutations were generated randomly. We used them to compare PD-APCn with other methods quantitatively through a set of metrics

following the previous work [55]. After experiments on synthetic datasets, we applied PD-APCn to two real protein sequence datasets, Cytochrome c and Ubiquitin, obtained from Pfam [34].

3.3.2 Synthetic Dataset Preparation

In this study, for the purpose of quantitative evaluation, three synthetic protein sequence datasets were generated. Dataset 1 is a synthetic dataset composed of 500 protein sequences, generated under the following procedure. First, 500 protein sequences were randomly generated at a random length of 50 to 150 under a uniform distribution of the 20 amino acids. Second, a protein segment with 30 amino acids “MKCSQCHTVEKGGKHK-TGPNLHGLFGRKTG” extracted from Human Cytochrome C (UniProt KB ID: P99999, positions 12 to 41) was used as the conserved pattern extracted from a real biological dataset. Third, this pattern was implanted at randomly generated positions among the 500 protein sequences with its position in all sequences recorded. To simulate mutational degeneracy, during the insertion of the conserved pattern, each of its position would undergo 5% chance of substitution, insertion and deletion mutation. Dataset 2 is a synthetic dataset composed of 1000 protein sequences, generated similar to the procedure used for generating Dataset 1 but double in size. Dataset 3 is a synthetic dataset composed of 2000 protein sequences. The first 1000 sequences were generated by the same procedure used for generating Dataset 1. An additional 1000 protein sequences were randomly generated with variable length of 50 to 150 under an uniform distribution of the 20 amino acids. They were considered as noise sequences.

3.3.3 Evaluation of Experiments on Synthetic Datasets

We evaluated PD-APCn with MEME [7] and GLAM2 [38] via these three datasets, where the conserved region positions are a priori known and considered as the ground-truth. The discovered conserved regions outputted by algorithms could then be compared with the ground-truth quantitatively. Hence, as illustrated by a previous work [55], True Positive (TP), False Positive (FP) and False Negative (FN) could be defined. TP refers to the conserved region positions overlapping with the predicted positions. FP refers to the predicted positions not overlapping with any conserved region positions. Also, any predicted positions on the noise protein sequences are considered as FP. FN refers to the conserved region positions not overlapping with any predicted positions. Fig. 3.5 provides a graphical illustration of the definition of TP, FP and FN. Based on TP, FP and FN, we could define Precision, Recall and Fmeasure, as below, illustrated by the previous work [55].

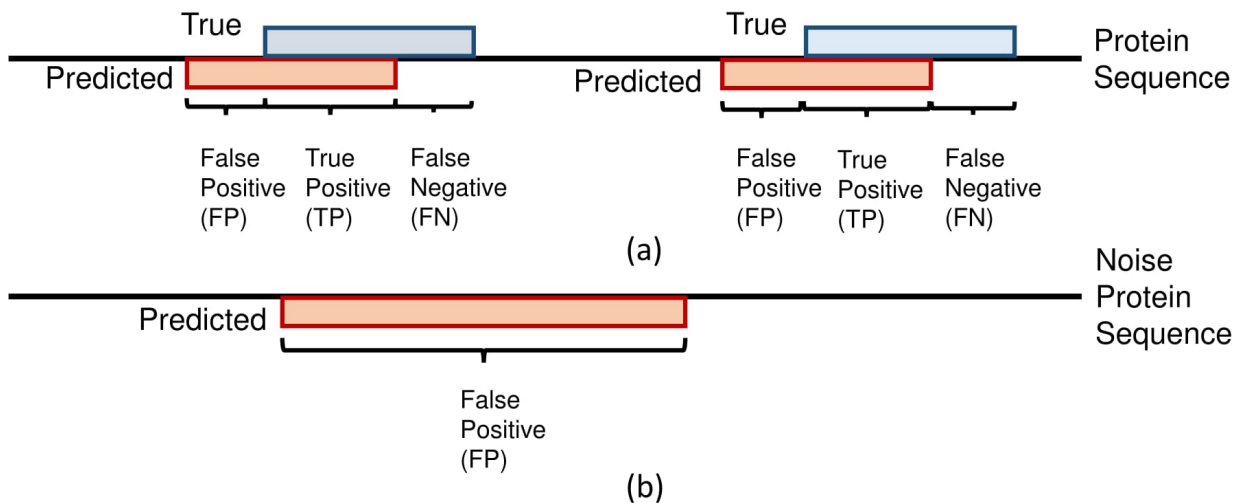


Figure 3.5: An illustration of the definition of True Positive (TP), False Positive (FP) and False Negative (FN) for the quantitative evaluation of the predicted conserved regions. The true and predicted patterns are illustrated as blue and red blocks respectively in the figure. In our experiments on synthetic datasets, the conserved region positions on a protein sequence were a priori known, as illustrated as blue blocks in the figure.

$$Precision = \frac{nTP}{nTP + nFP} \quad (3.8)$$

$$Recall = \frac{nTP}{nTP + nFN} \quad (3.9)$$

$$Fmeasure = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (3.10)$$

where nTP refers to the total number of TP, nFP refers to the total number of FP, and nFN refers to the total number of FN. Also, if both $Precision$ and $Recall$ are zero, $Fmeasure$ is defined as zero [55].

In our experiments, we applied MEME, GLAM2 and PD-APCn to discover the conserved regions from the input protein sequences. MEME [7] is a popular Protein or DNA motif discovery algorithm for bioinformatics scientists. Hence it was chosen for comparison in our experiments on the synthetic datasets. The version we adopted was 4.11.04, released on April, 2017 [7]. In our experiments, we had three options for MEME, by setting the

Table 3.1: Performance evaluation of PD-APCn on Dataset 1 (500 sequences)

	Precision	Recall	Fmeasure
GLAM2 [38] (nMotifs=1)	0.37840	1.00000	0.54904
GLAM2 [38] (nMotifs=2)	0.34745	1.00000	0.51572
GLAM2 [38] (nMotifs=3)	0.33325	1.00000	0.49991
MEME [7] (nMotifs=1)	0.99839	0.49630	0.66301
MEME [7] (nMotifs=2)	0.99261	0.77936	0.87315
MEME [7] (nMotifs=3)	0.99269	0.78816	0.87868
PD-APCn ($w_{seed}=3, gap_{break}=2$)	0.96348	0.89905	0.93015
PD-APCn ($w_{seed}=3, gap_{break}=3$)	0.96335	0.91655	0.93942

Table 3.2: Parameter investigation of PD-APCn on Dataset 1 (500 sequences)

	Precision	Recall	Fmeasure
PD-APCn ($w_{seed}=3, gap_{break}=2$)	0.96348	0.89905	0.93015
PD-APCn ($w_{seed}=4, gap_{break}=2$)	0.99584	0.82937	0.90501
PD-APCn ($w_{seed}=5, gap_{break}=2$)	0.99948	0.76369	0.86581
PD-APCn ($w_{seed}=3, gap_{break}=3$)	0.96335	0.91655	0.93942
PD-APCn ($w_{seed}=4, gap_{break}=3$)	0.99589	0.84077	0.91178
PD-APCn ($w_{seed}=5, gap_{break}=3$)	0.99948	0.77169	0.87094

number of motifs to search to be 1 (nMotifs=1) or 2 (nMotifs=2) or 3 (nMotifs=3). The other MEME [7] parameters remained default. GLAM2 [38] is an algorithm that is famous for gap pattern discovery. Thus it was chosen for comparison in our experiments on the synthetic datasets. The version we adopted was also 4.11.04, released on April, 2017 [7]. In our experiments, the GLAM2 parameters remained default. For the PD-APCn algorithm, we varied the setting of seed (pattern) width (w_{seed}) and also the breakpoint gap (gap_{break}) to investigate its robustness. All experiments were conducted on a laptop computer (i7-4700HQ CPU 2.4GHz, 16.0 GB RAM).

3.3.4 Experimental Results Analysis on Dataset 1 (500 sequences)

Dataset 1 is a synthetic dataset composed of 500 protein sequences containing a mutated protein segment with thirty amino acids. It does not contain noise sequences, and thus is a relatively simple dataset among the three. We applied MEME [7], GLAM2 [38] and PD-APCn on this dataset. For MEME, we had three parameter settings, i.e. the number of motifs to search to be 1 (nMotifs=1) or 2 (nMotifs=2) or 3 (nMotifs=3). For GLAM2, we

adopted the default parameter settings and investigated the top 3 motifs. For PD-APCn, we fixed the seed (pattern) width (w_{seed}) to be 3 and varied the breakpoint gap (gap_{break}) to be 2 and 3. Also, in step 1, we only used the top 6% of the seed patterns. In step 5, the similarity threshold was set as 0.05. In step 6, all the extended seed patterns needed to be reached.

Table 3.1 summarizes the experimental results on Dataset 1. We observed that MEME [7] obtained a high precision but a low recall. For MEME (nMotifs=1), the precision was 0.99839 but the recall was merely 0.49630, indicating that a significant portion of patterns were not discovered. For MEME (nMotifs=2), the precision increased to 0.99261 and the recall also increased to 0.77936. For MEME (nMotifs=3), the precision further increased to 0.99269 and the recall further increased to 0.78816, but on both cases the marginal increase was lower. We observed that GLAM2 obtained an extremely high recall, but an extremely low precision, leading to low fmeasure. For GLAM2 (nMotifs=1), GLAM2 (nMotifs=2), GLAM2 (nMotifs=3), the recall obtained was 1.00000, but the precision obtained was respectively 0.37840, 0.34745, 0.33325, leading to low Fmeasure 0.54904, 0.51572, 0.49991. For PD-APCn, it obtained a satisfactory level of both precision and recall, and thus obtained a higher level of Fmeasure, outperforming MEME and GLAM2 in this dataset. For PD-APCn ($w_{seed}=3$, $gap_{break}=2$), the obtained precision was 0.96348 and the recall was 0.89905. For PD-APCn ($w_{seed}=3$, $gap_{break}=3$), the obtained precision slightly decreased to 0.96335 but the recall increased to 0.91655, indicating that a significant portion of patterns were discovered. For both cases, PD-APCn obtained a slightly lower precision but a significantly higher level of recall, thus leading to a higher level of Fmeasure.

We next investigated the effects of the parameter settings on PD-APCn on Dataset 1. Table 3.2 summarized the experimental results by further setting the w_{seed} to be 4 and 5. We observed that by setting the w_{seed} larger (increasing from 3 to 5), the precision obtained was higher. PD-APCn ($w_{seed}=5$, $gap_{break}=2$) and PD-APCn ($w_{seed}=5$, $gap_{break}=3$) both obtained the highest level of precision as 0.99948, but at the same time obtained the lowest recall as 0.76369 and 0.77169 respectively. We also observed that by setting the gap_{break} from 2 to 3, the recall obtained was higher. We observed this was true not only for PD-APCn ($w_{seed}=3$) but also PD-APCn ($w_{seed}=4$) and PD-APCn ($w_{seed}=5$).

3.3.5 Experimental Results Analysis on Dataset 2 (1000 sequences)

Dataset 2 is a synthetic dataset composed of 1000 protein sequences containing a mutated protein segment with thirty amino acids. It does not contain noise sequences, but is a larger dataset comparing to the size of Dataset 1. We applied MEME [7], GLAM2 [38]

Table 3.3: Performance evaluation of PD-APCn on Dataset 2 (1000 sequences)

	Precision	Recall	Fmeasure
GLAM2 [38] (nMotifs=1)	0.46781	1.00000	0.63742
GLAM2 [38] (nMotifs=2)	0.41305	1.00000	0.58462
GLAM2 [38] (nMotifs=3)	0.35262	1.00000	0.52139
MEME [7] (nMotifs=1)	0.97967	0.39232	0.56028
MEME [7] (nMotifs=2)	0.97922	0.84919	0.90958
MEME [7] (nMotifs=3)	0.97930	0.85249	0.91151
PD-APCn ($w_{seed}=3, gap_{break}=2$)	0.96541	0.89065	0.92092
PD-APCn ($w_{seed}=3, gap_{break}=3$)	0.96462	0.91266	0.93792

Table 3.4: Parameter investigation of PD-APCn on Dataset 2 (1000 sequences)

	Precision	Recall	Fmeasure
PD-APCn ($w_{seed}=3, gap_{break}=2$)	0.96541	0.89065	0.92092
PD-APCn ($w_{seed}=4, gap_{break}=2$)	0.99654	0.82580	0.90317
PD-APCn ($w_{seed}=5, gap_{break}=2$)	0.99965	0.75798	0.86220
PD-APCn ($w_{seed}=3, gap_{break}=3$)	0.96462	0.91266	0.93792
PD-APCn ($w_{seed}=4, gap_{break}=3$)	0.99658	0.83731	0.91003
PD-APCn ($w_{seed}=5, gap_{break}=3$)	0.99965	0.76579	0.86723

Table 3.5: Performance evaluation of PD-APCn on Dataset 3 (2000 sequences)

	Precision	Recall	Fmeasure
GLAM2 [38] (nMotifs=1)	0.61117	1.00000	0.75867
GLAM2 [38] (nMotifs=2)	0.59827	1.00000	0.74865
GLAM2 [38] (nMotifs=3)	0.54501	1.00000	0.70551
MEME [7] (nMotifs=1)	0.99898	0.48957	0.65711
MEME [7] (nMotifs=2)	0.99261	0.77936	0.87315
MEME [7] (nMotifs=3)	0.93682	0.83278	0.88426
PD-APCn ($w_{seed}=3, gap_{break}=2$)	0.92997	0.89605	0.91269
PD-APCn ($w_{seed}=3, gap_{break}=3$)	0.93039	0.91266	0.92149

and PD-APCn on this dataset. The initial parameter setting was the same as those used in Dataset 1.

Table 3.3 summarizes the experimental results on Dataset 2. Similar to the results in Dataset 1, PD-APCn obtained a satisfactory level of both precision and recall, and thus obtained a higher level of Fmeasure, outperforming MEME and GLAM2 in this dataset. We also observed that GLAM2 obtained an extremely high recall, but an extremely low precision, leading to low fmeasure. For PD-APCn ($w_{seed}=3, gap_{break}=3$), it obtained the highest Fmeasure as 0.93792 in this dataset. Again, this high recall indicated that a significant portion of patterns were discovered. These results also demonstrated that scaling up the dataset two times larger did not affect the performance of PD-APCn.

We then investigated the effects of the parameter settings on PD-APCn on Dataset 2. Table 3.4 summarized the experimental results by further setting the seed width to be 4 and 5. The observation was consistent. By setting the w_{seed} larger (increasing from 3 to 5), the precision obtained was higher, but the recall was lower. PD-APCn ($w_{seed}=5, gap_{break}=2$) and PD-APCn ($w_{seed}=5, gap_{break}=3$) both obtained the highest level of precision as 0.99948, but at the same time obtained the lowest recall as 0.76369 and 0.77169 respectively. We also observed that by setting the gap_{break} from 2 to 3, the recall obtained was higher, consistently for $w_{seed}=3, w_{seed}=4$) and PD-APCn ($w_{seed}=5$).

3.3.6 Experimental Results Analysis on Dataset 3 (2000 sequences)

Dataset 3 is a synthetic dataset composed of 2000 protein sequences. Among them, 1000 sequences contained a mutated protein segment with thirty amino acids. The remaining sequences were noise sequences. Thus, it is a relatively challenging dataset among the

Table 3.6: Parameter investigation of PD-APCn on Dataset 3 (2000 sequences)

	Precision	Recall	Fmeasure
PD-APCn ($w_{seed}=3, gap_{break}=2$)	0.92997	0.89605	0.91269
PD-APCn ($w_{seed}=4, gap_{break}=2$)	0.99397	0.82580	0.90211
PD-APCn ($w_{seed}=5, gap_{break}=2$)	0.99965	0.75798	0.86220
PD-APCn ($w_{seed}=3, gap_{break}=3$)	0.93039	0.91266	0.92149
PD-APCn ($w_{seed}=4, gap_{break}=3$)	0.99406	0.83731	0.90898
PD-APCn ($w_{seed}=5, gap_{break}=3$)	0.99965	0.76579	0.86723

three datasets. We applied MEME [7], GLAM2 [38] and PD-APCn on this dataset. The initial parameter setting was the same as those used in Dataset 1.

Table 3.5 summarizes the experimental results on Dataset 3. We observed consistently that MEME [7] obtained a high precision but a low recall, indicating a large portion of patterns was not discovered. We also observed that GLAM2 obtained an extremely high recall, but an extremely low precision, leading to low fmeasure. PD-APCn obtained a satisfactory level of precision and recall, and thus a higher Fmeasure, outperforming MEME [7] and GLAM2. This consistent high recall indicated that PD-APCn has discovered a greater significant portion of patterns than MEME. As for the effects of the parameter settings on PD-APCn on this Dataset, Table 3.6 summarized the experimental results on this dataset with respect to the parameter setting of w_{seed} to be 4 and 5. We observed that by increasing w_{seed} from 3 to 5, the precision obtained was higher but the recall was lower.

Fig. 3.6 (a), (b) and (c) shows the top, 2nd and 3rd output of MEME, while Fig. 3.6 (d) shows the top APC outputted by PD-APCn. The top three PWMs outputted by MEME has a width of 15, 8 and 11 respectively. Note that the third one has substantial overlapping with the first two. The top APC (showing only the first 25 patterns) outputted by PD-APCn has a width of 35. It has captured the entire protein segment introduced in Dataset 3, i.e. “MKCSQCHTVEKGGKHKKTGPNLHGLFGRKTG” with 30 amino acids. It is clear here that MEME is much inferior in reflecting aligned protein segment to PD-APCn in this experiment. This explains their differences in their recalls.

3.3.7 Combined Analysis on Datasets 1, 2 and 3

As shown in Tables 3.2, 3.4 and 3.6, the Fmeasure obtained demonstrates its robustness to parameter settings, and also that introducing noise sequences or varying either the w_{seed} or the gap_{break} would affect little the performance of PD-APCn. By setting $w_{seed}=3$

Table 3.7: Runtime comparison of PD-APCn on Datasets 1, 2 and 3

	Dataset 1	Dataset 2	Dataset 3
GLAM2 (nMotifs=3) [38]	202.074s	334.273s	228.779s
MEME [7] (nMotifs=1)	368.401s	2315.512s	15721.029s
MEME [7] (nMotifs=2)	471.633s	2749.722s	17437.620s
MEME [7] (nMotifs=3)	570.683s	3155.81s	18786.427s
PD-APCn ($w_{seed}=3$, $gap_{break}=2$)	4.759s	12.531s	28.104s
PD-APCn ($w_{seed}=4$, $gap_{break}=2$)	5.143s	13.466s	30.309s
PD-APCn ($w_{seed}=5$, $gap_{break}=2$)	5.213s	13.997s	33.232s
PD-APCn ($w_{seed}=3$, $gap_{break}=3$)	4.843s	12.999s	28.232s
PD-APCn ($w_{seed}=4$, $gap_{break}=3$)	5.193s	13.653s	30.454s
PD-APCn ($w_{seed}=5$, $gap_{break}=3$)	5.726s	14.070s	33.696s

and $gap_{break}=3$, PD-APCn obtained high recall and Fmeasure consistently with at a little sacrifice of precision.

In addition to performance, runtime is also an important criterion. Table 3.7 summarized the runtime of GLAM2, MEME and PD-APCn among all parameter settings on all three datasets. It should be noted that as GLAM2 outputted all the top 3 motifs at once, it thus only had one row of record in our experiment. In Dataset 1 (500 protein sequences), MEME took at least 300s while PD-APCn took at most 6s. MEME [7] (nMotifs=3) took 570.683s to complete running to obtain its optimal Fmeasure (0.87868), while PD-APCn (seed width=3, breakpoint gap=3) took a much less time, 4.843s, but obtained an even higher Fmeasure (0.93942). It was a speed up of 117.84X. In Dataset 2 (1000 protein sequences), MEME took at least 2000s while PD-APCn took at most 15s. MEME [7] (nMotifs=3) took 3155.81s to complete running to obtain its optimal Fmeasure (0.91151), while PD-APCn (seed width=3, breakpoint gap=3) took a much less time, 12.299s, but obtained an even higher Fmeasure (0.93792). It was a speed up of 256.59X. In Dataset 3 (2000 protein sequences), MEME took at least 15000s while PD-APCn took at most 34s. MEME [7] (nMotifs=3) took 18786.427s to complete running to obtain its optimal Fmeasure (0.88426), while PD-APCn (seed width=3, breakpoint gap=3) took a much less time, 28.232s, but obtained an even higher Fmeasure (0.92149). It was a speed up of 665.43X.

In addition to computation speed-up, the significance of PD-APCn lies to its ability to discover and locate functional regions that have various types of mutations, including substitution, mutation and deletion. PD-APCn can discover the rare mutant pattern even with a support = 1. The discovery of rare mutant patterns will be significant for personalized medicine in the future.

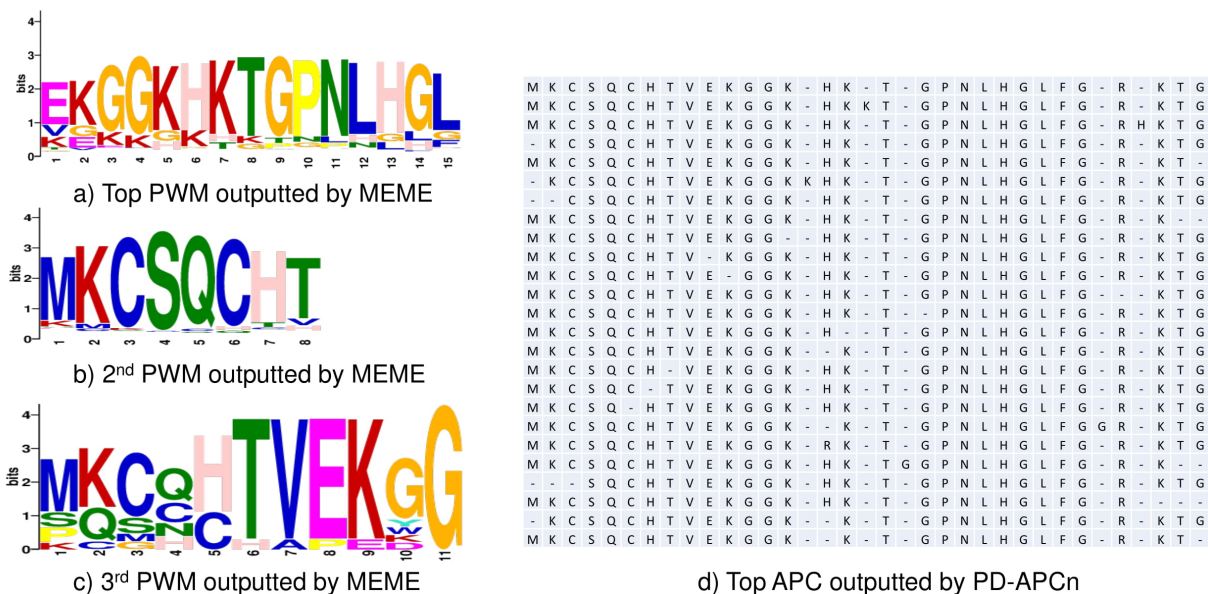


Figure 3.6: A comparison of the outputs by MEME [7] and PD-APCn on Dataset 3 (2000 sequences). (a) The top position weight matrix (PWM) outputted by MEME, with a width of 15. (b) The 2nd PWM outputted by MEME, with a width of 8. (c) The 3rd PWM outputted by MEME, with a width of 11. (d) The top APC outputted by PD-APCn (showing only the first 25 patterns), with a width of 35. It should be noted that the protein segment introduced in Dataset 3 is “MKCSQCHTVEKGGKHKHTGPNLHGLFGRKTG” with 30 amino acids, and it is entirely captured in (d). Further analysis such as functional subgroup discovery by disentanglement [156] will be possible via the APC discovered by PD-APCn.

3.3.8 Real Dataset Preparation and Parameter Setting

In this study, to validate the practical usability of PD-APCn, two real protein sequence datasets were obtained from Pfam [34]. The first dataset is Dataset Cytochrome C downloaded from Pfam (PF00034, Seed, Release 31) on March 15th, 2018. It has 66 sequences, with an average length of 91.11 amino acids. The second dataset is Dataset Ubiquitin downloaded from Pfam (PF00240, Seed, , Release 31) on March 15th, 2018. It has 63 sequences, with an average length of 72.62 amino acids. The parameter setting is as follows. w_{seed} is set as 3. gap_{break} is set as 0, so that the results can be comparable to those obtained in previous studies [77, 143].

3.3.9 Experimental Results Analysis on Dataset Cytochrome C

Cytochrome C is a heme-containing protein [154]. It is an essential component of the electron transport chain in the mitochondria [154], where the heme group plays an important role in accepting and transferring electrons.

The Pfam Hidden Markov Model (HMM) logo of Cytochrome C is shown in Fig. 3.7(a). Applying PD-APCn on the Dataset Cytochrome C, the first three APCs obtained are shown in Fig. 3.7(b), (c) and (d). The 1st APC has covered Cys (C) 14 [8, 14], Cys (C) 17 [8, 14] and His (H) 18 [129, 45]. His (H) 18 [129, 45] forms an axial ligand with the heme from the proximal front, i.e. the proximal heme binding site. Cys (C) 14 [8, 14] and Cys (C) 17 [8, 14] enhance and maintain the axial ligand between His18 and the heme. The 2nd APC has covered Tyr (Y) 97, which provides a hydrophobic environment for the function of Cytochrome C [36]. The 3rd APC has covered Met (M) 80 [129] which forms an axial ligand with the heme from the distal side, i.e. the distal heme binding site. It should be noted that the Pfam Hidden Markov Model (HMM) logo of Cytochrome C, as shown in Fig. 3.7(a), does not clearly indicate Met (M) 80. Fig. 3.8 gives a three-dimensional structure illustration. These results have validated the capability of PD-APCn to discover functional regions in real protein sequences.

3.3.10 Experimental Results Analysis on Dataset Ubiquitin

Ubiquitin plays an important role in a process called ubiquitination, where ubiquitin is attached to a substrate protein. It could either be a single ubiquitin protein or a chain of ubiquitin. To form a chain, an ubiquitin connects to another ubiquitin by binding its C-terminal tail to one of the seven lysine (K) amino acid of its linking partner. The seven

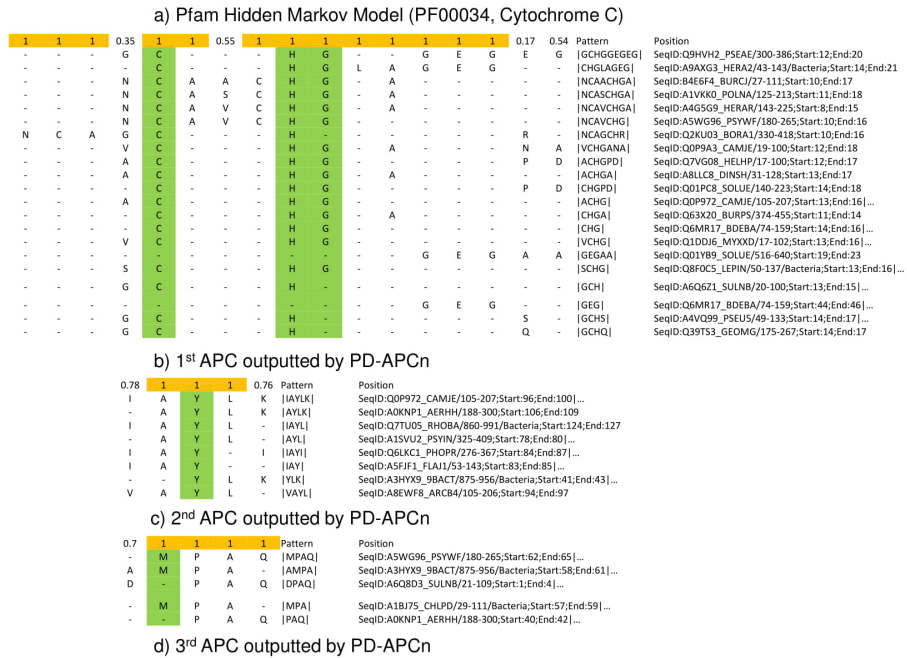
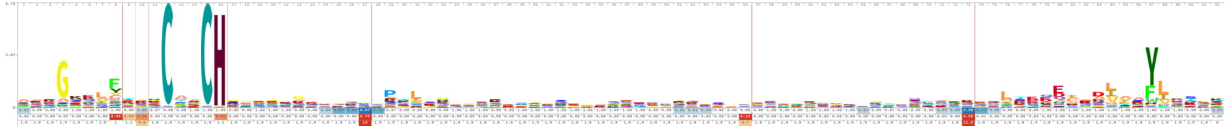


Figure 3.7: An illustration of the APCs outputted by PD-APCn on the Dataset Cytochrome C. (a) The Pfam Hidden Markov Model (HMM) logo of Cytochrome C. Applying PD-APCn on the Dataset Cytochrome C, the first three APCs obtained are shown in Fig. 3.7(b), (c) and (d). The column score is denoted as R1 [77]. The higher the score, the more conserved the column is. (b) The 1st APC outputted by PD-APCn on the Dataset Cytochrome C. It has covered His (H) 18 [129, 45], the proximal heme binding site, as well as Cys (C) 14 [8, 14] and Cys (C) 17 [8, 14] which assist the heme binding (c) The 2nd APC outputted by PD-APCn on the Dataset Cytochrome C. It has covered Tyr (Y), which provides a hydrophobic environment for the function of Cytochrome C [36]. (d) The 3rd APC outputted by PD-APCn on the Dataset Cytochrome C. It has covered Met (M) 80 [129], i.e. the distal heme binding site. These results have validated the capability of PD-APCn to discover functional regions in real protein sequences.

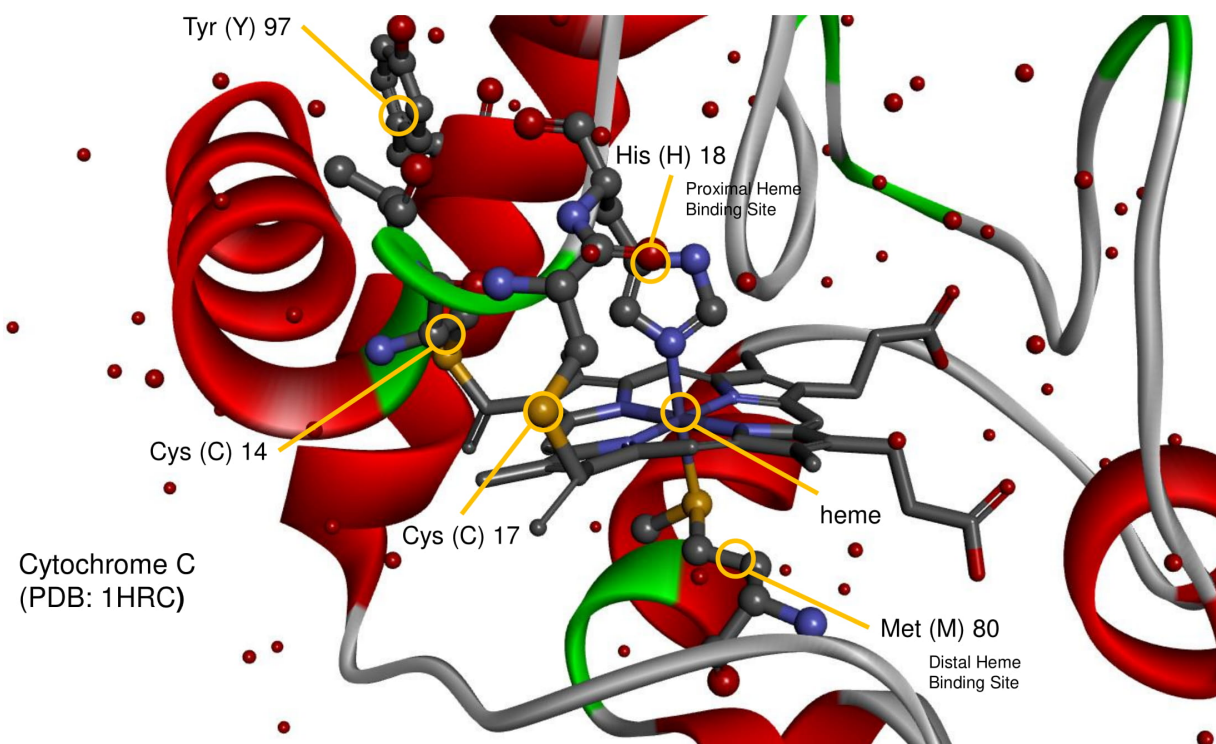


Figure 3.8: A three-dimensional structure of Cytochrome C obtained from Protein Data Bank (PDB) [12] (ID: 1HRC). This figure shows that the binding residues Cys (C) 14, Cys (C) 17, His (H) 18, Met (M) 80, Tyr (Y) 97 are all highlighted with $R1=1$ within our discovered APCs (Fig. 3.7(b)(c)(d)).

lysine (K) are Lys (K) 6, Lys (K) 11, Lys (K) 27, Lys (K) 29, Lys (K) 33, Lys (K) 48 and Lys (K) 63.

The Pfam Hidden Markov Model (HMM) logo of Ubiquitin C is shown in Fig. 3.9(a). Applying PD-APCn on the Dataset Ubiquitin C, the first four APCs obtained are shown in Fig. 3.9(b), (c), (d) and (e). The column score is denoted as R1 [77]. The higher the score, the more conserved the column is. The 1st APC has covered Lys (K) 48 and Lys (K) 63. The 2nd APC has covered Lys (K) 33. The 3rd APC has covered Lys (K) 27, Lys (K) 29 and Lys (K) 33. The 4th APC has covered Lys (K) 6 and Lys (K) 11. Hence, all seven lysine (K) have been covered, where they are important for the formation of ubiquitin chains [137, 25, 130]. Fig. 3.10 gives a three-dimensional structure illustration. These results have further validated the capability of PD-APCn to discover functional regions in real protein sequences.

3.4 Summary

In this chapter, we proposed a new algorithm Pattern-Directed Aligned Pattern Clustering (PD-APCn) [126, 127] to discover and locate functional regions that have various types of mutations, including substitution, mutation and deletion, in protein sequences, represented as APCs. For the rare mutants discovered, such as those with support = 1, it could place them into a hotspot mutant pool. These rare mutants could be important for biomedical research. Also, further analysis such as functional subgroup discovery by disentanglement [156] will be possible via the APC discovered by PD-APCn. It should be noted that these mutants are difficult to discover as revealed by the low recall of MEME [7] in our experiments. Hence, the final APCs obtained by PD-APCn [126, 127] are more stable and robust as it complies to the conditions determined by the more natural sequence structures and functionality inherent in the data. Such phenomena are manifested by the discovery results. It thus resolves a difficult problem of demarcating the size of a conserved region and avoids the exhaustive search of such size parameter to drive for an optional solution.

To evaluate the performance of PD-APCn [126, 127], we generated synthetic datasets with a priori known mutated protein sequence segments implanted. Among all the experiments on the three datasets, where each of them has a different size and noise level, PD-APCn [126, 127] has consistently demonstrated high performance in both effectiveness and efficiency. Comparing with the popular MEME [7], PD-APCn [126, 127] has manifested competitive performance, higher in recall and Fmeasure with significant computational speed up (up to 665x). Through parameter analysis, we demonstrated that PD-APCn [126, 127] has rendered consistently high performance among all datasets given

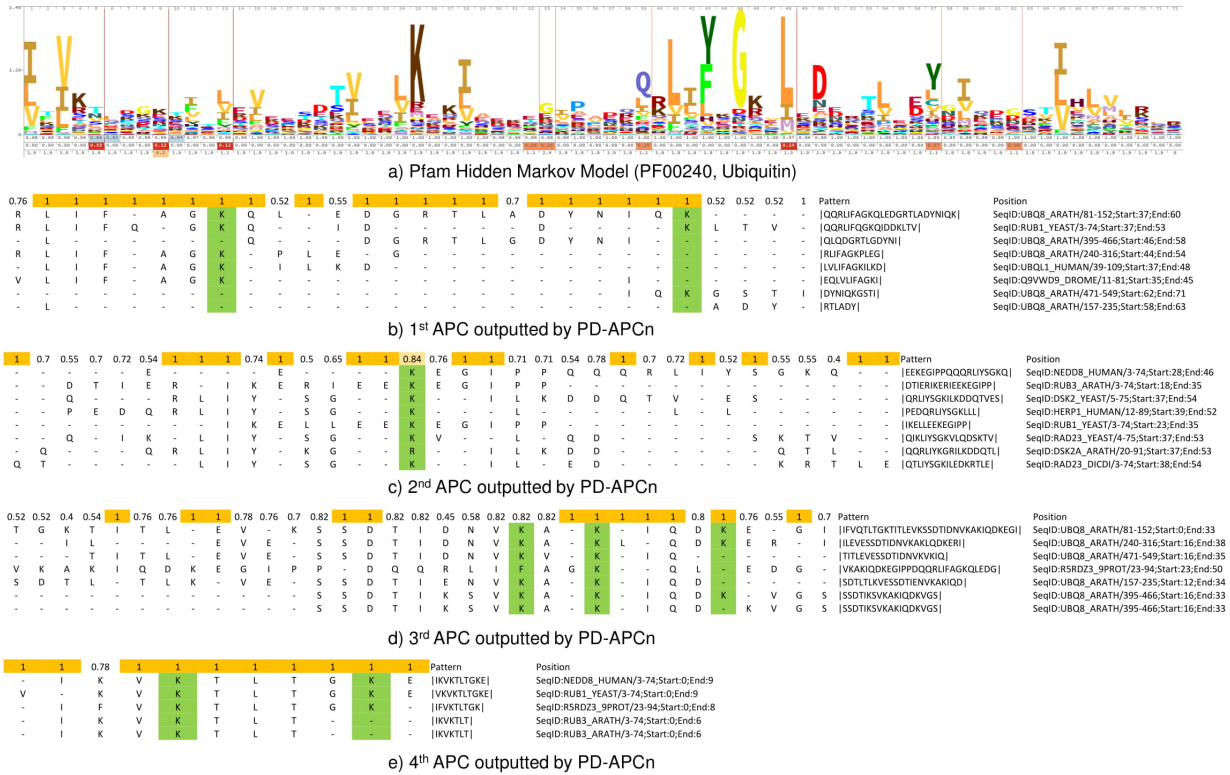


Figure 3.9: An illustration of the APCs outputted by PD-APCn on the Dataset Ubiquitin. (a) The Pfam Hidden Markov Model (HMM) logo of Ubiquitin. (b) The 1st APC outputted by PD-APCn on the Dataset Ubiquitin. It has covered Lys (K) 48 and Lys (K) 63. (c) The 2nd APC outputted by PD-APCn on the Dataset Ubiquitin. It has covered Lys (K) 33. (d) The 3rd APC outputted by PD-APCn on the Dataset Ubiquitin. It has covered Lys (K) 27, Lys (K) 29 and Lys (K) 33. (e) The 4th APC outputted by PD-APCn on the Dataset Ubiquitin. It has covered Lys (K) 6 and Lys (K) 11. All seven lysine (K) have been covered, where they are important for the formation of ubiquitin chains [137, 25, 130]. These results have further validated the capability of PD-APCn to discover functional regions in real protein sequences.

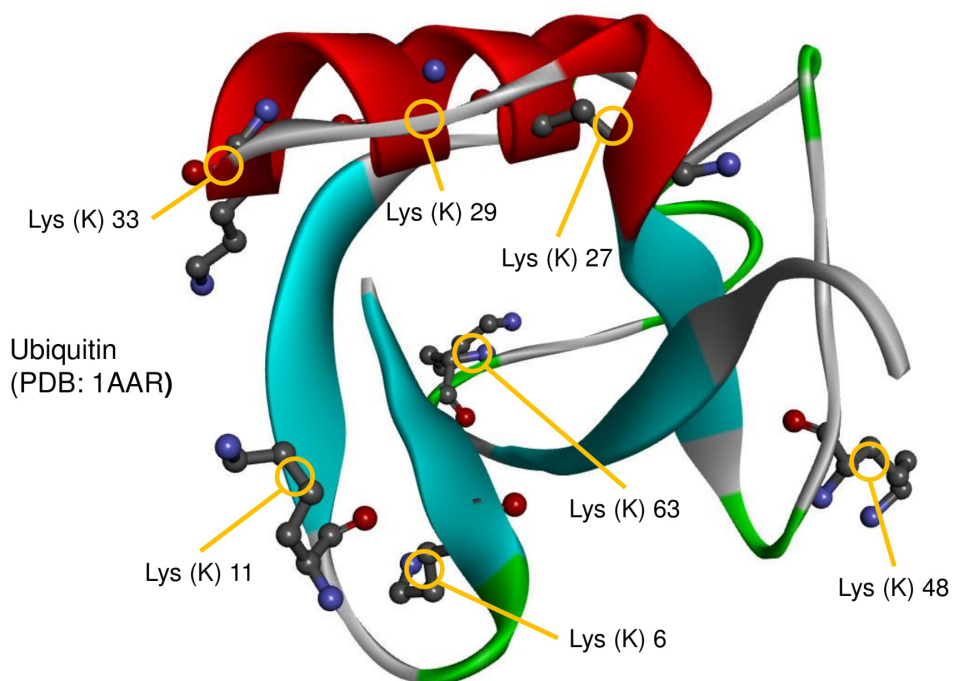


Figure 3.10: A three-dimensional structure of Ubiquitin obtained from Protein Data Bank (PDB) [12] (ID: 1AAR) is shown. The seven lysine (K), Lys (K) 6, Lys (K) 11, Lys (K) 27, Lys (K) 29, Lys (K) 33, Lys (K) 48 and Lys (K) 63, which are important for ubiquitination, are all covered by our discovered APCs, and highlighted, with $R1 > 0.8$ within our discovered APCs (Fig. 3.9(b)(c)(d)(e)).

indicating its robustness. We thus believe that PD-APCn [126, 127] would be important for the discovery of new functional regions from protein family sequences, as well as rare mutants that will be significant for personalized medicine in the future.

Chapter 4

Discovering Binding Cores from Protein-DNA interaction sequences using Protein-DNA Co-Occurrence APC

4.1 Introduction

As described in Chapter 2, the regions between a TF and a TFBS in close contact ($<3.5\text{\AA}$ [1, 99]) are referred to as Protein-DNA binding cores [18, 79] (see Figure 2.2). Understanding binding cores is of fundamental importance in deciphering Protein-DNA (TF-TFBS) binding and gene regulation. Limited by expensive experiments, it is promising to discover them with variations directly from sequence data. Although existing computational methods have produced satisfactory results, they are one-to-one mappings with no site-specific information on residue/nucleotide variations, where these variations in binding cores may impact binding specificity.

In this chapter, the study proposed a new representation known as Protein-DNA Co-occurrence APC for modeling binding cores by incorporating variations and an algorithm to discover them from only sequence data. A Protein-DNA Co-occurrence APC is a new representation model that is more compact than one-to-one pattern associations, as it packs many-to-many associations in one model, yet detailed enough to allow site-specific variants. In our experiment, the new algorithm took protein and DNA sequences from TRANSFAC (a Protein-DNA Interaction Sequence database in transaction format) as

input, and obtained binding cores with higher precision and much faster runtime ($\geq 1600x$) than that of its contemporaries. The new algorithm also discovered new protein-DNA binding cores that do not co-occur as one-to-one associated patterns in the raw data, via homology modeling.

4.2 Method

4.2.1 Problem Definition

Given a biological database DB with N transactions, the problem is to find a set of Protein-APCs \mathbb{C}_P and a set of DNA-APCs \mathbb{C}_D to form a set of Protein-DNA Co-Occurring APCs, i.e. $\mathbb{A} = \mathbb{C}_P \times \mathbb{C}_D$, such that the top K elements in \mathbb{A} maximize binding core verification measures. An overview is illustrated in Figure 4.1. A summary of the algorithm is provided in Algorithm 2.

4.2.2 Biological Database

To model the binding between Protein-DNA (TF-TFBS), we let $\Sigma_P = \{A, R, N, D, C, E, Q, G, H, I, L, K, M, F, P, S, T, W, Y, V\}$ and $\Sigma_D = \{A, C, G, T\}$ be alphabets of protein and DNA respectively, where $|\Sigma_P| = 20$ and $|\Sigma_D| = 4$. A protein sequence, s_P , is an element of Σ_P^* and a DNA sequence, s_D , is an element of Σ_D^* . A protein sequence is also denoted as a TF sequence and a DNA sequence is denoted as a TFBS sequence.

The input data is a biological database DB , which is a set of N transactions, i.e. $DB = \{T_1, T_2, \dots, T_N\}$. Each transaction describes the binding of one Protein (TF) sequence to many DNA (TFBS) sequences, i.e. $T_i = \{(s_P^i, \mathbb{S}_D^i) | \forall i = 1, 2, \dots, N\}$, where $s_P^i \in \Sigma_P^*$, $\mathbb{S}_D^i = \{s_D^1, s_D^2, \dots, s_D^{|T_i|}\}$ such that $s_D^i \in \Sigma_D^* \forall i = 1, 2, \dots, |T_i|$. For instance, a simplified biological database with only 4 transactions is shown in Table 4.1. A summary of notation is provided in Table 4.2.

4.2.3 APC Discovery

This section introduces an algorithm to mine APCs from Protein (TF) sequences and DNA (TFBS) sequences. The Protein sequences and DNA sequences are input to this algorithm independently to produce Protein-APCs and DNA-APCs.

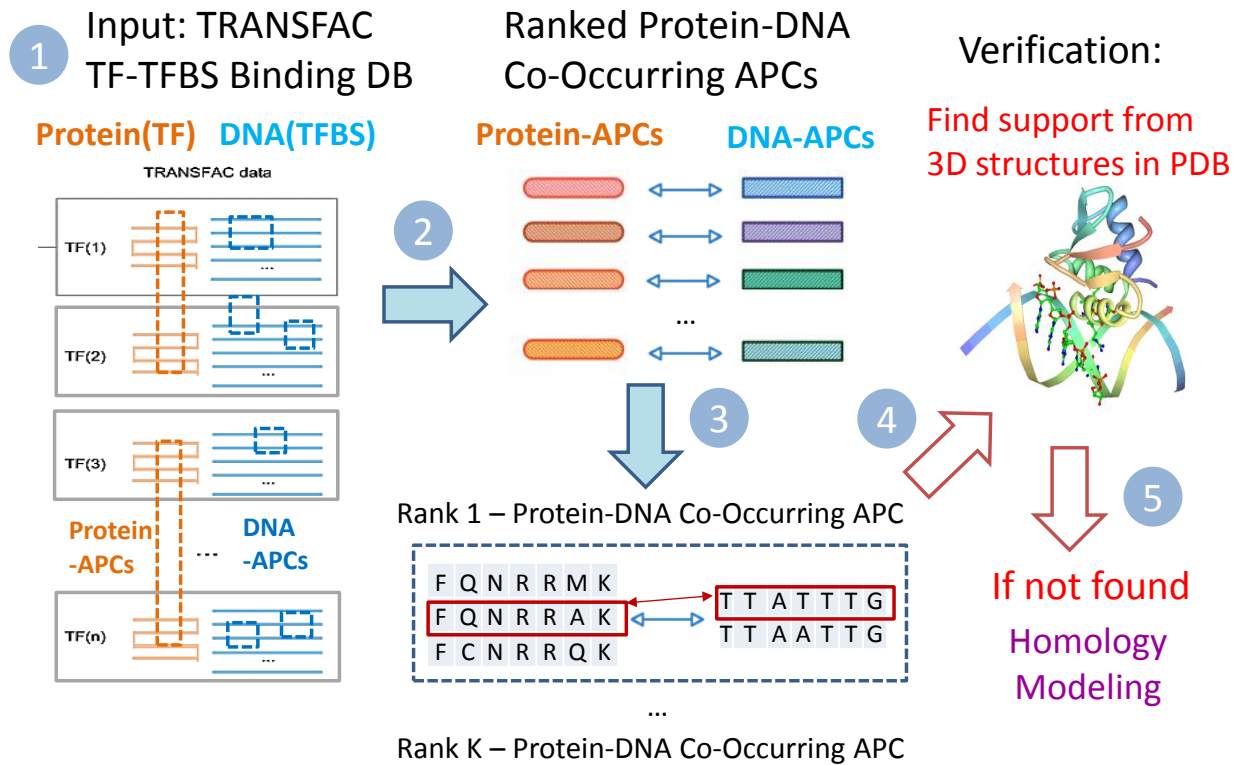


Figure 4.1: An overview of Protein-DNA (TF-TFBS) Binding Core discovery process. 1) The input is TRANSFAC [89], which is a database of Protein-DNA (TF-TFBS) binding sequences; 2) An Aligned Pattern Clustering algorithm [77] is applied to discover Protein-DNA Co-Occurring APCs and rank them according to their co-occurrence. 3) For the top-ranking Protein-DNA Co-Occurring APCs, binding core candidates are enumerated. 4) Each binding core candidate is checked if support can be found in the PDB. If found, the candidate is ascertained as a binding core. 5) If not found, homology modeling is conducted to an existing 3D structure closely matching to the candidate to check if the binding mechanism is chemically feasible. Replacing the combinatorial search of one-to-one co-occurrence in the entire transaction dataset to the many-to-many search of co-occurrences between patterns within each high-ranking Protein-DNA Co-Occurrence APC is key to the computational speed-up.

Table 4.1: A simplified example of TRANSFAC database on Protein-DNA interaction sequences

Transaction No. T_i	Protein (TF) Sequence s_P^i	DNA (TFBS) Sequence(s) S_D^i
1	FDERRMFR	{GACTTG}
2	MEDRKMFR	{ACTTCA}
3	MREFMVR	{GAGTTC}
4	VHMRHV	{GCACTT; AAGTAC}

Pattern discovery

First, we discover sequence patterns using a previously developed pattern discovery algorithm [144], referring to section 2.2.3. A pattern is defined as a sequence of ordered consecutive dependent symbols $p \in \Sigma^*$, where each symbol is either from Σ_P or Σ_D that is of the statistically significant [144]. A protein (TF) pattern is denoted as p_P and a DNA (TFBS) pattern is denoted as p_D . The resulting list of the discovered patterns are further pruned of redundant patterns and are represented by $\mathbb{P} = \{p^1, p^2, \dots, p^{|\mathbb{P}|-1}, p^{|\mathbb{P}|}\}$.

Aligned Pattern Clustering

Second, we group and align sequence patterns to maximize the vertical similarity of symbols between the patterns [77, 143], referring to section 2.2.3. Each cluster of aligned patterns is an APC. Let an APC be defined as

$$C = \text{ALIGN} \begin{pmatrix} p^1 \\ p^2 \\ \vdots \\ p^m \end{pmatrix}. \quad (4.1)$$

where each of the $|\mathbb{P}^l| = m$ patterns in the rows of C is of length $|C| = n$ by augmenting with $\Sigma \cup \{-\} \cup \{*\}$, where - and * denote ‘gap’ and ‘wildcard’ respectively. Hence, each aligned pattern is of the same length n .

A Protein-APC (TF-APC) is denoted as C_P and a DNA-APC (TFBS-APC) is denoted as C_D . Let a set of APCs be defined as $\mathbb{C} = \{C^1, C^2, \dots, C^{|\mathbb{C}|-1}, C^{|\mathbb{C}|}\}$.

Protein-APCs and DNA-APCs

Therefore we let a Protein-APC be a set of protein patterns $C_P^i = \mathbb{P}_P^i = \{p_P^{i,1}, p_P^{i,2}, \dots, p_P^{i,|\mathbb{P}_P^i|}\}$ where $p_P^{i,k} \in \Sigma_P^*, \forall k = 1, 2, \dots, |\mathbb{P}_P^i|$. Similarly, we let a DNA-APC be a set of DNA patterns $C_D^i = \mathbb{P}_D^i = \{p_D^{i,1}, p_D^{i,2}, \dots, p_D^{i,|\mathbb{P}_D^i|}\}$, where $p_D^{i,k} \in \Sigma_D^*, \forall k = 1, 2, \dots, |\mathbb{P}_D^i|$. We further let $\mathbb{C}_P = \{C_P^1, C_P^2, \dots, C_P^{|\mathbb{C}_P|}\}$ be a set of Protein-APCs, and $\mathbb{C}_D = \{C_D^1, C_D^2, \dots, C_D^{|\mathbb{C}_D|}\}$ be a set of DNA-APCs.

4.2.4 Protein-DNA Co-Occurring APCs

After discovering a set of Protein-APCs and a set of DNA-APCs, we then compose a set of Protein-DNA Co-Occurring APCs by associating each Protein-APC with each DNA-APC from TRANSFAC. We then rank each Protein-DNA Co-Occurring APC in descending order by a measure called the Co-Support, which takes a value between 0 and 1 inclusively.

Protein-APCs and DNA-APCs Associations

The relationship represented by a Protein-DNA Co-Occurring APC [74] $A_{i,j}$ is composed of a Protein-APC C_P^i and a DNA-APC C_D^j such that $A_{i,j} \in (C_P^i \times C_D^j)$ and

$$A_{i,j} = \{a_{i,j}^{m,n} | p_P^{i,m} \in C_P^i, p_D^{j,n} \in C_D^j\}, \quad (4.2)$$

in which $a_{i,j}^{m,n} = (p_P^{i,m}, p_D^{j,n})$ is a single one-to-one Protein-DNA pattern association. A set of Protein-DNA Co-Occurring APC $\mathbb{A} = \{A_{1,1}, A_{1,2}, \dots, A_{|\mathbb{C}_P|, |\mathbb{C}_D|}\}$.

Co-Support Measure

In a biological database DB with N transactions, we define the Co-Support of a Protein-DNA Co-Occurring APC $A_{i,j}$, i.e. $CS(A_{i,j})$, as the number of transactions that Protein-APC C_P^i and DNA-APC C_D^j co-occur over the total number of transactions. It measures how frequent the patterns in a Protein-APC and a DNA-APC co-occur in a biological database. The greater the value, the higher the co-occurrence of the patterns. We let

$$1. \text{trans}(C_P^i) = \bigcup_{k=1}^{|\mathbb{C}_P^i|} \text{trans}(p_P^{i,k});$$

Algorithm 2 APC Co-Occurrence Algorithm

Input: a Protein-DNA Binding Sequence DB with N transactions.

Output: a set of binding core candidates B

Step 1: Discover a set of Protein-APCs C_P from DB .

Step 2: Discover a set of DNA-APCs C_D from DB .

Step 3: Associate each Protein-APC with each DNA-APC

Step 4: Rank the APC pair by Co-Support

Step 5: Select the top K

return B

$$2. \text{trans}(C_D^j) = \bigcup_{k=1}^{|C_D^j|} \text{trans}(p_D^{j,k}),$$

in which $\text{trans}(p_P^{i,k})$ is the set of transactions where $p_P^{i,k}$ occurs; and $\text{trans}(p_D^{j,k})$ is the set of transactions where $p_D^{j,k}$ occurs. The Co-Support of a Protein-DNA Co-Occurring APC $A_{i,j}$ is hence defined as follows:

$$CS(A_{i,j}) = \frac{|\text{trans}(C_P^i) \cap \text{trans}(C_D^j)|}{N} \quad (4.3)$$

Forward-Confidence Measure

For an one-to-one Protein-DNA pattern association $a_{i,j}^{m,n} = (p_P^{i,m}, p_D^{j,n})$, we also compute its forward confidence [79, 148], i.e. $FConf$, to quantify the probability that, in the same transaction, the DNA-Pattern $p_D^{j,n}$ occurs given the Protein-Pattern $p_P^{i,m}$ occurs.

$$FConf(a_{i,j}^{m,n}) = \frac{|\text{trans}(p_P^{i,m}) \cap \text{trans}(p_D^{j,n})|}{|\text{trans}(p_P^{i,m})|} \quad (4.4)$$

4.2.5 Verification

To verify whether the Protein-DNA pattern associations in Protein-DNA Co-Occurring APCs are binding cores, we follow the procedure mentioned in [79, 148]. We first match them with the binding instance set called Extended PDB (See Section 4.3.1). We then apply homology modeling on the top-ranking cases not supported by the Extended PDB set to obtain viable binding cores.

Verification By PDB

Following [148], we define two levels of verification, i.e. ‘TF-verified’ & ‘Both-verified’ for a Protein-DNA pattern association $a_{i,j}^{m,n} = (p_P^{i,m}, p_D^{j,n})$. $a_{i,j}^{m,n}$ is said to be ‘TF-verified’, if at least one 5-mer obtained from $p_P^{i,m}$ is a substring of the TF-side of a binding instance. $a_{i,j}^{m,n}$ is said to be ‘Both-verified’, if any 5-mer obtained from $p_P^{i,m}$ and any 5-mer obtained from $p_D^{j,n}$ (with consideration on reverse complement) are substrings of the TF-side and the TFBS-side respectively of the same binding instance. All gaps and wildcards are removed from the patterns in the verification process.

Given a Protein-DNA Co-Occurring APC $A_{i,j}$, we define the set of all possible combination of Protein-DNA associated patterns for ‘TF-verified’ and ‘Both-verified’ as follows:

1. $V_{TF}(A_{i,j}) = \{a_{i,j}^{m,1} \in A_{i,j}\}$, where the 1 is a ‘dummy’ place holder of no significance;
2. $V_{Both}(A_{i,j}) = A_{i,j}$;
3. $\bar{V}_{TF}(A_{i,j}) = \{a \in V_{TF}(A_{i,j}) \mid a \text{ is TF-verified}\}$;
4. $\bar{V}_{Both}(A_{i,j}) = \{a \in V_{Both}(A_{i,j}) \mid a \text{ is Both-verified}\}$.

Given a set of Protein-DNA Co-Occurring APCs \mathbb{A} , we define two measures, i.e. R_{TF} and R_{Both} , for evaluating the verification performance.

$$R_{TF} = \frac{\sum_{i=1}^{|\mathbb{C}_P|} \sum_{j=1}^{|\mathbb{C}_D|} |\bar{V}_{TF}(A_{i,j})|}{\sum_{i=1}^{|\mathbb{C}_P|} \sum_{j=1}^{|\mathbb{C}_D|} |V_{TF}(A_{i,j})|} \quad (4.5)$$

$$R_{Both} = \frac{\sum_{i=1}^{|\mathbb{C}_P|} \sum_{j=1}^{|\mathbb{C}_D|} |\bar{V}_{Both}(A_{i,j})|}{\sum_{i=1}^{|\mathbb{C}_P|} \sum_{j=1}^{|\mathbb{C}_D|} |V_{Both}(A_{i,j})|} \quad (4.6)$$

The underlying meaning of Equations 4.5 and 4.6 of is accounting the sum of all verified cases in each Protein-DNA Co-Occurring APC over the sum of all possible variants in each Protein-DNA Co-Occurring APC.

Verification By Homology Modeling

Homology modeling constructs the three dimensional (3D) model of the ‘target’ protein at the atomic resolution level from its amino acid sequence and an experimentally constructed 3D structure. This approach is based on the concept that evolutionarily related proteins have similar sequences and naturally occurring homologous proteins have similar protein structure. It can be used to build the model of the protein-DNA or protein-RNA complexes. Following [79], we apply homology modeling on unverified Protein-DNA pattern associations in the Protein-DNA Co-Occurring APCs to show they are viable binding cores. The homologous 3D structures required by homology modeling can be quickly identified by referring to the verified Protein-DNA pattern associations within the same Protein-DNA Co-Occurring APCs.

4.3 Experiments and Results

4.3.1 Materials

Input Database: TRANSFAC

TRANSFAC [89] provides us with the sequences of TFs and their experimentally-proven binding DNA sequences for binding core discovery. As TFBSs embed in these DNA sequences, we denote them as TFBS sequences. Similar to the previous work [148], we employ TRANSFAC Professional version 2009.4 [89] in this study. Thus, entries not involving TF or TFBS sequences from it were discarded. To retain data quality, only TFBS sequences no shorter than 8 nucleotides were used. To reduce data redundancy, identical TFBS sequences corresponding to a TF were removed, in which only one of them was retained. After data pre-processing, we have one TF dataset with 706 full-length TF sequences (average length: 495), in which each TF on average binds 22 TFBS sequences (average length: 25).

Verification Data: Protein Data Bank

Protein Data Bank (PDB) [12] provides us with three-dimensional (3D) Protein-DNA complex structure data for verifying if the discovered patterns are indeed binding cores. We followed the approaches mentioned in [148] for pre-processing and collected from PDB 1226 distinct 3D Protein-DNA complex entries. For each PDB entry, a residue-nucleotide

pair is considered as binding if and only if its distance (between their closet atom-pairs) is less than or equal to 3.5\AA [1]. Starting with a residue-nucleotide binding pair, we extracted the adjacent residues and nucleotides to form a TF-TFBS binding sequence pair (which is also called a binding instance or a binding core). We set the length of both TF-side and TFBS-side sequences of a binding instance to be 9 [148]. Using this approach, we extracted 36679 binding instances. Together with 9 binding instances from annotations and literature obtained from [19], we have in total 36688 binding instances for verification. This set of binding instances is denoted as the Extended PDB.

4.3.2 Experimental Procedure

We first applied APC algorithm to mine a set of Protein-APCs and a set of DNA-APCs from TRANSFAC in two independent runs. We then formed a set of Protein-DNA Co-Occurring APCs by them and selected the top 100 Protein-DNA Co-Occurring APCs by their Co-Support Measure. We further computed R_{TF} and R_{Both} from the top 10 to top 100 by the Extended PDB for comparison. We finally applied homology modeling on two cases without support from Extended PDB to show that they are very likely to be binding cores. We summarize the setting of our approach as follows.

Procedure and Setting on Mining Protein-APCs

For TF sequences, pattern discovery was run with minimum occurrence of 10, length of 7 and default parameters of confidence interval of 3, and delta-closed of 0.9. A special condition of three consecutive amino acids is used to filter out patterns that are from acidic-rich, proline-rich, and glutamine-rich activation domains [115], which are unlikely binding cores; and, aligned pattern clustering was run based on global alignment, hamming distance considering gaps, and termination on 3 consecutive matches and 1 conserved column. The variation (dendrogram threshold) to be allowed is 4. No gaps and wildcards are allowed to be enclosed by amino acids.

Procedure and Setting on Mining DNA-APCs

For TFBS sequences, pattern discovery was run based on minimum occurrence of 100 and length of 7, as well as the default confidence interval of 3 and default delta-closed of 0.9. Aligned pattern clustering was then run based on global alignment, hamming distance (considering gaps), and termination on 3 consecutive matches and 1 conserved column. A

ID: 758		Co-Support: 0.0439													
Rank 1	VRVWFCN	VKIWFQN	IKIWFQN	WFCNRRQ	FCNRRQK	FQNRRMK	FQNRRAK	WFQNRRA	IWFQNRR	VWFQNRR					
TTATTG	0.0000	0.0000	0.0625	0.0417	0.0455	0.0385	0.0000	0.0000	0.0263	0.0000					
TTAATTG	0.1538	0.2400	0.6875	0.1667	0.1818	0.5769	0.3750	0.3684	0.4474	0.5294					
ID: 182		Co-Support: 0.0425													
Rank 2	EGCKGFF	CKGFFRR	CKGFFKR	KGFFRRS	GFFRRTI	KGFFRRT									
AGGTCAT	0.8333	0.6957	0.7273	0.5455	0.9000	0.8182									
AGGTCAG	0.7500	0.6522	0.6364	0.5455	0.8000	0.7273									
AGGTCAA	0.6667	0.6087	0.6364	0.6364	0.7000	0.6364									
AGGTCAC	0.7083	0.5217	0.6364	0.4545	0.7000	0.6364									
ID: 716		Co-Support: 0.0368													
Rank 4	VRVWFCN	VKIWFQN	IKIWFQN	WFCNRRQ	FCNRRQK	FQNRRMK	FQNRRAK	WFQNRRA	IWFQNRR	VWFQNRR					
CAATTAA	0.3077	0.3200	0.3750	0.2083	0.2273	0.4615	0.3125	0.3158	0.3421	0.4118					
ID: 721		Co-Support: 0.0368													
Rank 3	VRVWFCN	VKIWFQN	IKIWFQN	WFCNRRQ	FCNRRQK	FQNRRMK	FQNRRAK	WFQNRRA	IWFQNRR	VWFQNRR					
AATTTAA	0.3077	0.2400	0.4375	0.2500	0.2727	0.5000	0.3125	0.2632	0.3158	0.4118					
ID: 201		Co-Support: 0.0312													
Rank 5	EGCKGFF	CKGFFRR	CKGFFKR	KGFFRRS	GFFRRTI	KGFFRRT									
CAGGTCA	0.7917	0.6522	0.4545	0.3636	1.0000	0.9091									
TAGGTCA	0.5417	0.3913	0.3636	0.3636	0.5000	0.4545									
ID: 723		Co-Support: 0.0312													
Rank 6	VRVWFCN	VKIWFQN	IKIWFQN	WFCNRRQ	FCNRRQK	FQNRRMK	FQNRRAK	WFQNRRA	IWFQNRR	VWFQNRR					
TTTGCAT	0.4615	0.0000	0.1250	0.7917	0.8182	0.1154	0.0000	0.0000	0.0526	0.0000					
ID: 181		Co-Support: 0.0297													
Rank 7	EGCKGFF	CKGFFRR	CKGFFKR	KGFFRRS	GFFRRTI	KGFFRRT									
GACGTCA	0.2083	0.1304	0.1818	0.0909	0.2000	0.1818									
GAGGTCA	0.5833	0.5217	0.6364	0.4545	0.7000	0.6364									
ID: 586		Co-Support: 0.0283				ID: 592		Co-Support: 0.0283							
Rank 8	ESARRSR					Rank 9					ESARRSR				
ACGTGGC	0.9091					CACGTGG					0.8182				
						GACGTGG					0.5000				
ID: 720		Co-Support: 0.0283													
Rank 10	VRVWFCN	VKIWFQN	IKIWFQN	WFCNRRQ	FCNRRQK	FQNRRMK	FQNRRAK	WFQNRRA	IWFQNRR	VWFQNRR					
ATGCAAA	0.5385	0.0000	0.0000	0.8333	0.9091	0.0000	0.0000	0.0000	0.0000	0.0000					

Legend
Both verified
Not Both verified

Figure 4.2: The top 10 Protein-DNA Co-Occurring APCs (DNA variation = 1) where the ranking is based on Co-Support Measure and ID. For each table, the first row contains the patterns in the Protein-APC and the first column contains the patterns in the DNA-APC. A cell represents a Protein-DNA pattern association and the value within a cell is its forward confidence. The color scheme indicates if a Protein-DNA pattern association is Both-verified (Green) or not (Red). The number of green colored cells is 72 while the number of all cells is 111. The verification measure, R_{Both} , is thus 0.72.

minimum matching quality score of 0.85 is set to ensure 6 matched positions and pattern with length 7. The variation (dendrogram threshold) to be allowed is 1 or 2. No gaps and wildcards are allowed to be enclosed by nucleotides.

4.3.3 Comparative Schemes

To show that our results cannot be easily replicated, we developed two more schemes, i.e. ‘Unified Score Mode (USM)’ [19] and ‘Random’, to discover Protein-DNA Co-Occurring APCs from sequence data, since there are no existing algorithms.

In USM, we post-processed the output of the latest algorithm [19] developed for discovering TF-TFBS associated patterns (One-to-One Protein-DNA Pattern Associations) to produce representations similar to Protein-DNA Co-Occurring APCs. We first used complete-linkage hierarchical clustering (to ensure tightness) to cluster the top 1000 TF-TFBS associated patterns by the edit distance on the TF-side associated patterns and cut the dendrogram at 4. We then used complete-linkage hierarchical clustering again to cluster the TFBS-side associated patterns in each cluster formed by TF-sided associated pattern and cut the dendrogram at 1 or 2. Clusters of TF-TFBS associated patterns that are similar to the representation of Protein-DNA Co-Occurring APCs were produced. They are then re-ranked by Co-Support Measure in descending order and the top 100 of them were selected for comparison.

For USM, we used the parameters including Top=5, M=7, Mode=Sum/Normalized (Nor) and w=7. We refer ‘Mode’ to the type of the unified scores that rank the TF-TFBS associated patterns, where ‘Sum’ is the total score and ‘Normalized (Nor)’ is the total score normalized to the number of summed terms. We also refer ‘w’ to the width of both the TF-side and TFBS-side associated patterns. For the others, please refer to [19]. These parameter settings helped USM achieve its best verification performance [19].

In the scheme ‘Random’, on the top 100 Protein-DNA Co-Occurring APCs discovered by our algorithm, we randomly extracted a sequence segment from TRANSFAC to replace each pattern in each APC. We then re-computed the CoSupport, Measure re-ranked them, and re-computed R_{TF} and R_{Both} from top 10 to top 100. We repeated this process 100 times and reported the mean of R_{TF} and R_{Both} .

4.3.4 Top 10 Protein-DNA Co-Occurring APCs

In Figure 4.2 we show the verification performance of the Top 10 Protein-DNA Co-Occurring APCs. For DNA variation = 1, $R_{TF} = 1.00$, $R_{Both} = 0.72$; for DNA variation

$= 2$, $R_{TF} = 0.97$, $R_{Both} = 0.73$. For illustration, we showed the Top 10 Protein-DNA Co-Occurring APCs (DNA variation = 1) ranked by Co-Support Measure in a descending order. In tie cases, we ranked by ID in an ascending order. For each table, the first row contains the patterns in the Protein-APC and the first column contains the patterns in the DNA-APC. A cell represents a Protein-DNA pattern association and the value within a cell is its forward confidence. The color scheme refers to whether the Protein-DNA pattern association is Both-verified (Green) or not (Red). The number of green colored cells is 72 while the number of all cells is 111. Hence, $R_{Both} = 0.72$.

4.4 Discussion

4.4.1 Performance Comparison

The Protein-DNA Co-Occurring APCs or similar representations discovered by three different schemes: WeMine (our approach), USM and Random are compared on the verification performance in terms of R_{TF} and R_{Both} . Considering the DNA variation to be at most 1, Figures 4.3a and 4.3b show the Extended PDB verification in R_{TF} and R_{Both} respectively. WeMine is consistently better than other schemes from Top 10 to Top 100 in terms of R_{TF} , as shown in Figure 4.3a. The difference between WeMine and other algorithms is even larger in terms of R_{Both} , which is a stricter verification scheme, as shown in Figure 4.3b. Considering the DNA variation to be at most 2, Figures 4.3c and 4.3d show the Extended PDB verification in R_{TF} and R_{Both} respectively. WeMine is also observed to be consistently better than other schemes. These results show that the Protein-DNA Co-Occurring APCs discovered is neither random nor easily replicated by existing approaches.

Interestingly, referring to the rank 1 Protein-DNA Co-Occurring APC as shown in Figure 4.2, we observe that we have identified 3 Protein-DNA Pattern Associations, i.e. (FQNRRAK, TTATTTG), (WFQNRRA, TTATTTG), and (VWFQNR, TTATTTG) with zero forward confidence but both-verified. This demonstrates that the Protein-DNA Co-Occurring APC can model variants that do not exist in TRANSFAC. This implies that it has a stronger discovery power than previous algorithms [79, 148, 19] discovering one-to-one representations.

From the opposite perspective, we also notice some paired patterns that are not co-occurring are distant apart in three-dimensional structures.

A common problem in machine learning is the potential of over-fitting the model to the training data. Our method addresses overfitting in the following three manners. First, our

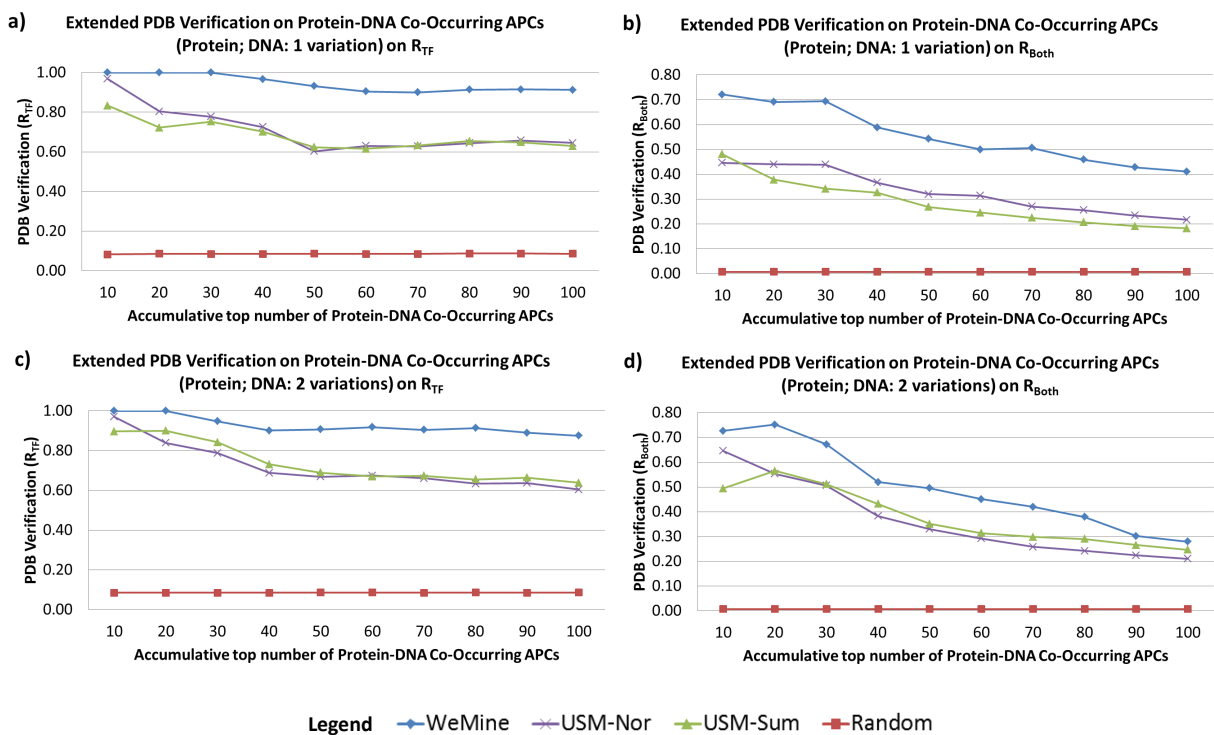


Figure 4.3: These figures illustrate the performance comparison between WeMine, USM and Random on Extended PDB verification. The x-axis is the accumulative top number of Protein-DNA Co-Occurring APCs, i.e. Top 1-10, Top 1-20,...,Top 1-100. The y-axis is the Extended PDB verification ratio corresponding to (a) R_{TF} considering DNA variation to be at most 1, (b) R_{Both} considering DNA variation to be at most 1, (c) R_{TF} considering DNA variation to be at most 2 and (d) R_{Both} considering DNA variation to be at most 2.

experimental framework is not supervised but unsupervised. In our experiments, we are not training predictive models and there are no class labels required in the model. Second, the new hypothetical complexes identified in this study are supported by high forward confidence. Third, the results remained stable over two different experimental conditions.

4.4.2 Run-time Comparison

We also compared the run-time used by the schemes WeMine, USM-Sum and USM-Nor in the experiment. The recorded run-time in seconds is summarized in Table 4.3. We observed

ID: 758		Co-Support: 0.0439								
Rank 1	VRVWFCN	VKIWFQN	IKIWFQN	WFCNRRQ	FCNRRQK	FQNRRMK	FQNRRAK	WFQNRRRA	IWFQNRR	VWFQNRR
TTATTG	0.0000	0.0000	0.0625	0.0417	0.0455	0.0385	0.0000	0.0000	0.0263	0.0000
TTAATTG	0.1538	0.2400	0.6875	0.1667	0.1818	0.5769	0.3750	0.3684	0.4474	0.5294

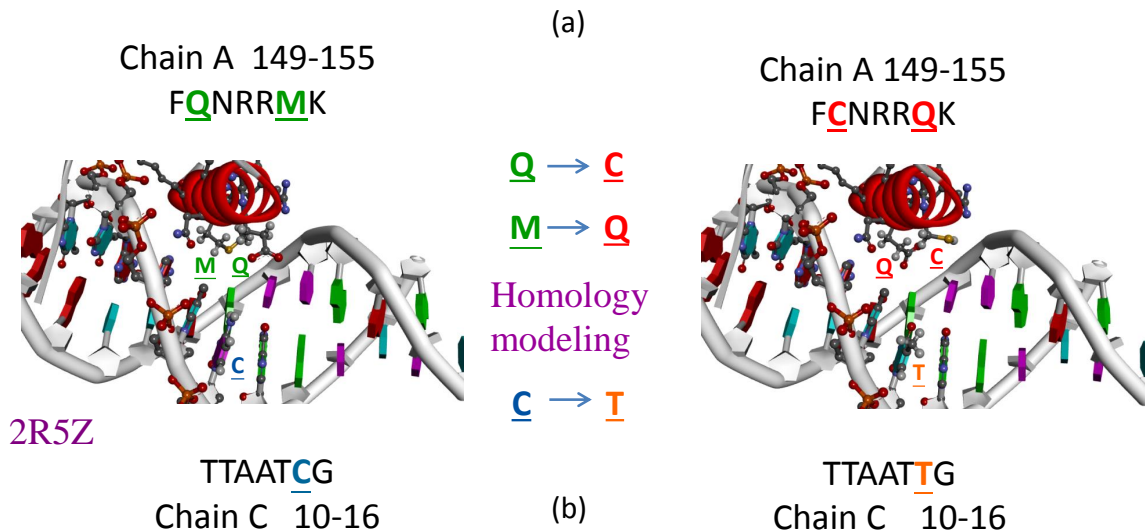


Figure 4.4: An illustration of how Protein-DNA Co-Occurrence APCs enabling Homology Modeling. (a) The top-ranking Protein-DNA Co-Occurring APCs (DNA variation = 1) is shown in the table. Binding core candidates denoted in green (red) have been verified (not verified) by PDB. Here, the target candidate (**FCNRRQK**, **TTAATTG**), shaded in red, is examined. Within the same Protein-DNA Co-Occurring APC, we search for cases in green closely matching to this target candidate to conduct homology modeling. (b) A 3D structure (2R5Z) with close matching in Protein-pattern (**FQNRRMK**) and DNA-pattern (**TTAATCG**) to the target case is found at the left of (b). In the middle of (b), mutations performed (Q150C and M154Q in Chain A; and C15 T in Chain C) are shown. At the right of (b), a homology model of the 3D structure of the target candidate is obtained.

that while both USM-Nor and USM-Sum used more than 20,000s, WeMine used only 12s. It should be noted that the post-processing on the outputs of USM used at most 3s and all experiments were run on the same computer (CPU:i5-2410M 2.3GHz, RAM:8.0GB). These results demonstrate a speed-up of $\geq 1600x$ on WeMine over USM [19]. The speed-up is brought by that USM needs to have $O(N)$ times of motif discovery (N refers to the number of transactions), while WeMine only needs two independent runs.

4.4.3 Homology Modeling

Case 1: Strong Protein-DNA Mutations

Consider the rank 1 Protein-DNA Co-Occurring APC in Figure 4.2, (FCNRRQK, TTAATTG), which is not supported by Extended PDB, has a forward confidence of 0.1818 that is higher than that of (FQNRRMK, TTAATTG), which is supported by PDB record 2RMZ. Due to the high similarity between these two Protein-DNA pattern associations, a few mutations can be introduced to 2R5Z to model the interested Protein-DNA binding. We performed mutations of Q150C, M154Q on the Chain A and C15T on the chain C, of the PDB record 2R5Z, as shown in Figure 4.4. Most of the hydrogen bonds are maintained at the Protein-DNA binding site. This indicates that the interaction between the DNA and protein is as strong as the original case. Intriguingly, we observed that the removal of sulfur which interacts with several nucleotides of DNA from M154Q mutation is replenished by sulfur from Q150C mutation, suggesting the flexibility of the variation of Protein-DNA binding in nature. Hence, (FCNRRQK, TTAATTG) is supported by homology modeling to be a very likely binding core.

Case 2: DNA Mutation

Consider the rank 1 Protein-DNA Co-Occurring APC in Figure 4.2, (WFCNRRQ, TTAATTG), which is not supported by Extended PDB, has a forward confidence of 0.1667 that is higher than that of (WFCNRRQ, TTATTTG), which is supported by PDB record 1CQT. Due to the high similarity between these two Protein-DNA pattern associations, a few mutations can be introduced to 1CQT to model the interested Protein-DNA binding. We performed a mutation of T222A on the Chain N of the PDB record 1CQT, as shown in Figure 4.5. The T222A mutation on the DNA still maintains the hydrogen bonds to residues in close proximity such as glutamine and asparagine. Hence, (WFCNRRQ, TTAATTG) is supported by homology modeling to be a very likely binding core.

4.5 Summary

In this chapter, we proposed a new representation denoted as Protein-DNA Co-Occurring Aligned Pattern Cluster (APC) for modeling Protein-DNA binding with variations. It is more compact than one-to-one pattern associations, as it packs many-to-many associations in one model, yet detailed enough to allow site-specific variants. We also developed a novel

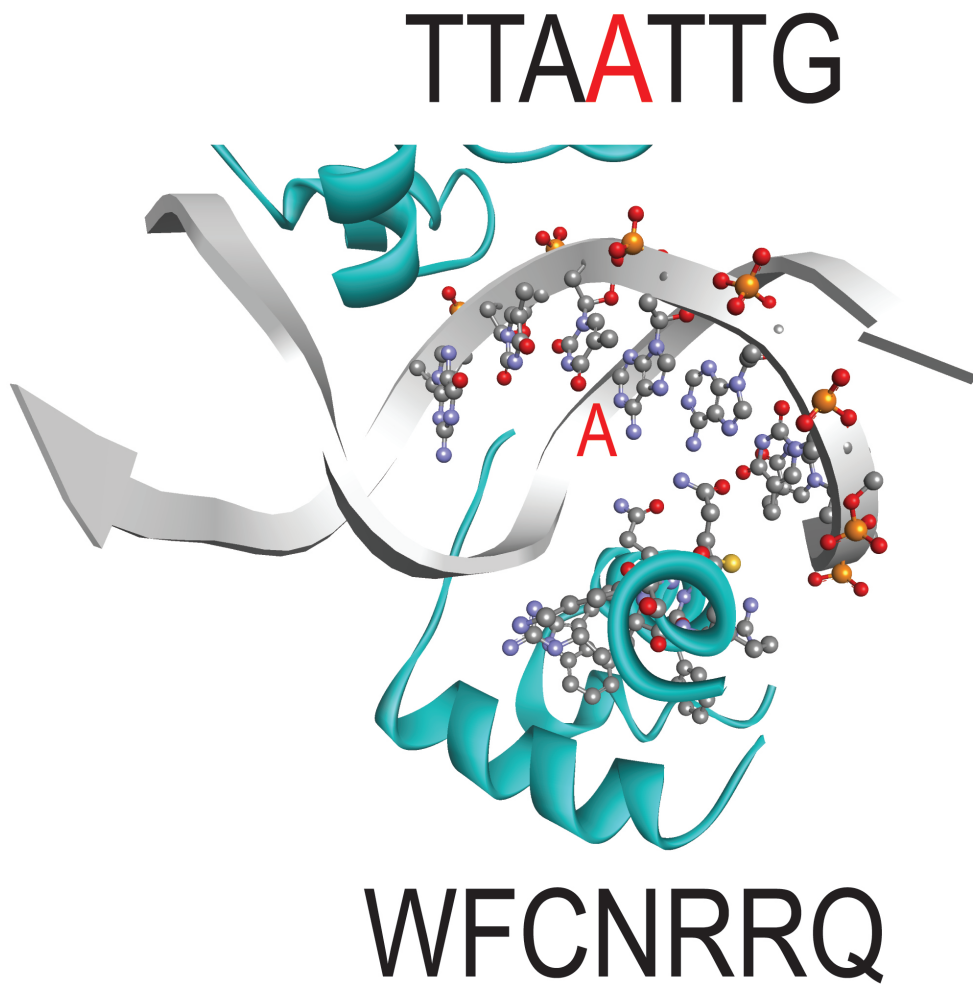


Figure 4.5: A homology modeling of 1CQT with a mutation of T222A on Chain N to model the Protein-DNA Binding (WFCNRRQ, TTAATTG).

algorithm to mine Protein-DNA Co-Occurring APCs, discovering binding cores at a higher precision that is faster ($\geq 1600x$) than other methods. We further demonstrated the use of our work for intuitive analysis to synthesize new knowledge in Protein-DNA binding. Two new binding cores, i.e. (FCNRRQK, TTAATTG) and (WFCNRRQ, TTAATTG) have been discovered by homology modeling assisted by Protein-DNA Co-Occurring APCs to locate close matching 3D structures. There are three benefits of Protein-DNA Co-Occurring APC to researchers: (1) site-specific information on variants, (2) significant speed-up and (3) discovery of binding cores that do not exist as one-to-one associations. We believe that this novel presentation will be useful in future applications involving Protein-DNA binding, in particular for assisting sequence-based Protein-DNA binding prediction, and enable the novel discovery of potential Protein-DNA co-evolution.

Table 4.2: A summary of notations with examples on Protein-DNA interaction sequences

Description	Notation	Example(s)
A Protein-APC	C_P^i	$C_P^1 = \{p_P^{1,1}=\text{MFR}, p_P^{1,2}=\text{MVR}\},$ $C_P^2 = \{p_P^{2,1}=\text{MRE}, p_P^{2,2}=\text{MRH}\}$
A DNA-APC	C_D^j	$C_D^1 = \{p_D^{1,1}=\text{ACTT}, p_D^{1,2}=\text{AGTT}\}$
Protein-APCs	\mathbb{C}_P	$\mathbb{C}_P = \{C_P^1, C_P^2\}$
DNA-APCs	\mathbb{C}_D	$\mathbb{C}_D = \{C_D^1\}$
A One-to-One Protein-DNA Pattern Association	$a_{i,j}^{m,n}$	$a_{2,1}^{1,2} = (\text{MRE}, \text{AGTT})$
A Protein-DNA Co-Occurring APC	$A_{i,j}$	$A_{1,1} = \{a_{1,1}^{1,1} = (\text{MFR}, \text{ACTT}),$ $a_{1,1}^{1,2} = (\text{MFR}, \text{AGTT}),$ $a_{1,1}^{2,1} = (\text{MVR}, \text{ACTT}),$ $a_{1,1}^{2,2} = (\text{MVR}, \text{AGTT})\}$ $A_{2,1} = \{a_{2,1}^{1,1} = (\text{MRE}, \text{ACTT}),$ $a_{2,1}^{1,2} = (\text{MRE}, \text{AGTT}),$ $a_{2,1}^{2,1} = (\text{MRH}, \text{ACTT}),$ $a_{2,1}^{2,2} = (\text{MRH}, \text{AGTT})\}$
Protein-DNA Co-Occurring APCs	\mathbb{A}	$\mathbb{A} = \{A_{1,1}, A_{2,1}\}$
Protein Pattern to Transactions	$\text{trans}(p_P)$	$\text{trans}(\text{MRE}) = \{3\}$ $\text{trans}(\text{MRH}) = \{4\}$
Protein APC to Transactions	$\text{trans}(C_P^i)$	$\text{trans}(C_P^1)$ $= \text{trans}(\text{MRE}) \cup \text{trans}(\text{MRH})$ $= \{3,4\}$
DNA Pattern to Transactions	$\text{trans}(p_D)$	$\text{trans}(\text{ACTT}) = \{1,2,4\}$ $\text{trans}(\text{AGTT}) = \{3\}$
DNA APC to Transactions	$\text{trans}(C_D^j)$	$\text{trans}(C_D^1)$ $= \text{trans}(\text{ACTT}) \cup \text{trans}(\text{AGTT})$ $= \{1,2,3,4\}$
Co-Support of a Protein-DNA Co-Occurring APC	$CS(A_{i,j})$	$CS(A_{2,1})$ $= \frac{ \text{trans}(C_P^2) \cap \text{trans}(C_D^1) }{4}$ $= \frac{ \{3,4\} }{4} = \frac{2}{4} = 0.5$
Forward Confidence of a Protein-DNA Co-Occurring APC	$FConf(a_{i,j}^{m,n})$	$FConf(a_{2,1}^{1,2})$ $= \frac{ \text{trans}(\text{MRE}) \cap \text{trans}(\text{AGTT}) }{ \text{trans}(\text{MRE}) }$ $= \frac{ \{3\} }{ \{3\} } = \frac{1}{1} = 1.0$
All Possible Variants (TF) of a Protein-DNA Co-Occurring APC	$V_{TF}(A_{i,j})$	$V_{TF}(A_{2,1})$ $= \{a_{2,1}^{1,1} = (\text{MRE}, \text{ACTT}),$ $a_{2,1}^{2,1} = (\text{MRH}, \text{ACTT})\}$
All Possible Variants (Both) of a Protein-DNA Co-Occurring APC	$V_{Both}(A_{i,j})$	$V_{Both}(A_{2,1}) = A_{2,1}$

Table 4.3: Runtime comparison between WeMine, USM-Nor and USM-Sum

Scheme	DNA Variation 1	DNA Variation 2
WeMine	12.42s	14.78s
USM-Nor	20152.30s	20151.18s
USM-Sum	20193.83s	20193.21s

Chapter 5

Predicting Protein-Protein Interaction Using Protein-Protein Co-occurrence APC

5.1 Introduction

Protein-Protein interaction (PPI) is important for various biological processes and functions in living cells such as metabolic cycles, DNA transcription and replication, and signaling cascades [40]. Following [104, 47], we refer to a PPI as an interaction that brings two different proteins A and B into direct physical contact, i.e. heterodimeric interactions. Protein-Protein interaction prediction refers to a process to predict if one protein will interact with another protein. It is critical for better understanding the molecular mechanisms inside the cell [40], and is particularly useful for discovering unknown functions of a protein [56].

Sequence-based Protein-Protein interaction prediction is a process to predict if one protein will interact with another protein using only their sequences as input to a computer program. As described in section 2.4.3, motivated by the general applicability of sequence-based methods and realization of the drawbacks of the existing algorithm, the objective of this chapter is to develop a new sequence-based Protein-Protein interaction prediction method which is (1) based on biologically interpretable features, (2) based on features that are more biologically realistic such as allowing variable widths and mutations, and (3) achieving satisfactory prediction performance. In this chapter, to accomplish these objectives, we propose a new algorithm WeMine-P2P, leveraging a new representation

model known as the Protein-Protein Co-Occurrence APC which captures co-occurrence functional regions allowing variable widths and mutations. Comprehensive experiments were also conducted to validate the effectiveness of WeMine-P2P with comparison with existing software.

5.2 Methods

Overview. An overview of our method in steps 1 to 6 is illustrated in Fig 5.1.

Problem definition. A protein pair, or a PPI pair is defined as a pair of protein sequences that can either be interacting or not interacting with one another. A protein-protein interaction pair, referred to as a positive PPI pair, is defined as a pair of protein sequences that can interact with each other. A protein-protein non-interaction pair, or a negative PPI pair, is defined as a pair of protein sequences that cannot (or are not yet known to) interact with each other. A PPI database includes protein sequences, as well as both positive and negative PPI pairs. We use it to train a model for predicting whether a new protein pair would interact or not. The PPI prediction output score would be within the range of 0 and 1 inclusively: the higher the score the more likely that the two protein sequences are predicted to be interacting.

Input PPI Database. The input dataset, denoted PPI Database (PPI-DB), consists of a set of protein sequences, as well as positive and negative PPI pairs. To model the protein sequence patterns, we let $\Sigma = \{A, R, N, D, C, E, Q, G, H, I, L, K, M, F, P, S, T, W, Y, V\}$ be the protein alphabet containing 20 amino acids, where $|\Sigma| = 20$. A protein sequence from the PPI database $S = \sigma_1\sigma_2 \dots \sigma_{|S|-1}\sigma_{|S|}$ is an element of Σ^* , where each $\sigma_i \in \Sigma$ and S is of length $|S|$. Let the set of input protein sequences be defined as $\mathbb{S} = \{S_x | x = 1, \dots, |\mathbb{S}|\} = \{S_1, S_2, \dots, S_{|\mathbb{S}|-1}, S_{|\mathbb{S}|}\}$.

Step 1: Label positive and negative PPI pairs. We label the positive and negative PPI pairs provided by PPI-DB as “+” class and “-” class respectively (Fig 5.1). This helps to form the training set for binary classification, in which a training sample is a protein pair pertaining either to a “+” or “-” class. Formally, we let $\mathbb{B} = \mathbb{S} \times \mathbb{S} = \{B_{1,1}, B_{1,2}, \dots, B_{|\mathbb{S}|,|\mathbb{S}|}\}$, where each protein pair $B_{x,y}$ is composed of two protein sequences S_x and S_y such that $B_{x,y} = (S_x, S_y)$.

Step 2: Obtain Aligned Pattern Clusters from PPI-DB. We obtain conserved regions from PPI-DB that maintain variable mutations and flexible length (Fig 5.1). It should be noted that the definitions here refer to section 2.2.3. To achieve this, first from the input protein sequences we use a pattern discovery algorithm [144] to discover

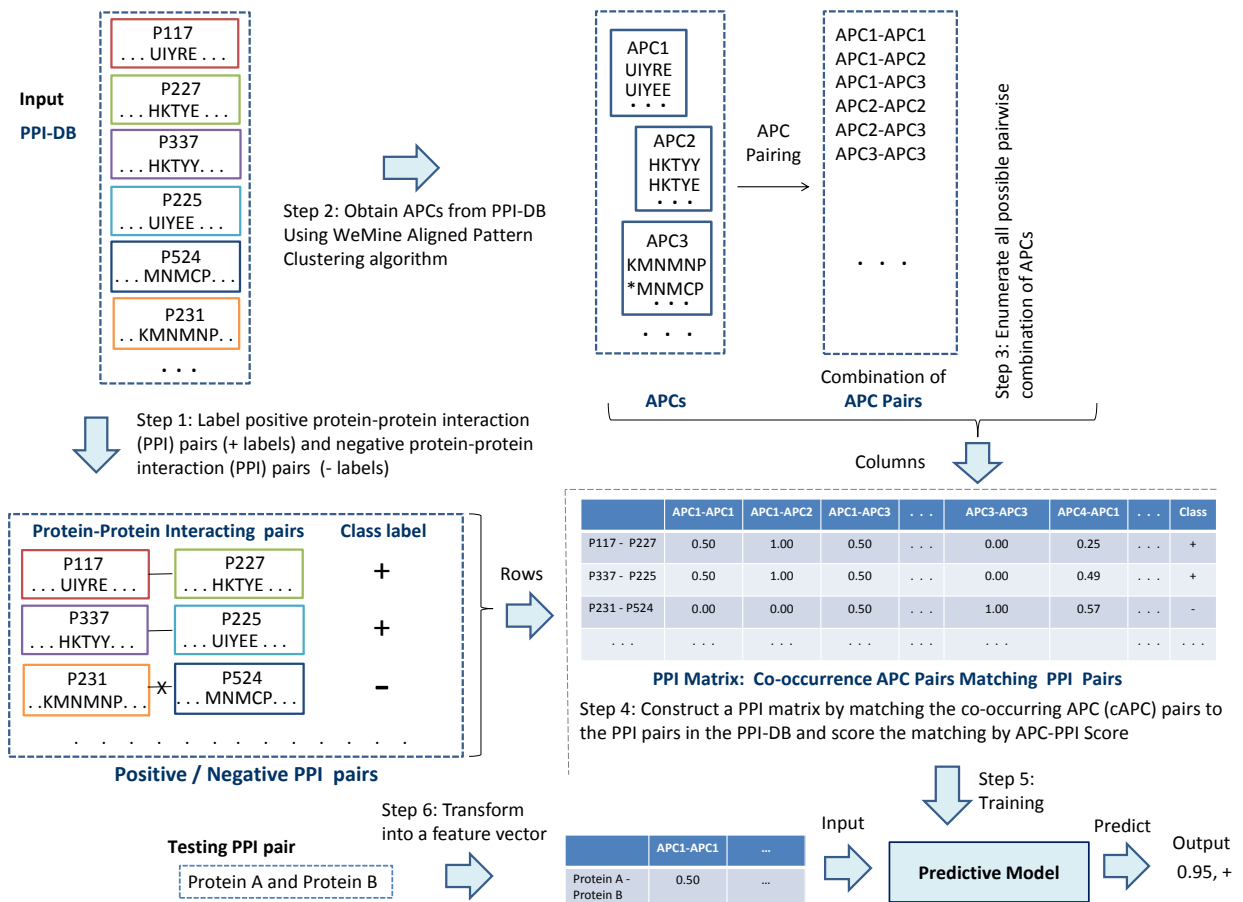


Figure 5.1: WeMine-P2P: a PPI Predictor. The input dataset, denoted as PPI Database (PPI-DB), consists of a set of protein sequences, as well as positive and negative PPI pairs. Each protein sequence has a unique ID, e.g. P117, P227...etc. For illustration, only some segments on a protein sequence are shown. To train a predictive model, positive and negative protein-protein interaction pairs are labeled by “+” and “-” labels respectively (Step 1). For extracting features, APCs are obtained from PPI-DB using WeMine Aligned Pattern Clustering algorithm (Step 2). All possible pairwise combination of APCs are then enumerated as co-occurring APC pairs (cAPC pairs) (Step 3). To construct a PPI matrix, cAPC pairs are then matched to the PPI pairs in the PPI-DB and the matchings are scored by APC-PPI (Step 4). A predictive model is trained on the PPI matrix, where each of its rows is a feature vector with a class label (+) or (-) as its last element (Step 5). Any protein pair can be turned into a feature vector by computing the APC-PPI of all cAPC pairs and concatenating the APC-PPIs. The feature vector can then be inputted to the trained model to output the classification results (Step 6).

sequence patterns. Here, we define an unaligned sequence pattern \bar{p} as an ordered sequence of interdependent symbols from Σ , i.e. $\bar{p} = \bar{\sigma}^1 \bar{\sigma}^2 \dots \bar{\sigma}^{|\bar{p}|}$, where $\bar{\sigma}^j \in \Sigma, \forall j = 1, 2, \dots, |\bar{p}|$, that pass the four statistical conditions defined in [144]. The list of unaligned patterns discovered is $\mathbb{P} = \{\bar{p}^i | i = 1, \dots, |\mathbb{P}|\} = \{\bar{p}^1, \bar{p}^2, \dots, \bar{p}^{|\mathbb{P}|-1}, \bar{p}^{|\mathbb{P}|}\}$. Next, we cluster and align these unaligned sequence patterns using the Aligned Pattern Clustering algorithm [77, 143]. Each cluster is an Aligned Pattern Cluster (APC) [77, 143], i.e.

$$C^l = \text{ALIGN} \begin{pmatrix} \bar{p}^1 \\ \bar{p}^2 \\ \vdots \\ \bar{p}^m \end{pmatrix} = \begin{pmatrix} p^1 \\ p^2 \\ \vdots \\ p^m \end{pmatrix} \quad (5.1)$$

$$= \begin{pmatrix} \sigma_1^1 & \sigma_2^1 & \dots & \sigma_n^1 \\ \sigma_1^2 & \sigma_2^2 & \dots & \sigma_n^2 \\ \vdots & \vdots & \vdots & \vdots \\ \sigma_1^m & \sigma_2^m & \dots & \sigma_n^m \end{pmatrix}_{m \times n}, \quad (5.2)$$

where $\sigma_j^i \in \Sigma \cup \{-\} \cup \{*\}$. Note that $-$ denotes a gap character and $*$ denotes a wildcard character. Each APC C^l contains m aligned patterns, where each of them is of length n , i.e. $p^i = \sigma_1^i \sigma_2^i \dots \sigma_n^i, \forall i = 1, 2, \dots, m$. Let a set of APCs be defined as $\mathbb{C} = \{C^l | l = 1, \dots, |\mathbb{C}|\} = \{C^1, C^2, \dots, C^{|\mathbb{C}|-1}, C^{|\mathbb{C}|}\}$.

Step 3: Enumerate all possible pairs of APC that co-occur. We enumerate all possible pairs of APCs and call a pair of APC as a co-occurring Aligned Pattern Cluster pair (cAPC pair) (Fig 5.1). We obtain a set of cAPC pairs as $\mathbb{A} = \mathbb{C} \times \mathbb{C} = \{A_{1,1}, A_{1,2}, \dots, A_{|\mathbb{C}|,|\mathbb{C}|}\}$, where there are in total $|\mathbb{C}| \times |\mathbb{C}| = N$ number of cAPC pairs. Each cAPC pair $A_{i,j}$ is composed of two APCs C^i and C^j such that $A_{i,j} = (C^i, C^j)$. These cAPC pairs are features extracted from PPI-DB instantly in order to predict PPI between PPI pairs.

Step 4: Construct a Protein-Protein Interaction Matrix. We use the PPI matrix M to register the match between cAPC pairs and PPI pairs. Each row in the PPI matrix M is associated with a PPI pair say $B_{x,y}$. Each column in the PPI matrix M is associated with a cAPC pair $A_{i,j}$, with the last column being the class label. Each cell in the matrix M registers the score of a cAPC pair $A_{i,j}$ co-occurring in the protein pair $B_{x,y}$ (Fig. 5.1). Hence, stating in a more specific way, each cell of the PPI matrix M is the $score(A_{i,j}, B_{x,y})$. This score indicates the strength of occurrence of patterns in the cAPC pair $A_{i,j}$ on the protein pair $B_{x,y}$. APC-PPI is devised to determine the value of $score(A_{i,j}, B_{x,y})$. APC-PPI is computed based on the best match between cAPC pair

and PPI pair, which is picking the best segment match between all permutations between the APC $A_{i,j} = (C_i, C_j)$ and protein sequences $B_{x,y} = S_x, S_y$. Here, a PPI pair $B_{x,y}$ is represented by a row of APC-PPIs, where each of them is considered as a PPI feature.

Match Score Given an APC C with m patterns with length n and a sequence segment $s = \sigma'_1 \sigma'_2 \dots \sigma'_n$, we define the Match Score, i.e. $match(C, s)$, as:

$$match(C, s) = \frac{1}{n} \sum_{i=1}^n x_i, \quad (5.3)$$

where

$$x_i = \begin{cases} 1, & \text{if } \sigma'_i = \sigma_i^j \forall j \in \{1, 2, \dots, m\} \\ 0, & \text{otherwise} \end{cases} \quad (5.4)$$

This score reflects the ratio of the characters in a sequence segment s matching the characters in an APC C of the same length n , as exemplified in Figure 5.2, achieved by Algorithm 3, which is designed to match approximately the patterns in an APC C to a sequence segment s of the same length. Algorithm 3 first checks if each character in the segment occurs in the APC column of the same index. It then sums up the number of matches, and normalizes the sum by the length of the segment.

Segment: **HAPPI**

APC: **NAPPA**
 HOPPY

Match: 4 Length of segment: 5

APCmatchingSegment Score = $4/5 = 0.8$

Figure 5.2: An example on how Match Score is calculated for a sequence segment with 5 characters and an APC with 2 rows.

Algorithm 3 Match Score

Input: An APC C of size $m \times n$, a sequence segment $s = \sigma'_1 \sigma'_2 \dots \sigma'_n$

Output: a value in range $[0 \ 1]$

for character σ'_i in s **do**

 Add match count if σ'_i is found in $\sigma_i^1 \sigma_i^2 \dots \sigma_i^m$ of C { i : column index; m : number of rows in APC C }

end for

return match count / $|s|$

MaxMatch Score Given an APC C with m patterns and length n , a protein sequence S , we define the MaxMatch Score, i.e. $MaxMatch(C, S)$, as:

$$MaxMatch(C, S) = \max_{i=1,2,\dots,|S|-n+1} Match(C, S[i, i+n]), \quad (5.5)$$

where $S[i, i+n] = \sigma_i \sigma_{i+1} \dots \sigma_{i+n-1}$ which is a substring of S . This score represents the strength of occurrence of an APC C on a protein sequence S , achieved by Algorithm 4. It uses a sliding window of the APC length over the sequence and computes a Match Score for all segments (at an amount of $|S| - n + 1$) of segments. The maximum Match Score is chosen as the output.

Algorithm 4 MaxMatch Score

Input: An APC C with m patterns and length n , a Protein sequence S

Output: A value in range $[0 \ 1]$

for $i = 1$ to $|S| - n + 1$ **do**

 Find Match Score of $S[i, i+n]$ and C

end for

return Maximum Match Score

APC-PPI Given a cAPC pair $A_{i,j}$ composed of APCs C^i and C^j , and a PPI pair $B_{x,y}$, with Protein sequences S_x and S_y , we define APC-PPI as:

$$APC - PPI(B_{x,y}, A_{i,j}) = \max\{p_1 \times p_2, p_3 \times p_4\}, \quad (5.6)$$

where $p_1 = MaxMatch(C_i, S_x)$, $p_2 = MaxMatch(C_j, S_y)$, $p_3 = MaxMatch(C_i, S_y)$ and $p_4 = MaxMatch(C_j, S_x)$. The APC-PPI measures the strength of occurrence of a cAPC pair on a PPI pair, obtained from Algorithm 5, which first calculates two MaxMatch

Scores for each of the two possible APC-Protein combinations. Then the product of the MaxMatch Scores in each APC-Protein combination is calculated. Then, each of the two APC-Protein combinations is associated with a score. The APC-PPI is the maximum one among those two scores.

Algorithm 5 APC-PPI

Input: A cAPC pair $A_{i,j}$ composed of APCs C^i and C^j , and a PPI pair $B_{x,y}$, with Protein sequences S_x and S_y .

Output: A value in range [0 1]

Let $Score1$ = Product of MaxMatch Score between (C^i, S_x) and (C^j, S_y)

Let $Score2$ = Product of MaxMatch Score between (C^i, S_y) and (C^j, S_x)

return Max of $Score1$ and $Score2$

Step 5: Train a predictive model based on PPI Matrix. We train a predictive model, specifically a Random Forest [15], based on the constructed PPI matrix. A random forest is an ensemble learning method. In this study, we use it mainly for binary classification, i.e. to predict if a protein pair is a positive or negative PPI pair. It operates by constructing a number of decision trees in the training process, then outputting the class label by voting, i.e. the mode of individual trees. We choose Random Forest as our predictive model because 1) it runs efficiently on large training sets and is easily parallelized [15]; 2) it can handle a large number of input variables without variable deletion [15]; 3) it seldom overfits the training set [15]. We adopt the machine learning package WEKA 3.7 [46] in training the Random Forest predictive model, using 3000 trees. It supports outputting the prediction probability in addition to the class label.

Step 6: Transform the testing protein pairs. Given a testing protein pair, we first transform it into a feature vector by computing the APC-PPI of all extracted cAPC pairs to itself. When the feature vector is constructed, we then input it to the predictive model to obtain a class label, and also the probability of the prediction (supported by WEKA [46]).

Feature analysis: cAPC pair selection. To analyze the features, we have developed a score to measure how distinct a cAPC pairs column, $A_{i,j}$, in the PPI matrix is. For example, the higher the score the cAPC pair can obtain, the more likely that it will co-occur in a positive PPI pair but less in a negative PPI pair. This score is built upon the APC-PPI but needs to be normalized to the number of PPI pairs (positive or negative) in the PPI matrix. We first define

$$tscore(A_{i,j}, B_{x,y}) = \begin{cases} \frac{score(A_{i,j}, B_{x,y})}{posPPI}, & \text{if +ve PPI pair,} \\ -\frac{score(A_{i,j}, B_{x,y})}{negPPI}, & \text{if -ve PPI pair,} \end{cases} \quad (5.7)$$

where $score(A_{i,j}, B_{x,y})$ is the APC-PPI, $posPPI$ is the total number of positive PPI pairs, and $negPPI$ is the total number of negative PPI pairs. The $tscore(A_{i,j}, B_{x,y})$ that relates to a cAPC pair $A_{i,j}$ is summed over all PPI pairs in \mathbb{B} . We define

$$hscore(A_{i,j}) = \sum_{\forall B_{x,y} \in \mathbb{B}} tscore(A_{i,j}, B_{x,y}) \quad (5.8)$$

We can then use $hscore$ to rank the cAPC pairs $A_{i,j}$.

5.3 Experiments and Results

5.3.1 Materials

In our experiments, 40 independent Yeast_Random datasets were downloaded from [104] at <http://www.marcottelab.org/differentialGeneralization>. The procedure to obtain these 40 datasets is described below. Yeast Protein-Protein Interaction (PPI) data (Saccharomyces_cerevisiae-20100304.txt) containing the protein sequences and the positive PPI pairs was acquired from the protein interaction network analysis platform [150]. Further pre-processing was applied to the proteins therein. First, the proteins were clustered using CD-HIT2 [80] with the requirement that they shared sequence identity less than 40%. Second, the proteins with less than 50 amino acids as well as homo-dimeric interactions were also removed. In total, 6806 Yeast protein sequences remained after the pre-processing.

It is shown by [104] that predictive models perform much better for test pairs that share components with the training set than for those that do not. Traditional cross-validation, however, overlooks this issue [104]. Hence, to prepare a training set with both positive and negative PPI pairs, a specific resampling process was conducted by [104] on the 6806 Yeast protein sequences to obtain 40 independent datasets. In each dataset, there are about 16000 PPI pairs for training and about 4000 PPI pairs (including C1, C2 and C3) for testing. It should be noted that the number of positive and negative PPIs is in equal amount. A simplified example dataset with training set and testing set C1, C2 and C3 is illustrated in Fig 5.3 with proteins existing in the training dataset in green and novel proteins not from the training dataset in red. The rationale for dividing the test set into three distinct classes is that, if not doing so, a test set may be dominated by pairs that share components with training pairs in the training set, yet such pairs may be a minority on the population level [104]. In other words, this is to assess if the classifier performs well

on pairs that are similar to the training pairs yet fails to generalize to the population level. The required generalization ability from the classifier increases with the number of novel protein sequences from C1 to C3.

5.3.2 Experimental design and parameter setting

As mentioned in section 5.3.1 Materials, we obtained in total 40 independent datasets provided by [104]. Each dataset has a training set of 16000 PPIs and a testing set of 4000 PPIs (80%-20% split). In our experiment, we first extracted features (Step 1, Step 2) from the training set, then used the features to construct a PPI matrix (Step 3, Step 4) and trained a predictive model based on the PPI matrix. In Step 1, we used WeMine Aligned Pattern Clustering algorithm [77, 143] to obtain APCs with length varying from 5 to 10 amino acids inclusively with the minimum support of 6, and the clustering threshold of 0.1. Other WeMine parameters remain default [77, 143]. We also trained 3000 trees in the Random Forest in Step 5 using Weka 3.7 [46]. Other Weka 3.7 parameters remain default [46]. We then transformed every PPI pair in the testing set into a feature vector (Step 6) and applied the trained model on it to output a class label and a score. We evaluated the predictive performance by computing the Area Under Curve (AUC) following [104] (see Table 5.3). We repeated the same procedure for all 40 independent datasets and computed the average AUC for comparison with Methods 1-7 in [104] (see Table 5.4).

5.3.3 Investigating the number of trees in the Random Forest

In the Step 5 of our Methodology, we used Random Forest as the prediction model and set the number of trees to be 3000. To illustrate that more trees would improve the prediction performance, we conducted two more experiments where one used 100 trees and another used 500 trees in the Random Forest. The results are shown in Table 5.1. We observe that WeMine-P2P achieved the best AUC performance when 3000 trees were used in the Random Forest. Hence, this setting was used throughout this study unless further specified.

5.3.4 Investigating the effectiveness of APC-PPI

APC-PPI measures the strength of an occurrence of a cAPC pair on a PPI pair. To investigate its effectiveness, we designed a new score, denoted as Random-APC-PPI. Given a cAPC pair $A_{i,j}$ composed of APCs C^i and C^j , and a PPI pair $B_{x,y}$, with Protein sequences S_x and S_y , we define Random-APC-PPI, i.e. $PPI_{random}(B_{x,y}, A_{i,j}) = \delta$, where δ is a

A simplified example dataset	
Training set:	Testing set C1
P01 – P02, +	P01 – P10, +
P01 – P03, +	P08 – P09, -
P07 – P08, +	
P09 – P10, +	Testing set C2
P01 – P07, -	P01 – P15, +
P02 – P09, -	P08 – P19, -
P03 – P07, -	
P03 – P10, -	Testing set C3
	P11 – P15, +
	P13 – P19, -

Figure 5.3: A simplified Protein-Protein Interaction sequence dataset example with a training set and a testing set with three distinct classes as defined in [104]. Each row is a pair of protein sequences with a class label. “+” means positive interactions and “-” means negative interactions. The positive PPI pairs are experimentally validated while the negative PPI pairs are sampled from the proteins within the same set that are not known to interact [103]. Proteins existing in the training dataset are in green and novel proteins not from the training dataset are in red. For example, in the training set, P01-P02 and P07-P08 are positive PPI pairs but P01-P07 is a negative PPI pair. All protein pairs in the testing sets are not found in the training set. However, all the protein sequences in C1 are in the training set, while in C2 only some protein sequences are in the training set, and in C3 no protein sequences are in the training set.

random value uniformly distributed in $[0, 1]$. We repeated the same experiment while using Random-APC-PPI instead of APC-PPI. The results are shown in Table 5.2. We observe that WeMine-P2P with APC-PPI outperformed the WeMine-P2P with Random-APC-PPI. This shows that APC-PPI is indeed effective for the predictive model construction.

5.3.5 Comparison to PIPE2

To illustrate the improvement made by WeMine-P2P on the use of co-occurring sequence segments, we compared the average AUC with those obtained by PIPE2, provided by [104]. Recall that PIPE2 [112, 109] uses the short amino acid sequences (fixed at length of 20) that co-occur frequently in given positive PPI pairs to make predictions on a testing PPI pair. As shown in Table 5.3, our results demonstrate that WeMine-P2P achieves better AUC performance in all three testing sets compared to PIPE2, indicating that WeMine-P2P outperformed PIPE2. WeMine-P2P is novel in the sense that 1) the length of sequence patterns is allowed to vary, coping with inherent functional association in the form of statistically significant patterns; 2) sequence patterns are clustered and aligned as Aligned Pattern Clusters (APCs) to relate to inherent functional conservation and variations; 3) nonlinear predictive models can then be trained with the feature vectors. Since WeMine-P2P has overcome the drawbacks of PIPE2, it does outperform it in the experiment.

5.3.6 Comparison to SVM-based Methods

To further illustrate the strength of WeMine-P2P, we compared its average AUC to the SVM-based methods that are well-known for achieving state-of-the-art predictive performance. The average AUC of SVM-based methods were obtained in [104]. As shown in Table 5.4, WeMine-P2P achieved comparable results, particularly for the testing sets C2 and C3, in which some testing protein sequences in C2 and all in C3 are new and not found in the training set (Fig. 5.3). For details please refer to section 5.3.1 Materials. This illustrates that WeMine-P2P has similar predictive power comparing to SVM-based methods for novel testing protein sequences. We have to point out that while assuming the pattern length $k = 3$, the feature dimension of SVM-based methods with Pairwise String Kernel [88, 51], though not computed nor stored, can be as large as $20^3 \times 20^3 = 64,000,000$. In WeMine-P2P, the feature dimension is only around 50,000, while allowing the variation of residues with the pattern length varying from 5 to 10. It is a potential reduction of 1280x in feature dimension. With such a large-scale reduction, the feature analysis of WeMine-P2P is much easier compared to that of a SVM using Pairwise String Kernel approaches. This

Table 5.1: Performance comparison of WeMine-P2P with different trees on the average Area Under Curve (AUC) among 40 independent datasets \pm the standard deviation

Number of Trees	Testing set C1	Testing set C2	Testing set C3
100	0.76 \pm 0.02	0.60 \pm 0.02	0.56 \pm 0.02
500	0.78 \pm 0.02	0.60 \pm 0.02	0.58\pm0.02
3000	0.79\pm0.02	0.61\pm0.02	0.58\pm0.02

Table 5.2: Performance comparison of WeMine-P2P with APC-PPI and Random-APC-PPI on the average Area Under Curve (AUC) among 40 independent datasets \pm the standard deviation

	Testing set C1	Testing set C2	Testing set C3
Random-APC-PPI	0.50 \pm 0.01	0.50 \pm 0.02	0.50 \pm 0.02
APC-PPI	0.79\pm0.02	0.61\pm0.02	0.58\pm0.02

would make biological knowledge discovery much easier. Also, while the feature vector is fixed in SVM-based methods, WeMine-P2P could extract features from the input data, allowing them to be biologically interpretable as described in the next section. Note that Methods 5 and 7 do not use SVM directly but are variants of SVM-based methods [104].

5.3.7 Analysis of the features with high *h*score

This section reports our investigation with high *h*score values discovered by WeMine-P2P, as shown in Tables 5.5 and 5.6. We focused our analysis on the training data in the independent dataset (ID = 11). In this dataset, we found about 250 APCs according to the default parameter setting. This means that we would have about $250 \times 250 = 50,000$ cAPC pairs. We adopted the *h*score defined in section 5.2. Methodology in order to compute a feature score (within -1 and 1 inclusively) for each feature (i.e.cAPC pair). The higher the score, the more likely the cAPC pair co-occur in positive PPI and less likely they co-occur in negative PPI. The features are ranked from the highest to lowest. The top

Table 5.3: Performance comparison of PIPE2 and WeMine-P2P on the average Area Under Curve (AUC) among 40 independent datasets \pm the standard deviation

Method	Testing set C1	Testing set C2	Testing set C3
Method 6 (PIPE2 [109, 112])	0.75 \pm 0.02	0.59 \pm 0.04	0.52 \pm 0.04
WeMine-P2P	0.79\pm0.02	0.61\pm0.02	0.58\pm0.02

Table 5.4: Performance comparison of SVM-based methods and WeMine-P2P on the average Area Under Curve (AUC) among 40 independent datasets \pm the standard deviation

Method	Testing set C1	Testing set C2	Testing set C3
Method 1 (SVM-SignatureProduct) [88, 51]	0.82 \pm 0.01	0.61\pm0.02	0.58 \pm 0.03
Method 2 (SVM-MetricLearningPairwiseKernel) [136]	0.84\pm0.01	0.60 \pm 0.02	0.59\pm0.03
Method 3 (SVM-ConjointTriadFeature) [121]	0.61 \pm 0.01	0.53 \pm 0.01	0.50 \pm 0.01
Method 4 (SVM-AutoCovariance) [43]	0.76 \pm 0.02	0.57 \pm 0.02	0.54 \pm 0.03
Method 5 (RF-AutoCovariance) [43]	0.80 \pm 0.01	0.58 \pm 0.01	0.55 \pm 0.02
Method 7 (catRAPID) [9]	0.58 \pm 0.02	0.54 \pm 0.02	0.52 \pm 0.03
WeMine-P2P	0.79 \pm 0.02	0.61\pm0.02	0.58 \pm 0.02

Table 5.5: Top 10 cAPC pairs in *hscore*

Description	1st APC ID	2nd APC ID	<i>hscore</i>
1st cAPC pair	1465525	9692312	0.018337
2nd cAPC pair	1465525	9698509	0.018083
3rd cAPC pair	1465525	1465525	0.018030
4th cAPC pair	1465525	9487593	0.017986
5th cAPC pair	1465525	9728806	0.017978
6th cAPC pair	1465525	8234623	0.017748
7th cAPC pair	1465525	9590335	0.017430
8th cAPC pair	1465525	9658538	0.017391
9th cAPC pair	8234623	9658538	0.017231
10th cAPC pair	9642970	9658538	0.017229

10 cAPC pairs are shown in Table 5.5 and their corresponding APCs are shown in Table 5.6. Here 10 is an arbitrary number, and it represents the top 0.02% of the features.

We observed that there are 9 unique APCs within the top 10 cAPC pairs, as shown Table 5.5. Among these 9 APCs, 8 of them are likely to represent a segment in the compositionally biased region, as shown in Table 5.6. For example, “AMAMAAMAMAMA” is a compositionally biased region in which “A” and “M” are enriched. According to [4], a compositionally biased region is composed of amino acids that have locally shifted frequencies, i.e. in some local regions, particular amino acids appear much more/less often than expected.

We observed a similar phenomenon in APC 1465525 (enriched for “A” and “Q”), APCs 9487593, 9692312, 8234623, 9658538 (enriched for “E”), APC 9642970 (enriched for “K” and “R”), and APC 9698509 (enriched for “K” and “E”), and APC 9728806 (enriched for “D” and “E”). These enriched regions can play important roles in PPI [135]. 1) the adaptation of organisms to extreme ecological niches [128]; 2) forming amyloids [3] or other cellular functions [50]; 3) determine certain properties of proteins [94].

In addition, computational bias regions can contribute directly in PPI. It is reported that some computational bias regions are strongly associated with intrinsically disordered sequences [135], which are found to be enriched for certain types of amino acids [27]. Intrinsically disordered sequences have no stable secondary and/or tertiary structure [27] but have the potential to associate with many partners due to multiple possible metastable conformations [90]. Those meditating regions are often to be SLiMs [29, 93], which are usually less than 10 amino acids in length [90]. These short interaction motifs may easily evolve and have the potential to rapidly change protein interactions and cellular signaling [90].

Table 5.6: APCs in the top 10 cAPC pairs

APC ID	APC in Regular Expression	Description
1465525	[AQ]QAQ[VA]	enriched for "A" and "Q"
9692312	[GADSN E ITV][VSANDG E LK][E GDIQPLFNTS] EE [NTASGQVLRKID][DLGEIKANQTSVRYF][DRKA]	enriched for "E"
9698509	[KDEVIL][EGLSKNV] E [VKE L QRIAF][KQREKTS][KQED]	enriched for "K" and "E"
9487593	[KIEALNV][KDENIA] E [LNSITVQRADK] E [QEK][LAQ]	enriched for "E"
9728806	[I EDNLSFG][TIDLNSEFGKQ][LIKFTD E V R DE][ANSID F E][TDV S ILKAYEQN][LAKLEDSQ][MD]	enriched for "D" and "E"
8234623	E [GLQTNKDSIVRAE] EEE [DE][GKQSTNRAL] E	enriched for "E"
9590335	[KSE][INE]VD[GLADKE][LD]	
9658538	[NL][DSEV] E [GVKDE] E [SGVDKE] EE	enriched for "E"
9642970	[RKE][KDIR][RAEKD][RK][LASE][ASKK]	enriched for "K" and "R"

Hence, allowing flexibility in binding segment length and residue variations is important to capture these signals.

By performing such an analysis, biologists not only can obtain a binary indication of PPI prediction but also get a sense of the type of sequence pattern pairs participating in PPI. This can assist them in subsequent experiment design and even provide hints on how to block interactions. It should be noted that all these pieces of knowledge can be discovered using only sequence data by WeMine-P2P without any a priori knowledge. This benefits biologists greatly if they wish to discover new knowledge about PPI beyond computational prediction.

5.4 Summary

In this chapter, we have furnished a new sequence-based Protein-Protein interaction prediction method WeMine-P2P that adopts interpretable biologically realistic features. We have demonstrated that our approach WeMine-P2P is not only able to yield superior or comparable predictive results but can also discover knowledge for PPIs through analyzing the interpretable discriminative features, to a certain degree, by a significant reduction of feature dimension. The knowledge discovered in the interpretable feature space can be useful for building better predictive models in the future. Through 40 independent experiments, we showed that (1) WeMine-P2P outperforms the well-known algorithm, PIPE2, which also utilizes co-occurring amino acid sequence segments but does not allow variable lengths and pattern variations; (2) WeMine-P2P achieves satisfactory PPI prediction performance, comparable to the SVM-based methods particularly among unseen protein sequences with a potential reduction of feature dimension of 1280x; (3) Unlike SVM-based methods, WeMine-P2P renders interpretable biological features from which we observed that co-occurring sequence patterns from the compositional bias regions are more discriminative. Since no prior information on PPI has been incorporated, WeMine-P2P is extendable to other biosequence applications in the future.

Chapter 6

Conclusion and Future Work

6.1 Contributions and Novelty

In this thesis, a new algorithm named Pattern-Directed Aligned Pattern Clustering (PD-APCn) was developed to discover and locate functional regions with mutations as APCs. To the best of our knowledge, these functional regions with mutations are difficult to be identified by existing algorithms, as illustrated by the results revealed in our experiments. Among all the experiments on the three synthetic datasets, where each of them has a different size and noise level, PD-APCn has consistently demonstrated higher recall and Fmeasure scores, when it was compared with the popular MEME. For operational efficiency, PD-APCn had a significant computational speed up (up to 665x) compared with the popular MEME. PD-APCn has also rendered consistently high performance among all datasets given indicating its robustness. In addition, PD-APCn also offers a succinct comprehensible display format of the output with direct traceable references to the pattern locations in sequences with known ids indexed by a suffix tree when compared with PWM-based approaches such as the popular MEME. Thus, PD-APCn is an effective, efficient, robust and comprehensive functional region identification algorithm. We believe that PD-APCn will play a significant role for the discovery of new functional regions from biosequences. This will be significant for drug discovery, administration and personalized medicine in the future.

For the application in Protein-DNA interaction sequences, a new model to represent Protein-DNA binding cores known as Protein-DNA Co-Occurring Aligned Pattern Clusters (APCs) was developed. This new model is more compact than one-to-one pattern associations, as it packs many-to-many associations in one model, yet detailed enough to

allow site-specific variants. Furthermore, an efficient algorithm was developed to discover Protein-DNA Co-Occurring APCs from Protein-DNA binding sequences. The discovery algorithm was faster than its counterpart by at least 1600x. On the biological aspect, a Protein-DNA Co-Occurring APC enables us to discover new Protein-DNA binding cores by pairing up the Protein patterns and DNA patterns within the APC. This can capture the Protein-DNA binding core candidates that do not co-occur as one-to-one mapped TF-TFBS associated patterns, an indication of its stronger discovery power. Two new Protein-DNA binding cores were discovered based on the follow-up homology modeling.

For the application in Protein-Protein interaction sequence, Aligned Pattern Clusters (APCs) [77, 143] were introduced to represent the co-occurring sequence patterns in Protein-Protein Interaction (PPI) (between two protein chains). This study demonstrates the first successful use of APCs in PPI, compared to the previous studies [77, 143, 73, 74]. Second, based on APCs, the novel co-occurring Aligned Pattern Cluster pairs (cAPC pairs) were used for modeling the co-occurring sequence patterns in PPI. Comparing to existing sequence-based prediction models, cAPC pairs are more biologically realistic because sequence patterns with variable length and variants are allowed. Third, using APC-PPIs to encode predictive features of PPI pair, a new PPI prediction system, WeMine-PPI, was developed. The experimental results demonstrated that WeMine-PPI outperformed PIPE2 [109, 112], which is a popular prediction algorithm based on co-occurring sequence patterns, and was comparable to the state-of-the-art SVM methods, while allowing a biologically intuitive understanding of the feature vector.

6.2 Limitations and Future Work

In this section, we discuss the current limitations of the proposed work and the potential extension in the future.

6.2.1 Comprehensive analysis of the parameter setting of breakpoint gap and seed width in PD-APCn

In Pattern-Directed Aligned Pattern Clustering (PD-APCn) algorithm, there are two important parameter settings, which are 1) seed width; and 2) breakpoint gap. As demonstrated in Table 3.2, Table 3.4 and 3.6, the performance achieved by PD-APCn is robust, particularly comparing with the popular MEME algorithm. For completeness, one direction is to extend the analysis to investigate the best parameter setting of seed width

and breakpoint gap, on an even larger scale of experiments. This study would help researchers better understand how these parameters can be optimized in different datasets in the future.

6.2.2 Discovering Protein-DNA Binding Cores from a new Protein-DNA interaction sequence database

Protein Binding Microarray (PBM) is a new Protein-DNA interaction sequence database leveraging high-throughput sequencing technologies. In addition to sequences given, binding intensity, measuring how strong the binding between each combination of the protein and a DNA sequence, is also provided in the database. An illustration of the database is provided in Fig 6.1. One possible extension would be to discover Protein-DNA Binding Cores from this new type of database.

6.2.3 Improving the Prediction Performance of WeMine-P2P

Although the prediction performance of WeMine-P2P is competitive against the state-of-the-art SVM-based method, there is still room for improvement. One direction to improve the prediction performance is by introducing additional features into the feature vector. One choice is the pairwise protein global similarity scores based on the popular BLOSUM62 matrix [28]. The rationale is that the global protein similarity information, which has not been adopted in the current model, can complement the protein similarity modeled by APCs. By introducing additional features, the prediction performance should be further enhanced.

6.2.4 Extending the representation of protein in other bioinformatics applications

In this thesis, a new representation of protein is developed by using Protein-Protein Co-Occurrence to encode as a feature vector via MaxMatchScore. Fig. 6.2 provides an illustration. A direction for future work is to extend this representation of protein in other bioinformatics applications. For example, MutationTaster [120] and SNPdryad [147] are computer programs to predict if a non-synonymous human single nucleotide polymorphism (SNP) [120, 147] is deleterious or not. These programs take a non-synonymous human SNP and the sequence of the human protein that the SNP is on as input, and then output a

Protein Binding Microarray (PBM) Data

ID	Protein	DNA (All possible 8-mers)	Intensity
01	RNDCE...QQQG	AAAAAAAA	10.1
		AAAAAAAC	1.2
	On average
	~500 residues	TTTTTTTT	2.5
02	GQERR...NNNG	AAAAAAAA	12.4
		AAAAAAAC	0.9
	On average
	~500 residues	TTTTTTTT	1.7
...			
86	RREQQ...MNGT	AAAAAAAA	0.7
		AAAAAAAC	12.9
	On average
	~500 residues	TTTTTTTT	0.2

Figure 6.1: An illustration of Protein Binding Microarray (PBM) data. Each row contains a unique protein sequence and all possible DNA sequences with 8 base pairs (8-mers). The binding intensity, measuring how strong the binding between each combination of the protein and a DNA sequence, is provided. The higher the value, the stronger the binding. On average, in the database, each protein has about 500 residues. There are in total 86 TFs in the database.

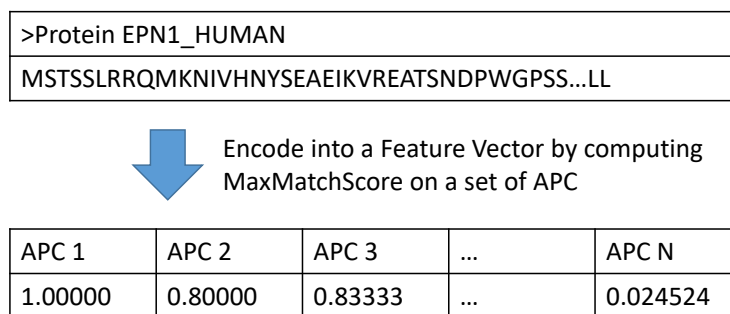


Figure 6.2: An illustration of encoding a protein sequence into a feature vector via APC and MaxMatchScore. The key idea here is that by computing MaxMatchScore of an APC on a protein sequence, a better score can be obtained. Repeating this step on a set of APCs, a list of real values can be obtained. This list of real values can then be a feature vector for machine learning.

score. The higher the score, the more deleterious the input nsSNP is predicted to be. These programs have only adopted the protein domains but not leveraged APC in their prediction models. Incorporating APCs as complementary information into these programs could potentially improve their performance.

6.3 Conclusion

Protein-DNA and Protein-Protein interactions have been studied for years but discovering new interaction knowledge remains challenging. Different types of biochemical experiments and computational methods have been proposed but each of them has their own limitations. In this thesis, we proposed a new sequence-based algorithm to discover functional regions with mutations as Aligned Pattern Clusters (APCs), and developed the use of Protein-DNA and Protein-Protein Co-Occurrence APC to capture co-occurrence functional regions in Protein-DNA and Protein-Protein interactions. Experimental results on both synthetic and real datasets validated their effectiveness and efficiency. We thus believe that our work can significantly contribute to advancing the frontiers in bioinformatics and biomedical research.

References

- [1] S. Ahmad, M.M. Gromiha, and A. Sarai. Analysis and prediction of dna-binding proteins and their binding residues based on composition, sequence and structural information. *Bioinformatics*, 20(4):477–486, 2004.
- [2] S. Ahmad, O. Keskin, A. Sarai, and R. Nussinov. Protein–DNA interactions: structural, thermodynamic and clustering patterns of conserved residues in DNA-binding proteins. *Nucleic acids research*, 36(18):5922–5932, 2008.
- [3] Simon Alberti, Randal Halfmann, Oliver King, Atul Kapila, and Susan Lindquist. A systematic survey identifies prions and illuminates sequence features of prionogenic proteins. *Cell*, 137(1):146–158, 2009.
- [4] Kirill S Antonets and Anton A Nizhnikov. sarp: A novel algorithm to assess compositional biases in protein sequences. *Evolutionary bioinformatics online*, 9:263, 2013.
- [5] Alberto Apostolico, Maxime Crochemore, Martin Farach-Colton, Zvi Galil, and S Muthukrishnan. 40 years of suffix trees. *Communications of the ACM*, 59(4):66–73, 2016.
- [6] Sharon E Ashbrook, John M Griffin, and Karen E Johnston. Recent advances in solid-state nuclear magnetic resonance spectroscopy. *Annual Review of Analytical Chemistry*, 11(1):485–508, 2018. PMID: 29324182.
- [7] Timothy L Bailey, James Johnson, Charles E Grant, and William S Noble. The meme suite. *Nucleic acids research*, 43(W1):W39–W49, 2015.
- [8] Paul D Barker and Stuart J Ferguson. Still a puzzle: why is haem covalently attached in c-type cytochromes? *Structure*, 7(12):R281–R290, 1999.

- [9] Matteo Bellucci, Federico Agostini, Marianela Masin, and Gian Gaetano Tartaglia. Predicting protein associations with long noncoding rnas. *Nature Methods*, 8(6):444–445, 2011.
- [10] Michael F Berger and Martha L Bulyk. Universal protein-binding microarrays for the comprehensive characterization of the dna-binding specificities of transcription factors. *Nature protocols*, 4(3):393–411, 2009.
- [11] Michael F Berger, Anthony A Philippakis, Aaron M Qureshi, Fangxue S He, Preston W Estep, and Martha L Bulyk. Compact, universal dna microarrays to comprehensively determine transcription-factor binding site specificities. *Nature biotechnology*, 24(11):1429–1435, 2006.
- [12] Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, Talapady N Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The protein data bank, 1999–. In *International Tables for Crystallography Volume F: Crystallography of biological macromolecules*, pages 675–684. Springer, 2006.
- [13] Ivano Bertini, Gabriele Cavallaro, and Antonio Rosato. Cytochrome c: occurrence and functions. *Chemical reviews*, 106(1):90–115, 2006.
- [14] Sarah EJ Bowman and Kara L Bren. The chemistry and biochemistry of heme c: functional bases for covalent attachment. *Natural product reports*, 25(6):1118–1130, 2008.
- [15] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
- [16] Michael J Buck and Jason D Lieb. Chip-chip: considerations for the design, analysis, and application of genome-wide chromatin immunoprecipitation experiments. *Genomics*, 83(3):349–360, 2004.
- [17] Tak-Ming Chan, Kwong-Sak Leung, Kin-Hong Lee, Man-Hon Wong, Terrence Chi-Kong Lau, and Stephen Kwok-Wing Tsui. Subtypes of associated protein–dna (transcription factor–transcription factor binding site) patterns. *Nucleic acids research*, 40(19):9392–9403, 2012.
- [18] Tak-Ming Chan, Gang Li, Kwong-Sak Leung, and Kin-Hong Lee. Discovering multiple realistic tfbs motifs based on a generalized model. *BMC bioinformatics*, 10(1):321, 2009.

- [19] Tak-Ming Chan, Leung-Yau Lo, Ho-Yin Sze-To, Kwong-Sak Leung, Xinshu Xiao, and Man-Hon Wong. Modeling associated protein-DNA pattern discovery with unified scores. *IEEE/ACM transactions on computational biology and bioinformatics/IEEE, ACM*, 10(3):696–707, 2013.
- [20] Tak-Ming Chan, Ka-Chun Wong, Kin-Hong Lee, Man-Hon Wong, Chi-Kong Lau, Stephen Kwok-Wing Tsui, and Kwong-Sak Leung. Discovering approximate-associated sequence patterns for protein–DNA interactions. *Bioinformatics*, 27(4):471–478, 2011.
- [21] Xue-Wen Chen and Mei Liu. Prediction of protein–protein interactions using random decision forest framework. *Bioinformatics*, 21(24):4394–4400, 2005.
- [22] ENCODE Project Consortium et al. An integrated encyclopedia of dna elements in the human genome. *Nature*, 489(7414):57–74, 2012.
- [23] Thomas Dandekar, Berend Snel, Martijn Huynen, and Peer Bork. Conservation of gene order: a fingerprint of proteins that physically interact. *Trends in biochemical sciences*, 23(9):324–328, 1998.
- [24] Francesca Diella, Niall Haslam, Claudia Chica, Aidan Budd, Sushama Michael, Nigel P Brown, Gilles Travé, and Toby J Gibson. Understanding eukaryotic linear motifs and their role in cell signaling and regulation. *Front Biosci*, 13:6580–6603, 2008.
- [25] Ivan Dikic, Soichi Wakatsuki, and Kylie J Walters. Ubiquitin-binding domains-from structures to functions. *Nature reviews Molecular cell biology*, 10(10):659, 2009.
- [26] Holger Dinkel, Sushama Michael, Robert J Weatheritt, Norman E Davey, Kim Van Roey, Brigitte Altenberg, Grischa Toedt, Bora Uyar, Markus Seiler, Aidan Budd, et al. Elmthe database of eukaryotic linear motifs. *Nucleic acids research*, page gkr1064, 2011.
- [27] A Keith Dunker, J David Lawson, Celeste J Brown, Ryan M Williams, Pedro Romero, Jeong S Oh, Christopher J Oldfield, Andrew M Campen, Catherine M Ratliff, Kerry W Hipps, et al. Intrinsically disordered protein. *Journal of Molecular Graphics and Modelling*, 19(1):26–59, 2001.
- [28] Sean R Eddy et al. Where did the blosum62 alignment score matrix come from? *Nature biotechnology*, 22(8):1035–1036, 2004.

- [29] Richard J Edwards, Norman E Davey, and Denis C Shields. Slimfinder: a probabilistic method for identifying over-represented, convergently evolved, short linear motifs in proteins. *PLoS one*, 2(10):e967, 2007.
- [30] Richard J Edwards and Nicolas Palopoli. Computational prediction of short linear motifs from protein sequences. In *Computational Peptidology*, pages 89–141. Springer, 2015.
- [31] Anton J Enright, Ioannis Iliopoulos, Nikos C Kyripides, and Christos A Ouzounis. Protein interaction maps for complete genomes based on gene fusion events. *Nature*, 402(6757):86–90, 1999.
- [32] Jason Ernst and Manolis Kellis. Chromhmm: automating chromatin-state discovery and characterization. *Nature methods*, 9(3):215–216, 2012.
- [33] Robert D Finn, Teresa K Attwood, Patricia C Babbitt, Alex Bateman, Peer Bork, Alan J Bridge, Hsin-Yu Chang, Zsuzsanna Dosztányi, Sara El-Gebali, Matthew Fraser, et al. Interpro in 2017-beyond protein family and domain annotations. *Nucleic acids research*, 45(D1):D190–D199, 2016.
- [34] Robert D Finn, Alex Bateman, Jody Clements, Penelope Coggill, Ruth Y Eberhardt, Sean R Eddy, Andreas Heger, Kirstie Hetherington, Liisa Holm, Jaina Mistry, et al. Pfam: the protein families database. *Nucleic acids research*, 42(D1):D222–D230, 2013.
- [35] Robert D Finn, Jody Clements, and Sean R Eddy. Hmmer web server: interactive sequence similarity searching. *Nucleic acids research*, 39(suppl_2):W29–W37, 2011.
- [36] Zoey L Fredericks and Gary J Pielak. Exploring the interface between the n-and c-terminal helices of cytochrome c by random mutagenesis within the c-terminal helix. *Biochemistry*, 32(3):929–936, 1993.
- [37] Martin C Frith, Ulla Hansen, John L Spouge, and Zhiping Weng. Finding functional sequence elements by multiple local alignment. *Nucleic acids research*, 32(1):189–200, 2004.
- [38] Martin C Frith, Neil FW Saunders, Bostjan Kobe, and Timothy L Bailey. Discovering sequence motifs with arbitrary insertions and deletions. *PLoS computational biology*, 4(5):e1000071, 2008.

- [39] Henry A Gabb, Richard M Jackson, and Michael JE Sternberg. Modelling protein docking using shape complementarity, electrostatics and biochemical information. *Journal of molecular biology*, 272(1):106–120, 1997.
- [40] Anne-Claude Gavin, Markus Bösch, Roland Krause, Paola Grandi, Martina Marzioch, Andreas Bauer, Jörg Schultz, Jens M Rick, Anne-Marie Michon, Cristina-Maria Cruciat, et al. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, 415(6868):141–147, 2002.
- [41] Alvaro J González and Li Liao. Predicting domain-domain interaction based on domain profiles with feature selection and support vector machines. *BMC bioinformatics*, 11(1):537, 2010.
- [42] Michael Goodson, Brian A Jonas, and Martin A Privalsky. Corepressors: custom tailoring and alterations while you wait. *Nuclear receptor signaling*, 3, 2005.
- [43] Yanzhi Guo, Lezheng Yu, Zhining Wen, and Menglong Li. Using support vector machine combined with auto covariance to predict protein–protein interactions from protein sequences. *Nucleic acids research*, 36(9):3025–3030, 2008.
- [44] Yuchun Guo, Shaun Mahony, and David K Gifford. High resolution genome wide binding event finding and motif discovery reveals transcription factor spatial binding constraints. *PLoS computational biology*, 8(8):e1002638, 2012.
- [45] Stephen J Hagen, Ramil F Latypov, Dimitry A Dolgikh, and Heinrich Roder. Rapid intrachain binding of histidine-26 and histidine-33 to heme in unfolded ferrocyanochrome c. *Biochemistry*, 41(4):1372–1380, 2002.
- [46] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.
- [47] Tobias Hamp and Burkhard Rost. Evolutionary profiles improve protein–protein interaction prediction from sequence. *Bioinformatics*, 31(12):1945–1950, 2015.
- [48] Andrew J Hampshire, David A Rusling, Victoria J Broughton-Head, and Keith R Fox. Footprinting: a method for determining the sequence selectivity, affinity and kinetics of dna-binding ligands. *Methods*, 42(2):128–140, 2007.
- [49] Israel Hanukoglu. Electron transfer proteins of cytochrome p450 systems. In *Advances in molecular and cell biology*, volume 14, pages 29–56. Elsevier, 1996.

- [50] Paul M Harrison. Exhaustive assignment of compositional bias reveals universally prevalent biased regions: analysis of functional associations in human and drosophila. *BMC bioinformatics*, 7(1):441, 2006.
- [51] Trevor Hastie, Robert Tibshirani, Jerome Friedman, and James Franklin. The elements of statistical learning: data mining, inference and prediction. *The Mathematical Intelligencer*, 27(2):83–85, 2005.
- [52] Lance M Hellman and Michael G Fried. Electrophoretic mobility shift assay (emsa) for detecting protein–nucleic acid interactions. *Nature protocols*, 2(8):1849–1861, 2007.
- [53] Yuen Ho, Albrecht Gruhler, Adrian Heilbut, Gary D Bader, Lynda Moore, Sally-Lin Adams, Anna Millar, Paul Taylor, Keiryn Bennett, Kelly Boutilier, et al. Systematic identification of protein complexes in *saccharomyces cerevisiae* by mass spectrometry. *Nature*, 415(6868):180–183, 2002.
- [54] Michael M Hoffman, Orion J Buske, Jie Wang, Zhiping Weng, Jeff A Bilmes, and William Stafford Noble. Unsupervised pattern discovery in human chromatin structure through genomic segmentation. *Nature methods*, 9(5):473–476, 2012.
- [55] Jianjun Hu, Bin Li, and Daisuke Kihara. Limitations and potentials of current motif discovery algorithms. *Nucleic acids research*, 33(15):4899–4913, 2005.
- [56] Lun Hu and Keith Chan. Discovering variable-length patterns in protein sequences for protein-protein interaction prediction. *IEEE Transactions on NanoBioscience*, 14(4), 2015.
- [57] Lin-Jun Huang, Hui-Ling Cao, Ya-Jing Ye, Yong-Ming Liu, Chen-Yan Zhang, Qin-Qin Lu, Hai Hou, Peng Shang, and Da-Chuan Yin. A new method to realize high-throughput protein crystallization in a superconducting magnet. *CrystEngComm*, 17(6):1237–1241, 2015.
- [58] Maxwell A Hume, Luis A Barrera, Stephen S Gisselbrecht, and Martha L Bulyk. Uniprobe, update 2015: new tools and content for the online database of protein-binding microarray data on protein–dna interactions. *Nucleic acids research*, page gku1045, 2014.
- [59] T Hunt. Protein sequence motifs involved in recognition and targeting: a new series. *Trends Biochem. Sci*, 15:305, 1990.

- [60] Takashi Ito, Tomoko Chiba, Ritsuko Ozawa, Mikio Yoshida, Masahira Hattori, and Yoshiyuki Sakaki. A comprehensive two-hybrid analysis to explore the yeast protein interactome. *Proceedings of the National Academy of Sciences*, 98(8):4569–4574, 2001.
- [61] Ronald Jansen, Haiyuan Yu, Dov Greenbaum, Yuval Kluger, Nevan J Krogan, Sambath Chung, Andrew Emili, Michael Snyder, Jack F Greenblatt, and Mark Gerstein. A bayesian networks approach for predicting protein-protein interactions from genomic data. *Science*, 302(5644):449–453, 2003.
- [62] Yanyan Jiang, Peter Oliver, Kay E Davies, and Nick Platt. Identification and characterization of murine *scara5*, a novel class a scavenger receptor that is expressed by populations of epithelial cells. *Journal of Biological Chemistry*, 281(17):11834–11845, 2006.
- [63] S. Jones, H.P. Shanahan, H.M. Berman, and J.M. Thornton. Using electrostatic potentials to predict dna-binding sites on dna-binding proteins. *Nucleic acids research*, 31(24):7189–7198, 2003.
- [64] Simon P Kanaan, Chengbang Huang, Stefan Wuchty, Danny Z Chen, and Jesús A Izaguirre. Inferring protein-protein interactions from multiple protein domain combinations. In *Computational Systems Biology*, pages 43–59. Springer, 2009.
- [65] A.E. Kel, E. Göbbling, I. Reuter, E. Chermushkin, O.V. Kel-Margoulis, and E. Wingender. Matchtm: a tool for searching transcription factor binding sites in dna sequences. *Nucleic acids research*, 31(13):3576–3579, 2003.
- [66] Motoo Kimura et al. Evolutionary rate at the molecular level. *Nature*, 217(5129):624–626, 1968.
- [67] Motoo Kimura and Tomoko Ohta. On some principles governing molecular evolution. *Proceedings of the National Academy of Sciences*, 71(7):2848–2852, 1974.
- [68] Jack Lester King and Thomas H Jukes. Non-darwinian evolution. *Science*, 164(3881):788–798, 1969.
- [69] Sonoko Kinjo, Norikazu Monma, Sadahiko Misu, Norikazu Kitamura, Junichi Imoto, Kazutoshi Yoshitake, Takashi Gojobori, and Kazuho Ikeo. Maser: one-stop platform for ngs big data from analysis to visualization. *Database*, 2018, 2018.

- [70] Bostjan Kobe, Gregor Guncar, Rebecca Buchholz, Thomas Huber, Bohumil Maco, Nathan Cowieson, JenniferL Martin, Mary Marfori, and JadeK Forwood. Crystallography and protein-protein interactions: biological interfaces and crystal contacts. *Biochemical Society Transactions*, 36(6):1438, 2008.
- [71] D.S. Latchman. Transcription factors: an overview. *The international journal of biochemistry & cell biology*, 29(12):1305–1312, 1997.
- [72] En-Shiun Annie Lee, Sanderz Fung, Ho-Yin Sze-To, and Andrew KC Wong. Confirming biological significance of co-occurrence clusters of aligned pattern clusters. In *Bioinformatics and Biomedicine (BIBM), 2013 IEEE International Conference on*, pages 422–427. IEEE, 2013.
- [73] En-Shiun Annie Lee, Sanderz Fung, Ho-Yin Sze-To, and Andrew KC Wong. Discovering co-occurring patterns and their biological significance in protein families. *BMC bioinformatics*, 15(12):S2, 2014.
- [74] En-Shiun Annie Lee, Kwong-Sak Leung, Ho-Yin Sze-To, Terrence Chi-Kong Lau, Man-Hon Wong, and Andrew KC Wong. Discovering protein-dna binding cores by aligned pattern clustering. In *Bioinformatics and Biomedicine (BIBM), 2014 IEEE International Conference on*, pages 125–130. IEEE, 2014.
- [75] En-Shiun Annie Lee, Ho-Yin Antonio Sze-To, Man-Hon Wong, Kwong-Sak Leung, Terrence Chi-Kong Lau, and Andrew KC Wong. Discovering protein-dna binding cores by aligned pattern clustering. *IEEE/ACM transactions on computational biology and bioinformatics*, 14(2):254–263, 2017.
- [76] En-Shiun Annie Lee, Fiona J Whelan, Dawn ME Bowdish, and Andrew KC Wong. Partitioning and correlating subgroup characteristics from aligned pattern clusters. *Bioinformatics*, 32(16):2427–2434, 2016.
- [77] En-Shiun Annie Lee and Andrew KC Wong. Ranking and compacting binding segments of protein families using aligned pattern clusters. *Proteome science*, 11(1):S8, 2013.
- [78] Christina Leslie and Rui Kuang. Fast string kernels using inexact matching for protein sequences. *Journal of Machine Learning Research*, 5(Nov):1435–1455, 2004.
- [79] Kwong-Sak Leung, Ka-Chun Wong, Tak-Ming Chan, Man-Hon Wong, Kin-Hong Lee, Chi-Kong Lau, and Stephen KW Tsui. Discovering protein–dna binding sequence

- patterns using association rule mining. *Nucleic acids research*, 38(19):6324–6337, 2010.
- [80] Weizhong Li and Adam Godzik. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*, 22(13):1658–1659, 2006.
- [81] Olivier Lichtarge, Henry R Bourne, and Fred E Cohen. An evolutionary trace method defines binding surfaces common to protein families. *Journal of molecular biology*, 257(2):342–358, 1996.
- [82] X Liu, Douglas L Brutlag, Jun S Liu, et al. BioProspector: discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes. In *Pac Symp Biocomput*, volume 6, pages 127–138, 2001.
- [83] Xin Luo, Zhuhong You, Mengchu Zhou, Shuai Li, Hareton Leung, Yunni Xia, and Qingsheng Zhu. A highly efficient approach to protein interactome mapping based on collaborative filtering framework. *Scientific reports*, 5, 2015.
- [84] Nicholas M Luscombe and Janet M Thornton. Protein–dna interactions: amino acid conservation and the effects of mutations on binding specificity. *Journal of molecular biology*, 320(5):991–1009, 2002.
- [85] N.M. Luscombe, S.E. Austin, H.M. Berman, and J.M. Thornton. An overview of the structures of protein-dna complexes. *Genome biology*, 1(1):reviews001, 2000.
- [86] K.D. MacIsaac and E. Fraenkel. Practical strategies for discovering regulatory dna sequence motifs. *PLoS computational biology*, 2(4):e36, 2006.
- [87] Rebecca EK MacPherson, Sofia V Ramos, Rene Vandenboom, Brian D Roy, and Sandra J Peters. Skeletal muscle plin proteins, atgl and cgi-58, interactions at rest and following stimulated contraction. *American Journal of Physiology-Regulatory, Integrative and Comparative Physiology*, 304(8):R644–R650, 2013.
- [88] Shawn Martin, Diana Roe, and Jean-Loup Faulon. Predicting protein–protein interactions using signature products. *Bioinformatics*, 21(2):218–226, 2005.
- [89] Volker Matys, Olga V Kel-Margoulis, Ellen Fricke, Ines Liebich, Sigrid Land, A Barre-Dirrie, Ingmar Reuter, D Chekmenev, Mathias Krull, Klaus Hornischer, et al. Transfac® and its module transcompel®: transcriptional gene regulation in eukaryotes. *Nucleic acids research*, 34(suppl_1):D108–D110, 2006.

- [90] Floriane Montanari, Denis C Shields, and Nora Khaldi. Differences in the number of intrinsically disordered regions between yeast duplicated proteins, and their relationship with functional divergence. *PloS one*, 6(9):e24989, 2011.
- [91] Alexandre V Morozov, James J Havranek, David Baker, and Eric D Siggia. Protein–dna binding specificity predictions with structural models. *Nucleic acids research*, 33(18):5781–5798, 2005.
- [92] Anders M Näär, Bryan D Lemon, and Robert Tjian. Transcriptional coactivator complexes. *Annual review of biochemistry*, 70(1):475–501, 2001.
- [93] Victor Neduva and Robert B Russell. Peptides mediating interaction networks: new leads at last. *Current opinion in biotechnology*, 17(5):465–471, 2006.
- [94] Victor Neduva and Robert B Russell. Proline-rich regions in transcriptional complexes: heading in many directions. *Science Signaling*, 2007(369):pe1–pe1, 2007.
- [95] Andrew F Neuwald, Jun S Liu, and Charles E Lawrence. Gibbs motif sampling: detection of bacterial outer membrane protein repeats. *Protein science*, 4(8):1618–1632, 1995.
- [96] H Chau Nguyen, Riccardo Zecchina, and Johannes Berg. Inverse statistical problems: from the inverse ising problem to data science. *Advances in Physics*, 66(3):197–261, 2017.
- [97] Irene MA Nooren and Janet M Thornton. Diversity of protein–protein interactions. *The EMBO journal*, 22(14):3486–3492, 2003.
- [98] Clare M OConnor, Jill U Adams, and Jennifer Fairman. Essentials of cell biology. *Cambridge, MA: NPG Education*, 1, 2010.
- [99] Y. Ofran, V. Mysore, and B. Rost. Prediction of dna-binding residues from sequence. *Bioinformatics*, 23(13):i347–i353, 2007.
- [100] Yanay Ofran and Burkhard Rost. Isis: interaction sites identified from sequence. *Bioinformatics*, 23(2):e13–e16, 2007.
- [101] Mikhail Pachkov, Ionas Erb, Nacho Molina, and Erik Van Nimwegen. Swissregulon: a database of genome-wide annotations of regulatory sites. *Nucleic acids research*, 35(suppl 1):D127–D131, 2007.

- [102] Peter J Park. Chip-seq: advantages and challenges of a maturing technology. *Nature Reviews Genetics*, 10(10):669–680, 2009.
- [103] Yungki Park and Edward M Marcotte. Revisiting the negative example sampling problem for predicting protein-protein interactions. *Bioinformatics*, 27(21):3024–3028, 2011.
- [104] Yungki Park and Edward M Marcotte. Flaws in evaluation schemes for pair-input computational predictions. *Nature methods*, 9(12):1134, 2012.
- [105] Jodi R Parrish, Keith D Gulyas, and Russell L Finley. Yeast two-hybrid contributions to interactome mapping. *Current opinion in biotechnology*, 17(4):387–393, 2006.
- [106] Matteo Pellegrini, Edward M Marcotte, Michael J Thompson, David Eisenberg, and Todd O Yeates. Assigning protein functions by comparative genome analysis: protein phylogenetic profiles. *Proceedings of the National Academy of Sciences*, 96(8):4285–4288, 1999.
- [107] James R Perkins, Ilhem Diboun, Benoit H Dessailly, Jon G Lees, and Christine Orengo. Transient protein-protein interactions: structural, functional, and network properties. *Structure*, 18(10):1233–1243, 2010.
- [108] Brian G Pierce, Kevin Wiehe, Howook Hwang, Bong-Hyun Kim, Thom Vreven, and Zhiping Weng. Zdock server: interactive docking prediction of protein-protein complexes and symmetric multimers. *Bioinformatics*, 30(12):1771–1773, 2014.
- [109] Sea Pitre, C North, Md Alamgir, Matthew Jessulat, Andrew Chan, Xiaobin Luo, James R Green, M Dumontier, Frank Dehne, and Ashkan Golshani. Global investigation of protein-protein interactions in yeast *saccharomyces cerevisiae* using re-occurring short polypeptide sequences. *Nucleic acids research*, 36(13):4286–4294, 2008.
- [110] Sylvain Pitre, Md Alamgir, James R Green, Michel Dumontier, Frank Dehne, and Ashkan Golshani. Computational methods for predicting protein-protein interactions. In *Protein-Protein Interaction*, pages 247–267. Springer, 2008.
- [111] Sylvain Pitre, Frank Dehne, Albert Chan, Jim Cheetham, Alex Duong, Andrew Emili, Marinella Gebbia, Jack Greenblatt, Mathew Jessulat, Nevan Krogan, et al. Pipe: a protein-protein interaction prediction engine based on the re-occurring short polypeptide sequences between known interacting protein pairs. *BMC bioinformatics*, 7(1):365, 2006.

- [112] Sylvain Pitre, Mohsen Hooshyar, Andrew Schoenrock, Bahram Samanfar, Matthew Jessulat, James R Green, Frank Dehne, and Ashkan Golshani. Short co-occurring polypeptide regions can predict global protein interaction maps. *Scientific reports*, 2:239, 2012.
- [113] Roman Prytuliak, Michael Volkmer, Markus Meier, and Bianca H Habermann. Hh-motif: de novo detection of short linear motifs in proteins by hidden markov model comparisons. *Nucleic acids research*, 45(W1):W470–W477, 2017.
- [114] Joseph J Rasimas, Sambit R Kar, Anthony E Pegg, and Michael G Fried. Interactions of human o6-alkylguanine-dna alkyltransferase (agt) with short single-stranded dnas. *Journal of Biological Chemistry*, 282(5):3357–3366, 2007.
- [115] Jacques E Remacle, Gerd Albrecht, Reginald Brys, Gerhard H Braus, and Danny Huylebroeck. Three classes of mammalian transcription activation domain stimulate transcription in *Schizosaccharomyces pombe*. *The EMBO journal*, 16(18):5722–5729, 1997.
- [116] Rolando Rodriguez, Glay Chinaea, Nelia Lopez, Tirso Pons, and Gert Vriend. Homology modeling, model and software evaluation: three related resources. *Bioinformatics (Oxford, England)*, 14(6):523–528, 1998.
- [117] Remo Rohs, Xiangshu Jin, Sean M West, Rohit Joshi, Barry Honig, and Richard S Mann. Origins of specificity in protein-dna recognition. *Annual review of biochemistry*, 79:233, 2010.
- [118] A. Sarai and H. Kono. Protein-DNA recognition patterns and predictions. *Annu. Rev. Biophys. Biomol. Struct.*, 34:379–398, 2005.
- [119] Ulf Schaefer, Sebastian Schmeier, and Vladimir B Bajic. Tcof-db: dragon database for human transcription co-factors and transcription factor interacting proteins. *Nucleic acids research*, 39(suppl 1):D106–D110, 2011.
- [120] Jana Marie Schwarz, Christian Rödelsperger, Markus Schuelke, and Dominik Seelow. Mutationtaster evaluates disease-causing potential of sequence alterations. *Nature methods*, 7(8):575–576, 2010.
- [121] Juwen Shen, Jian Zhang, Xiaomin Luo, Weiliang Zhu, Kunqian Yu, Kaixian Chen, Yixue Li, and Hualiang Jiang. Predicting protein–protein interactions based only on sequences information. *Proceedings of the National Academy of Sciences*, 104(11):4337–4341, 2007.

- [122] Tatiana Skarina, Xiaohui Xu, Elena Evdokimova, and Alexei Savchenko. High-throughput crystallization screening. In *Structural Genomics and Drug Discovery*, pages 159–168. Springer, 2014.
- [123] Gary D Stormo and Yue Zhao. Determining the specificity of protein–dna interactions. *Nature Reviews Genetics*, 11(11):751–760, 2010.
- [124] Antonio Sze-To, Sanderz Fung, En-Shiun Annie Lee, and Andrew KC Wong. Predicting protein-protein interaction using co-occurring aligned pattern clusters. In *Bioinformatics and Biomedicine (BIBM), 2015 IEEE International Conference on*, pages 55–60. IEEE, 2015.
- [125] Antonio Sze-To, Sanderz Fung, En-Shiun Annie Lee, and Andrew KC Wong. Prediction of protein–protein interaction via co-occurring aligned pattern clusters. *Methods*, 110:26–34, 2016.
- [126] Antonio Sze-To and Andrew KC Wong. Pattern-directed aligned pattern clustering. In *Bioinformatics and Biomedicine (BIBM), 2017 IEEE International Conference on*, pages 28–35. IEEE, 2017.
- [127] Antonio Sze-To and Andrew KC Wong. Discovering patterns from sequences using pattern-directed aligned pattern clustering. *IEEE/ACM transactions on nanobioscience*, 2018.
- [128] Xavier Tadeo, Blanca López-Méndez, Tamara Trigueros, Ana Laín, David Castaño, and Oscar Millet. Structural basis for the aminoacid composition of proteins from halophilic archea. *PLoS biology*, 7(12):2821, 2009.
- [129] Tsunehiro Takano and Richard E Dickerson. Conformation change of cytochrome c: I. ferrocyanochrome c structure refined at 1· 5 Å resolution. *Journal of molecular biology*, 153(1):79–94, 1981.
- [130] Denis Tempé, Muriel Brengues, Pauline Mayonove, Hayat Bensaad, Céline Lacrouts, and May C Morris. The alpha helix of ubiquitin interacts with yeast cyclin-dependent kinase subunit cks1. *Biochemistry*, 46(1):45–54, 2007.
- [131] Markus F Templin, Dieter Stoll, Jochen M Schwenk, Oliver Pötz, Stefan Kramer, and Thomas O Joos. Protein microarrays: promising tools for proteomic research. *Proteomics*, 3(11):2155–2166, 2003.

- [132] Julie D Thompson, Benjamin Linard, Odile Lecompte, and Olivier Poch. A comprehensive benchmark study of multiple sequence alignment methods: current challenges and future perspectives. *PloS one*, 6(3):e18093, 2011.
- [133] Martin Tompa, Nan Li, Timothy L Bailey, George M Church, Bart De Moor, Eleazar Eskin, Alexander V Favorov, Martin C Frith, Yutao Fu, W James Kent, et al. Assessing computational tools for the discovery of transcription factor binding sites. *Nature biotechnology*, 23(1):137, 2005.
- [134] Andrew P Turnbull, Susan M Boyd, and Björn Walse. Fragment-based drug discovery and protein–protein interactions. *Research and Reports in Biochemistry*, 4:13–26, 2014.
- [135] Vladimir N Uversky and A Keith Dunker. Understanding protein non-folding. *Biochimica et Biophysica Acta (BBA)-Proteins and Proteomics*, 1804(6):1231–1264, 2010.
- [136] Jean-Philippe Vert, Jian Qiu, and William S Noble. A new pairwise kernel for biological network inference with support vector machines. *BMC bioinformatics*, 8(Suppl 10):S8, 2007.
- [137] Senadhi Vijay-Kumar, Charles E Bugg, Keith D Wilkinson, and William J Cook. Three-dimensional structure of ubiquitin at 2.8 a resolution. *Proceedings of the National Academy of Sciences*, 82(11):3582–3585, 1985.
- [138] Martin Vingron and Michael S Waterman. Sequence alignment and penalty choice: Review of concepts, case studies and implications. *Journal of molecular biology*, 235(1):1–12, 1994.
- [139] Dianhui Wang and Sarwar Tapan. A robust elicitation algorithm for discovering DNA motifs using fuzzy self-organizing maps. *IEEE Transactions on Neural Networks and Learning Systems*, 24(10):1677 – 1688, 2013.
- [140] Fiona J Whelan, Conor J Meehan, G Brian Golding, Brendan J McConkey, and Dawn ME Bowdish. The evolution of the class a scavenger receptors. *BMC evolutionary biology*, 12(1):227, 2012.
- [141] David S Wilson, Guojun Sheng, Susie Jun, and Claude Desplan. Conservation and diversification in homeodomain-DNA interactions: a comparative genetic analysis. *Proceedings of the National Academy of Sciences*, 93(14):6886–6891, 1996.

- [142] Christof Winter, Andreas Henschel, Anne Tuukkanen, and Michael Schroeder. Protein interactions in 3d: from interface evolution to drug discovery. *Journal of structural biology*, 179(3):347–358, 2012.
- [143] Andrew KC Wong and En-Shiun Annie Lee. Aligning and clustering patterns to reveal the protein functionality of sequences. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 11(3):548–560, 2014.
- [144] Andrew KC Wong, Dennis Zhuang, Gary CL Li, and En-Shiun Annie Lee. Discovery of delta closed patterns and noninduced patterns from sequences. *Knowledge and Data Engineering, IEEE Transactions on*, 24(8):1408–1421, 2012.
- [145] Ka-Chun Wong, Yue Li, Chengbin Peng, and Zhaolei Zhang. Signalspider: probabilistic pattern discovery on multiple normalized chip-seq signal profiles. *Bioinformatics*, 31(1):17–24, 2015.
- [146] Ka-Chun Wong, Chengbin Peng, Man-Hon Wong, and Kwong-Sak Leung. Generalizing and learning protein-dna binding sequence representations by an evolutionary algorithm. *Soft Computing*, 15(8):1631–1642, 2011.
- [147] Ka-Chun Wong and Zhaolei Zhang. Snpdryad: predicting deleterious non-synonymous human snps using only orthologous protein sequences. *Bioinformatics*, 30(8):1112–1119, 2014.
- [148] Po-Yuen Wong, Tak-Ming Chan, Man-Hon Wong, and Kwong-Sak Leung. Predicting approximate protein-DNA binding cores using association rule mining. In *Data Engineering (ICDE), 2012 IEEE 28th International Conference on*, pages 965–976. IEEE, 2012.
- [149] MM Woolfson. The development of structural x-ray crystallography. *Physica Scripta*, 93(3):032501, 2018.
- [150] Jianmin Wu, Tea Vallenius, Kristian Ovaska, Jukka Westermarck, Tomi P Mäkelä, and Sampsa Hautaniemi. Integrated network analysis platform for protein-protein interactions. *Nature methods*, 6(1):75–77, 2009.
- [151] Xuhua Xia. Position weight matrix, gibbs sampler, and the associated significance tests in motif characterization and prediction. *Scientifica*, 2012, 2012.
- [152] Zhu-Hong You, Keith CC Chan, and Pengwei Hu. Predicting protein-protein interactions from primary protein sequences using a novel multi-scale local feature representation scheme and the random forest. *PLoS ONE*, 10(5), 2015.

- [153] Zhu-Hong You, Ying-Ke Lei, Jie Gui, De-Shuang Huang, and Xiaobo Zhou. Using manifold embedding for assessing and predicting protein interactions from high-throughput experimental data. *Bioinformatics*, 26(21):2744–2751, 2010.
- [154] Sobia Zaidi, Md Imtaiyaz Hassan, Asimul Islam, and Faizan Ahmad. The role of key residues in structure, function, and stability of cytochrome-c. *Cellular and molecular life sciences*, 71(2):229–255, 2014.
- [155] Fengfeng Zhou, Victor Olman, and Ying Xu. Large-scale analyses of glycosylation in cellulases. *Genomics, proteomics & bioinformatics*, 7(4):194–199, 2009.
- [156] Pei-Yuan Zhou, Antonio Sze-Tzo, and Andrew KC Wong. Discovery and disentanglement of protein aligned pattern clusters to reveal subtle functional subgroups. In *Bioinformatics and Biomedicine (BIBM), 2017 IEEE International Conference on*, pages 62–69. IEEE, 2017.