Transparent Decision Support Using Statistical Evidence

by

Andrew Michael Hamilton-Wright

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, 2005

I hereby declare that I am the sole author of this 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.

Abstract

An automatically trained, statistically based, fuzzy inference system that functions as a classifier is produced. The hybrid system is designed specifically to be used as a decision support system. This hybrid system has several features which are of direct and immediate utility in the field of decision support, including a mechanism for the discovery of domain knowledge in the form of explanatory rules through the examination of training data; the evaluation of such rules using a simple probabilistic weighting mechanism; the incorporation of input uncertainty using the vagueness abstraction of fuzzy systems; and the provision of a strong confidence measure to predict the probability of system failure.

Analysis of the hybrid fuzzy system and its constituent parts allows commentary on the weighting scheme and performance of the "Pattern Discovery" system on which it is based.

Comparisons against other well known classifiers provide a benchmark of the performance of the hybrid system as well as insight into the relative strengths and weaknesses of the compared systems when functioning within continuous and mixed data domains.

Classifier reliability and confidence in each labelling are examined, using a selection of both synthetic data sets as well as some standard real-world examples.

An implementation of the work-flow of the system when used in a decision support context is presented, and the means by which the user interacts with the system is evaluated.

The final system performs, when measured as a classifier, comparably well or better than other classifiers. This provides a robust basis for making suggestions in the context of decision support.

The adaptation of the underlying statistical reasoning made by casting it into a fuzzy inference context provides a level of transparency which is difficult to match in decision support. The resulting linguistic support and decision exploration abilities make the system useful in a variety of decision support contexts.

Included in the analysis are case studies of heart and thyroid disease data, both drawn from the University of California, Irvine Machine Learning repository.

Acknowledgements

I'd like to thank Dan Stashuk for support, patience and many enlightening discussions throughout this work. Without these, this work would be quite different, and have significantly less immediate application.

Tim Doherty has greatly aided my understanding of the clinical environment, to which this work is destined, and has thereby significantly assisted in the choosing of the goal.

Andrew Wong and Yang Wang deserve significant thanks, not only for their ideas regarding the Pattern Discovery algorithm, but also for the use of some of the software from Pattern Discovery Inc. which was used to verify the output of the routines developed here.

Hamid Tizhoosh has provided considerable experience and insight into the functioning of the fuzzy systems upon which this work is based and has introduced me to several new ideas within this arena.

I would like to thank the other members of my committee for their input and suggestions regarding the final form of this document and the arguments therein; without these ideas, the work would be significantly poorer.

I would like to sincerely thank my parents, family and friends for their support and encouragement while pursuing a goal which I know seemed like a crazy one to many people.

Perhaps most significantly however: thanks to Kirsty, for making all this seem like a good idea.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

> *The more I learn, the more I realize I don't know. The more I realize I don't know, the more I want to learn.*
>
> — Albert Einstein

The pattern extraction techniques of the "Pattern Discovery" (PD) algorithm developed by Wang and Wong (Wang, 1997; Wong and Wang, 1997) is extended into the fuzzy inference domain to form a decision support system[*] (DSS).

The classification performance of the resulting fuzzy hybrid system is higher than that of PD alone and a metric to characterize the confidence in each decision can be formed, producing a decision support system that allows a transparent explanation of the decision process.

1.0.1 Rationale

Using a fuzzy inference system (FIS) as the basis of a decision support tool will allow transparent decisions to be suggested using a linguistic framework based on sound statistical data. Such a system may be used with a wide spectrum of data types from financial to resource management, however the application of immediate interest to the author is the area of clinical diagnostic support. For this reason, real-world data from two bio-medical domains are used in the enclosed analysis, and the presentation of the final system centres on the discussion of heart disease data acquired from the well-known machine learning repository

[*] An index is provided at the end of the manuscript linking all key terms to their locations in the text.

at the University of California, Irvine.[†]

This document describes the production, function and performance of an FIS based DSS which uses adapted rules extracted using the PD algorithm. The system and its evaluation tools have been written in the C++ and Python programming languages by the author, with some dependency on the GNU Scientific Library (Galassi, Davies *et al.*, 2005) and the routines in LAPACK (Anderson, Bai *et al.*, 1999). All design and implementation efforts outside of these libraries have been the author's own.

1.1 Decision Support

The use of decision support systems is an area which has been studied for more than twenty years with application areas as diverse as finance, emergency response, environmental management and many medical applications.

Underlying all work in decision support is an interest in the management and reporting of the estimated reliability of the decision, measured by the probability of successfully indicating the correct label (see Larsson, Hayes-Roth *et al.*, 1997; López de Mántaras, 1991; Aha, Kibler and Albert, 1991; Cordella, Foggia *et al.*, 1999; Levitin, 2002, 2003; Gurov, 2004, 2005). Without a measure of the reliability of a system, the suggested analysis is useless, as graceful failure cannot be assured in the face of variable quality data and inference. It is therefore critical that the system presented in this work is evaluated in terms of the reliability of the decisions. Reliability measurement and system confidence prediction will be presented in Chapter 2, and the quality of the confidence measure used within the hybrid system described in this work will be discussed in depth in Chapter 9.

1.2 Decision Support

Decision support is a difficult field to define. In order to avoid spending many pages attempting to produce a definition, we can content ourselves with that provided by Silver (1991, pp.13):

> *A decision support system is a computer-based information system that supports people engaged in decision-making activities.*
>
> *In this context, "'support"' is intended to mean that*

[†] The famous UCI machine learning repository contains "real-world" data used for comparing techniques within the machine learning community. It is available online at `http://www.ics.uci.edu/~mlearn/MLRepository.html`.

1. *the system assists human decision makers to exercise judgement—that is, the system is an aid for the person or persons making the decision; and*
2. *the system does not make the decision—that is, the system helps decision makers exercise judgement but does not replace the human decision makers.*

Both of these points are crucial to the design of the hybrid system described in this work.

Silver's first point indicates that the system design is intended to ease the making of a decision; this implies that the format of the decision support data must be driven by the need for a human being to easily comprehend the suggestions, rapidly understand its import and assimilate the data with possibly many other data elements at their disposal at the time the decision is to be made.

For example, in heart disease, there are a large number of symptoms and markers for disease which must be taken into account for any single patient. Importantly, these markers change depending on a patient's age and other factors. It would be reasonable then for a DSS to correlate and summarize the patient's data, highlighting the strongest and most informative markers. Data which is not relevant to the decision will be shown in only a cursory way, or not at all. By following such a methodology, a decision making user is given the information they need to make an informed decision without the tedium involved in a manual collation and exploration. In this way, a software tool can support decision making.

The second point speaks to the important design principle that the decision support system is *not authoritative*. At all times it must be kept in mind that any decision support system cannot function as a black box, but must instead be as transparent and interpretable as possible, assisting in the formation of a judgement. As part of this transparency, it is therefore important that a user may explore alternate decision paths in order to come to a comfortable, informed decision of their own making, evaluating the data from any particular source within the DSS in context with the data available from all other sources.

1.2.1 History

The field of decision support began with the needs of Management Information Systems (MIS) in the 1970's. Several works (Morton, 1971; Sprague and Carlsons, 1982; House, 1983; Schniederjans, 1987) describe the then growing need for middle management to have access to computer based systems to interpret the ever larger quantities of data. Previously, data for management decision making was available in (daily or weekly) printed reports, which were produced and collated with the expectation that the reader would absorb the presented data with enough depth of understanding to form reasoned and informed opinions on the contents.

The subsequent expansion of both business and its management along with the now familiar "information explosion" drove the need for a new tool which could condense a great deal of reporting data and characterize it in terms of possible courses of action. Using such a tool, the data previously available in the reports would be linked to a course of action in terms of degrees of support for that particular course. A manager using this type of tool could then construct a business plan, taking into account the recommended actions suggested by the DSS.

Once a tool of this type was created, it was obvious that its the applicability extended far beyond the role of middle-management supply-line and financial decision making in a mid-size company.

Current common areas of application of decision support systems include:

Conflict Resolution and Generalized Decision Making: a field which produces general tools using the concepts of decision support. These tools are as general in approach and therefore as universal in application as possible (Hipel, Fang and Kilgour, 1993; Kilgour, Fang and Hipel, 1995; Rajabi, Kilgour and Hipel, 1998; Sage and Rouse, 1999; Hipel, Kilgour *et al.*, 2001).

Environmental Management: including mapping and management of toxins (Booty, Lam *et al.*, 1997) and watershed management (Hipel, Yin and Kilgour, 1995; León, Lam *et al.*, 1997; Young, Lam *et al.*, 1997; Yyrdusev, 1997; Hipel and Ben-Haim, 1999). This area has a great deal of ongoing research, as shown by recent workshops (Cortés and Sànchez-Marrè, 1999; Cortés, Sànchez-Marrè and Wotawa, 2003),

Medical Decision Making and Disease Characterization: a field in which common areas of application are laboratory data management (Cowan, 2003), patient monitoring (Gibb, Auslander and Griffin, 1994; de Graaf, van den Eijkel *et al.*, 1997; Abu-Hanna and de Keizer, 2003; Montani, Magni *et al.*, 2003), public health (O' Carroll, Yasnoff *et al.*, 2002) and disease characterization or evaluation of prognoses (Shortliffe and Perreault, 1990; Friedland, 1998; Kukar, Kononenko *et al.*, 1999; Innocent, 2000a,b; Colombet, Dart *et al.*, 2003; Coiera, 2003). A new and growing area in which this technology is finding application is in modelling outbreaks, providing a recognition tool for fast response (Penaloza and Welch, 1997; Zhang, Fiedler and Popovich, 2004; Brillman, Burr *et al.*, 2005; Costa, Dunyak and Mohtashemi, 2005; Devadoss, Pan and Singh, 2005; Guthrie, Stacey and Calvert, 2005; Majowicz and Stacey, 2005).

Business Management Information Systems: this is still a major application area for decision support, and many texts provide discussion of this area of application (see any of Scott, Claton and Gibson, 1991; Adelman, 1992; Sage, 1991).

1.2.2 Medical Decision Support

Medical decision support literature ranges from discussions training physicians in the underlying data tools needed to understand decision support (Harris and Boyd, 1995; Kononenko, 2001; Kukar, 2003; Bennett, Casebeer *et al.*, 2005) through handbooks assisting in the construction and design of decision support systems (Berner, 1988; Keller and Trendelenburg, 1989; López de Mántaras, 1991; Larsson *et al.*, 1997)

The literature on medical decision support devotes more time than the management literature to a discussion of the reliability and desired confidence in the decision. Most of the decisions made by clinicians are binary (two-outcome) tests for the presence or absence of a particular condition. The discussions regarding these tests are couched in the terminology of Receiver Operating Characteristic (ROC) curves‡ rather than the probability-of-failure measure common in other approaches.

1.3 Decision Support Tools

Any tool meant to aid a decision maker adds information to the decision process. Care must be taken to decrease the cognitive load while increasing understanding. The human user must remain the final decision maker, integrating the information supplied from several channels.

The decision making user:

- retains responsibility for any decision — in the medical community (among others) there are ethical, legal and trust issues at stake;
- always has access to a higher level view of the data, frequently including data from other sources;
- is an expert in their field and wants to correlate their knowledge and hypotheses with analytical results from this and other systems.

Any system that attempts to "replace" the decision maker or to "take over" any part of the analytical process will most likely be met with distrust and will not be used. From an ethical standpoint, such a system is inadmissible in any decision making arena.

The objective of a DSS tool is to augment the capabilities of the decision maker by providing an automated means of integrating facts and correlating measurements which are otherwise difficult or time consuming to asses. The results of this automated process are available as a condensed logical suggestion which can be incorporated into a larger context, along with any other sources of information available.

‡An overview of the construction and utility of ROC curves is provided in Section 2.3.1 in Chapter 2.

If the results of a decision system are to be combined in a larger scope, the decision support tool *cannot* be a "black box" as described in Wiener (1948, 1961, pp. *xi*).

A useful decision support tool must therefore exhibit all of the following attributes:

transparency: if a decision maker can not determine on what grounds an automated characterization is being suggested, they will rightly not trust the conclusions of such a system. It is critical that at all levels the decisions produced by any DSS may be easily accessed and exposed to analysis.

speed: the suggested characterization must be produced on a reasonable time-scale. If, for example in the medical domain, the system is too slow, then the results will be irrelevant, as the examination will be over and the patient will have left, preventing any iterative analysis. Correlation of the suggested results with other data must happen during the decision process.

graceful degradation: if the system must fail, this must occur with grace, indicating an increasing possibility of decision failure as a greater chance of error is encountered.

conservatism: as the degree of aggressiveness or conservatism of a classifier relates to the balancing of different types of error (false negative versus false positive), the means of choosing this balance lies in the mechanism used to integrate the automated suggestion with other data. The system must therefore support analysis of the decision confidence, providing a means to separate likely errors from quality decisions.

simplicity of use: all DSS tools function by enriching the decision environment. It is therefore easy to overwhelm the decision maker, providing so much information that the decision process is made more difficult. This effect is well described by Shortliffe and Perreault (1990) and by Kononenko (2001).

Notably missing from the above list is "optimal performance as a classifier," as all of the above factors must be taken into account in DSS design. While the frequency at which a DSS suggests the correct course of action is important, the transparency of the system outweighs the need for an optimal classifier. Any classifier with suboptimal performance, but "white box" transparency will be superior to a optimal "black box" classifier, assuming that a quality measure is also provided to give an estimate of the "white box" system reliability.

What is desired is a simple, transparent system which will allow the decision maker to easily see the decision confidence, while providing a means to "drill down" through the decision process to inspect each phase of the decision construction.

1.4 Hybrid Pattern Discovery/Fuzzy Inference System

For this work, a hybrid approach has been selected, using a combination of methods derived from fuzzy inference systems, and from the "Pattern Discovery" (PD) algorithm.

The work described here uses a rule framework generated using the PD algorithm, adapting this framework to function within the context of fuzzy inference. The results produced by fuzzy inference are then presented as a means of supporting each of several possible decision outcomes in the context of their contextual data.

The PD algorithm described in Wang (1997); Wong and Wang (1997, 2003) and in Wang and Wong (2003) is an inherently probabilistic, unsupervised learning algorithm for discrete-valued data capable of discovering polythetic patterns without exhaustive search.

These patterns are discovered through analysis of labelled training data using a contingency table to isolate true patterns from background noise, which can then be used to fill in any missing values in new data samples; treating a single column as a "label" column allows the PD algorithm to function as a supervised learning classifier. A description of this algorithm is provided in Chapter 3.

The PD algorithm was developed to deal with discrete (ordinal and nominal) data; a preliminary investigation in this work evaluates the performance of the PD algorithm to allow it to function in a continuous data domain through data quantization. Such quantization always comes with a cost, as the fine-grained detail that may be present in the underlying process is masked by the application of relatively coarse quantization intervals.

The extension and recasting of the quantized data into an FIS will allow some of the cost of quantization to be reduced, as the bin boundaries can be softened and the artificial nature of the crisp quantization bound can be diminished. The construction of the FIS is provided in Chapter 4 in which adaptations are made to the PD rules in order to improve their classification performance.

The resulting PD/FIS system is then evaluated in a decision support context after a discussion of the production of a confidence measure which estimates the reliability of each suggestion produced.

1.4.1 Outline of Decision Support and its Evaluation

Figure 1.1 indicates the data flow in the system. Training data records are presented to the PD algorithm. Maximum marginal entropy (MME), as described in Gokhale (1999); Chau (2001), is used as a discretization mechanism, producing a set of crisp events and allowing the basic PD algorithm to function in a continuous data domain.

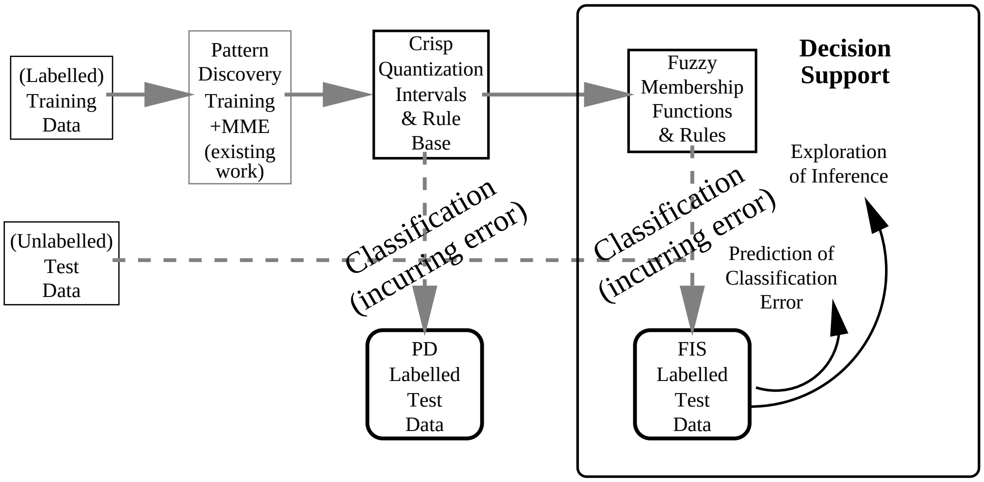


Figure 1.1: Data Flow Through Hybrid System

The events based on these crisp quantization intervals are explored by the PD algorithm, generating a set of "patterns" (or rules). The combination of a fuzzified version of the quantization intervals, rules produced through PD plus a new weighting scheme (that improves system performance) are together used as the basis of an FIS. This is described in Chapter 4.

The performance of the resulting FIS and PD systems are compared using several synthetic and real examples.

A number of synthetic data type distributions used for comparison are described in Chapter 5. The analysis of the performance of the systems on these distributions is presented in Chapters 6 and 7 for the continuous-valued PD algorithm and the fuzzy system respectively.

Analysis of the system on real-world data is performed using clinical thyroid and heart disease data found at the UCI (University of California, Irvine) Machine Learning Repository (Newman, Hettich *et al.*, 1998).[§] A description of these data sets and the analysis and discussion of the hybrid classifier performance is provided in Chapter 8.

[§] *i.e.*, `http://www.ics.uci.edu/~mlearn/MLRepository.html`

1.5 Confidence Estimation and System Reliability

The FIS based classifier provides degrees of support for multiple output classes. Using these degrees of support, a confidence measure is created that predicts the probability of classifying an input record correctly, based on a certainty-type measure relating the internal decision consistency or conflict with the probability of having produced an erroneous suggestion. This confidence scheme thereby provides an indication of the probability of failure, and can therefore be seen as a predictor of system reliability.

A discussion of the calculation and use of reliability measures is presented in Chapter 9, along with a discussion of classifier and inference reliability and confidence analysis as implemented in the FIS.

Finally, the use of the PD/FIS as a decision support system is analyzed in Chapter 10, and the overall conclusions and recommendations for future work are found in Chapter 11.

1.6 Summary

The purpose of this work is to introduce and describe a means to extract knowledge from training data for use in decision support. The primary motivation behind the data analysis and knowledge discovery process is to use the resulting knowledge base in the context of supporting complex, high risk decisions. The resulting tool must therefore be a decision support system of the highest quality.

A high quality decision support system must transparently perform two actions:

1. provide a correct labelling (*i.e.*, classify input data) with at least reasonable frequency and
2. report a confidence in the suggested labelling that truly measures the probability of an inaccurate suggestion.

This implies that for decision support, a good classifier is required as a basis; this need not, however, be the “best” possible classifier if the “best” classifier is a “black box”. A suboptimal, “white box” classifier exhibiting a good confidence measure is significantly more useful than an optimal “black box”. Such a “white box” system is more trustworthy, and thereby more reliable than an “optimal” classifier whose function cannot be understood in the context of a human user making a larger decision.

Part I

Preliminaries

Chapter 2

Pattern Recognition and Decision Support

Discovery is to see what everybody else has seen, and think what nobody else has thought.

— Albert von Szent-Gyorgi

The work described here involves the development of a decision support system based on a classifier. For this reason, it is important to create a context of surrounding work in both decision support and classification to which this system relates.

The most important relationship is the background and description of the PD system itself. Due to the importance of this topic, an in-depth discussion will be left to Chapter 3, which will provide the algorithm and describe its use and relevant theory. To provide a background for the discussion of other methods, however, this chapter will introduce the general form of the PD system prior to the comparative discussion.

Once this background is provided, the reader is given an overview of various types of classification systems which can be used for decision support. The relative merits of each system are discussed in relation to a PD based design. This provides a context for the motivation for using the PD based FIS described in this work.

The last section of this chapter will outline the past use of reliability metrics within decision support, and discuss how decision reliability is managed and measured in other systems.

2.1 Pattern Discovery Overview

In general, the PD system introduced by Wang (1997) was designed to locate and describe statistically significant patterns observed in discrete (integer, ordinal or nominal) training data.

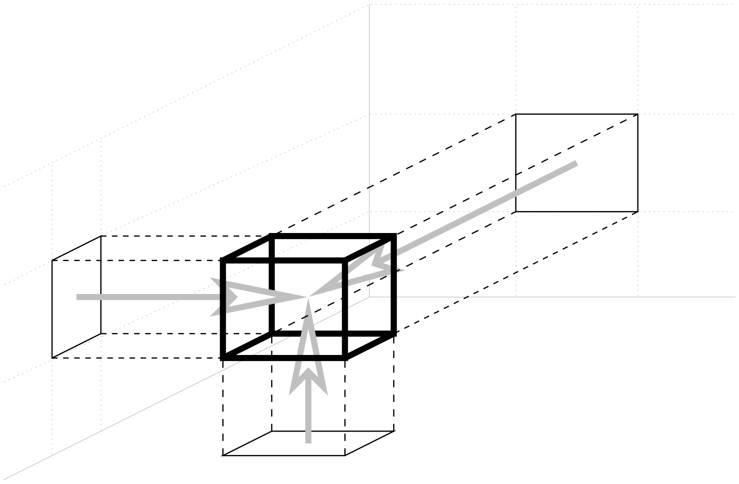


Figure 2.1: A Three-Dimensional Hyper-Cell

In order to function in a continuous domain, this work presents the PD pattern extraction algorithm with discretized data constructed via independent analysis of the density of the feature values along each dimension using marginal maximum entropy (MME) partitioning (Gokhale, 1999; Chau, 2001).

The PD algorithm then constructs a contingency table from these discretized training values to create an event-based (hyper-cell) partitioning of the input space, as shown in Figure 2.1. In this figure, a three-dimensional hyper-cell is shown as the intersection of the space defined by three cells on their respective axes. The hyper-cell is the discretely bounded space associated with unique quanta along (in this case) the axes x, y and z. A four dimensional hyper-cell would simply include a similar value along the axis w. Hyper-cells, which form first-order events, are related to each other by rules extracted through mutual occurrence of observed values. Potential rules must pass a test of statistical rigour in order to be considered, preventing patterns describing correlations due to random noise from being considered as rules.

Classification decisions are made using a nonlinear-weighted, information-theory based estimation of the relative likelihoods of each possible labelling. These estimates are calculated by using the set of rules triggered by matching input values.

Relative degrees of likelihood are calculated for each label, relating each possible choice within a spectrum running from total support through complete refutation, incorporating both positive and negative logic rules to describe the relationships present in the data. The existence of negative logic rules allows the

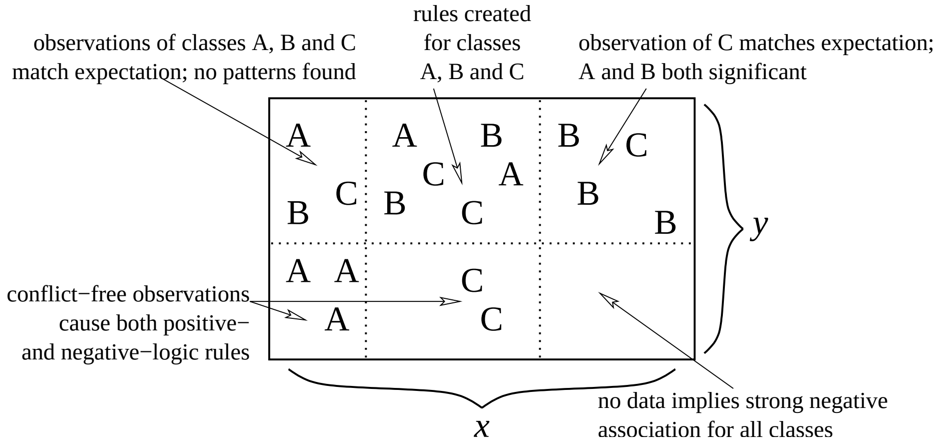

rules created
for classes
A, B and C
observations of classes A, B and C
match expectation; no patterns found
observation of C matches expectation;
A and B both significant
A
B
C
A
B
B
C
A
C
B
C
B
B
y
A A
A
C
C
conflict−free observations
cause both positive−
and negative−logic rules
x
no data implies strong negative
association for all classes

Figure 2.2: Pattern Discovery Rule Extraction Based on Uniform Random Null Hypothesis

characterizations of known negative relationships, as well as providing context to the degrees of positive relationship.

While the theory behind the pattern extraction algorithm rests on analysis of entropy, the rationale behind each decision can be discovered by comparing the observed probabilities of matching data occurring in each hyper-cell.

An example is shown in Figure 2.2 where a 3×2 grid is populated with data from three classes, “A”, “B” and “C”. There are 6 instances of each of the classes. The null hypothesis therefore states that the number of occurrences should be close to 1 in each cell (ignoring the effects of variance in this very tiny example). Each of the cells in this example have been constructed to demonstrate a particular facet of the PD hypothesis testing scheme.

Beginning in the top-left corner, we see a cell for which the observation (1 occurrence of each class) matches the hypothesis. For this cell, no patterns will be recorded as the model perfectly predicts this data.

The top-centre cell holds twice the number of predicted occurrences of each class. For this cell, patterns will be produced for all classes; each pattern will indicate that there is a significant association of its label with this class. As each pattern predicts the same number of occurrences, all the weights will be identical. These patterns will therefore record knowledge about the data without being useful for classification purposes, as there is no information present useful for this purpose. Note that this is quite different from the first cell, where there was no information present at all; the fact that this cell forms a data-dense region may be of interest to a user even though there is no decision-specific information.

Moving to the top-right cell, we see a single occurrence of class “C”, again the value predicted by the model. There will therefore be no pattern referencing class “C” constructed for this cell. Classes “A” and “B”, however, both differ from the model; class “A” by a negative deviation, and class “B” by a quite strong positive one. Class “A” will therefore have a negative rule indicating that it will *not* likely appear here, class “B” will have a strongly weighted positive rule indicating it is a likely occurrence.

Continuing across the bottom row of Figure 2.2, the two left-most cells will have positive logic rules for classes “A” and “C” respectively. In these cells there will be negative logic rules for each class which is not observed.

The bottom-right cell contains no data, indicating that classes “A”, “B” and “C” will all have a negative association with this cell. Again, this is not useful for classification purposes, as based on this training data we have no knowledge of a most-likely labelling for this cell. What is present is the strong knowledge that data in this cell is rare, this knowledge is valuable to a user as any data which may appear in this cell after training is of the utmost interested, even if the PD algorithm cannot suggest a possible labelling. If this strong knowledge is contrasted with the weak knowledge found in the top-left cell, it is apparent that direct knowledge of event rarity is quite different from the knowledge of random occurrence, as a random occurrence simply indicates that there is no information present.

The presence of both positive and negative logic rules, and the source of the rules in calculation of mutual occurrence provides a transparency of inference extending down through the rule base to the underlying data distribution. This level of transparency makes the use of PD highly advantageous as the foundation for a DSS, along with the benefits of the statistical basis of both its rules and the MME input quantization.

Simply put, in a domain such as decision support where the importance of transparency is paramount, the accessibility of simple, robust, statistical arguments makes a PD based system very attractive.

2.2 Pattern Recognition and Classification Techniques

“Pattern recognition” is the study of techniques for the extraction and matching of a known pattern in a test data set. This is a subset of the larger field of “machine learning.” To discuss the features of the PD based FIS system described in this work, a description of the data analysis issues will be presented, followed by a brief description of various classification algorithms which are popularly used.

2.2.1 Considerations Derived from the Data Domain

The domain of the input data, and the universe of possible values within it, is a defining characteristic for classification algorithms. Some classifiers function best on continuous data, some on discrete. This is largely due to the structure of the data representation within the classifier as it relates to the structure of the data itself. The largest portion of real-world data is, however, continuous. In order to use discrete methods on continuous data, some form of quantization must be applied.

Classification of Continuous-Valued Data

Classifiers designed for use with floating-point values generally view the data domain as an n-dimensional space in which one or more surfaces will be placed, to form boundaries between decision regions. Each region is unambiguously associated with one of K labels associated with the data set.

If these surfaces happen to form an orthogonal planar division of the input feature space it is quite easy to explain the mapping between feature values and output labels. This scenario, which is very unlikely when locating an optimal input space division, is a common characteristic of quantized/discrete analysis schemes and is largely what makes them so easily explained.

Note that even when a classifier is described as functioning on "continuous" data, the domain of the input data for a supervised classifier is still not $x \in \mathbb{R}$. There are two major reasons for this: the underlying representational limits of a digital "floating point" representation; and the fact that in any finite amount of training data, the infinitely large universe of real numbers (*i.e.*, $\mathbb{R}$) cannot be realized. Instead, a relatively small number of distinct values will be observed, though the universe of values which are possible in a training set is quite large (and bounded only by the digital representation).

Classification of Discrete Data

"Discrete data" by contrast can be defined as a data universe in which the set of *possible* values is both finite and, usually, small. The largest universe of such data will be $x \in \mathbb{I}$; smaller nominative or ordinal sets are also described as discrete data.

Classifiers of discrete data tend to be relatively simple as there is a known finite universe of possible cases into which each data element can fit.

It is therefore possible to come up with an exhaustive description (at least for relatively small universes) which can eliminate the need for generalization to large sections of the data universe. A complete description of the universe can be created using a large contingency table outlining all possible choices —

that is, the data can be divided among a set of orthogonal hyper-cells, where cells are created independently along each axis, and thereby group input data values.

As the universe of possible combinations grows, the main problem in a discrete classification system is that it may be impossible to ascertain a probable labelling for some cells in the contingency table, as some data value combinations may never be observed during training.

One of the major problems with the application of discrete algorithms is the fact that a large fraction of real-world data is continuous. In order to use a discrete algorithm on continuous data, the data must be quantized or "discretized". Adapting quantized data to discrete algorithms is therefore an important topic.

The case of dividing input data into discrete quanta can be seen as similar to the division of the input space as managed under continuous data. A quantization algorithm tends to construct the (orthogonal) input divisions described above as unlikely when using a continuous algorithm. The difference between the orthogonal division of the discrete quantization and the "least error" division of the continuous algorithms is a significant source of error when using continuous data in a discrete algorithm. The benefit is that feature independent orthogonal quantization is transparent, and easy to explain. Such quantization produces an event-based data space in which discrete algorithms can be used.

As will be shown in the discussion on the performance of the fuzzy inference system (FIS), some of the elements in the discretized data can be accommodated by using rules speaking to data with similar locality when discrete analysis is performed on quantized continuous data. This alleviates some of the cost incurred by using quantization.

2.2.2 Learning System Architectures

A summary will now be presented of several popular classification architectures. In each case, the strengths and weaknesses relative to the proposed PD/FIS system will be discussed, in terms of the intended application area of decision support systems.

Back-propagation Artificial Neural Networks

A back-propagation network (Rumelhart, Hinton and Williams, 1986; Minsky and Papert, 1988) trains by using a random initialization of weights describing a set of partitions; an error surface is iteratively minimized by successively considering the error relative to each input point many times.

This is a gradient-descent method, so therefore back-propagation networks are prone to issues with local minima in the error space. The main obstacle to their application in this domain is the resulting lack of interpretability of the final stable state.

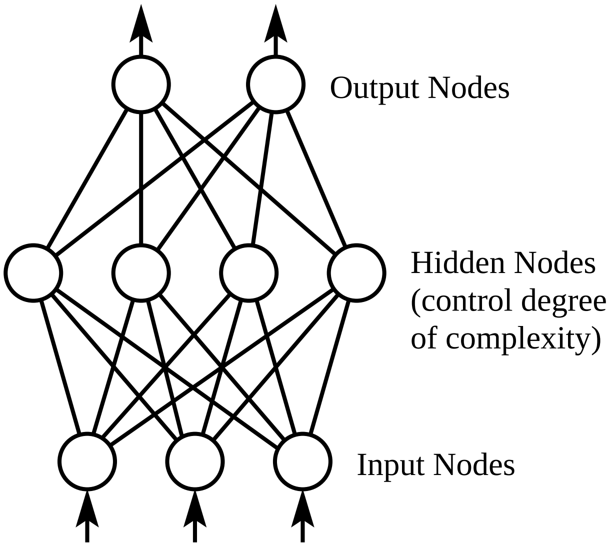


Figure 2.3: A Back-Propagation Neural Network

This type of network is constructed by a set of input, output and hidden nodes, as shown in Figure 2.3 (adapted from similar figures found in Rumelhart *et al.* (1986); Minsky and Papert (1988); Simpson (1991); Hertz, Krogh and Palmer (1991) and others). Each node is fully connected to each subsequent node; the value in each node is therefore passed on to each subsequent node after being scaled by a weight value associated with the link (*i.e.*, the lines in Figure 2.3).

The back-propagation algorithm describes how to tune these weights to reduce the observed error on training data. The weights are usually seeded with randomized values.

Each node in the hidden layer (or layers) of a back-propagation network allows a greater degree of non-linearity in the final partition space by controlling the location and angle of some high-dimensional hyper-plane. While the geometry of these planes is accessible through an examination of the weights, the actual topology of the space is not easily visualizable, and certainly is not explainable to a user not familiar with the mathematics involved.

A further complication is that the gradient descent search from a randomly initialized topology is not likely to produce a division of input features which is logical in anything other than an abstract mathematical sense. In particular, the divisions of the input space will appear arbitrary to a casual user, again requiring a mathematical explanation to assure the user of their correctness and logic.

For this reason, a simpler division of the input feature space and a more intuitive description of the resulting decisions surfaces will produce a more accessible and transparent classification engine, resulting in a more understandable (and therefore better) decision support system.

Such a network is sometimes referred to as a multi-layer perceptron (MLP).

Expectation-Maximization (E-M)

The E-M algorithm (Duda, Hart and Stork, 2001) operates on a maximum-likelihood probabilistic (Bayesian) approach. The main feature of the E-M algorithm is that its design takes into account missing feature values, replacing their values by a maximum-likelihood estimation when required during training.

While this admirable feature makes it robust and useful on many real-world continuous and discrete data sets when missing values are present, it suffers from many of the same drawbacks as back-propagation in terms of its transparency and interpretability.

E-M is not a gradient descent algorithm; instead the E-M algorithm maximizes the log-likelihood of all observed data values, filling in expected values for any missing data points. The global log-likelihood expectation is maximized by an iterative search based on a simple assumed model $\boldsymbol{\Theta}$ which is gradually fit to match the observed likelihoods of the available data. This algorithm is explained in detail in Duda *et al.* (2001, pp. 124–128).

While not a gradient descent algorithm, the resulting fit to a parametric error surface will be just as opaque as the local minima fit optimizations of back-propagation. Essentially, the E-M algorithm is still an iterative fit to an error surface; the means by which the fit is produced is not instructive in determining the value of any final rule. In particular, both the input space divisions and resulting classification geometry will again require significant mathematical analysis to convey the relationship between the final rules and the input data to a user.

Support Vector Machines (SVM) and Maximum Margin Classifiers

Support Vector Machines (SVM) (Vapnik, 1995; Joachims, 1998, 2005; Cristianini and Shawe-Taylor, 2000; Duda *et al.*, 2001) and the larger family of "kernel" based classifiers such as maximum margin classification (Xu, Neufeld *et al.*, 2005) actually increase the dimensionality of the input space (potentially to an infinite number of dimensions) by projection.

The advantage of doing this is to move the data into a high dimensional space in which the data is well separated, and classification with low error rates is possible.

Excellent performance results may be had by this technique, however from an interpretability point of view this idea obscures the inner workings of the algorithm even more than is the case in back-propagation as instead of working in a space with analogous dimensionality to the input space, a user must now understand the projection by which the kernel methods expand the number of degrees of freedom for the problem.

Bayesian Decision Theory

The ideas behind Bayesian Decision Theory (Bayes, 1763; Duda *et al.*, 2001) involve the probabilistic minimization of computed risk.

The main limitation in Bayesian decision theory is the assumption that the distribution from which input data points is drawn is well understood and can be characterized in terms of its distribution and overall likelihood.

The PD algorithm used in this work does not make any such assumptions, and as will be shown in later chapters is largely insensitive to the actual distribution. This feature makes PD an interesting candidate for DSS design.

Dempster-Shafer Theory

The decision technique referred to as Dempster-Shafer (D-S) Theory (Dempster, 1968; Shafer and Pearl, 1976; Shafer, 1990; Shafer and Pearl, 1990; Yager, Fedrizzi and Kacprzyk, 1994) is the combination of strict maximum probability based assignment with a belief model. The addition of belief to probability theory provides a mechanism for the representation of the differing amounts of knowledge available in various situations, and in particular, allows a representation of conflict and uncertainty within the mechanism of probabilistic inference.

As an example, of the function of D-S theory, let us consider a case where two witnesses, Laura and Monica report on whether a burglary has just occurred at a store. At the time of the incident, Laura was across the street from the store, waiting for a friend. Monica, on the other hand was sitting on a bench reading a book, and therefore not paying as much attention to her surroundings. Let us therefore represent our degree of belief in statements from the two witnesses as $B_L = 0.9$ and $B_M = 0.6$, indicating that we have a high degree of belief in Laura's testimony, and a lesser degree in Monica's.

If Laura states that a burglary took place at the store, and if Monica disagrees, then we can represent our understanding using D-S theory. First, based on only Laura's statements, we would have a 0.9 degree of belief that a robbery took place, but a 0 degree of belief that one did not. This is due to the fact that discounting Laura's story does not contradict the possibility that a robbery occurred unobserved. Similarly, Monica's story gives us a 0.6 degree of belief that a robbery did *not* occur, but a zero degree that one did.

Treating the witnesses as independent, we can simply multiply values to calculate probabilities, as described in Dempster (1968). Therefore we can calculate the probability that Laura is reliable but Monica is not

$$R_L = B_L \times (1 - B_M) = 0.9 \times 0.4 = 0.36 \tag{2.1}$$

or that Laura is not reliable but Monica is

$$R_M = (1 - B_L) \times B_M = 0.1 \times 0.6 = 0.06; \tag{2.2}$$

we can also calculate the probability that neither is reliable

$$R_\emptyset = (1 - B_L) \times (1 - B_M) = 0.1 \times 0.4 = 0.04. \tag{2.3}$$

We must normalize each of these possibilities over the universe of total possibility. Note that in this case, the universe does not sum to 1.0 because Laura and Monica have taken opposing views, and in consequence there is no possibility that they are both correct. The size of the universe of possibility in this case may therefore be expressed as the sum of the three probabilities just discussed, or

$$\mathcal{U} = R_L + R_M + R_\emptyset = 0.36 + 0.06 + 0.1 = 0.52. \tag{2.4}$$

Taking each of these possibilities in turn over the universe of possibility, we get:

$$\begin{aligned}
\Pr(R_L) &= \frac{R_L}{\mathcal{U}} = \frac{0.36}{0.52} = 0.69 \\
\Pr(R_M) &= \frac{R_M}{\mathcal{U}} = \frac{0.06}{0.52} = 0.12 \\
\Pr(R_\emptyset) &= \frac{R_\emptyset}{\mathcal{U}} = \frac{0.1}{0.52} = 0.19
\end{aligned} \tag{2.5}$$

This allows us to represent the probability that Laura is correct (and there was a robbery) at 0.69; the probability that Monica is correct (and there was no robbery) at 0.12; and the additional probability of not knowing whether a robbery occurred or not as 0.19.

As explained in Zadeh's (1984) review of Shafer and Pearl (1976), the result of D-S modelling is an inference system which is based on probabilities involving sets of elements, rather than on a point probability model. Even though in his review Zadeh states that (in his opinion) "D-S theory does not capture the human mode of reasoning about possibility," this model of representation remains interesting,

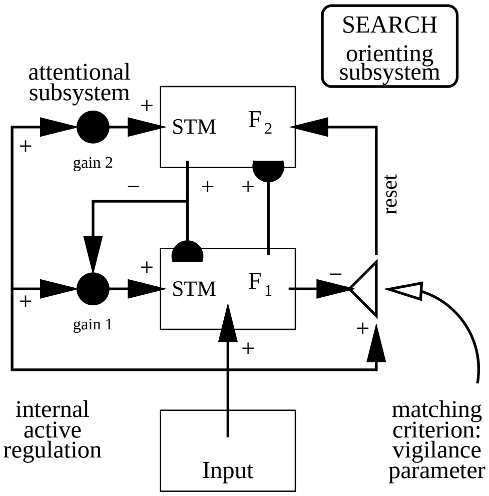


Figure 2.4: Typical ART Neural Network

as attested by the ongoing and vigorous publication activity in this field. A model based on D-S theory to provide decision support in some manner designed to fit the purpose evaluated in this work would certainly be possible, and a future work could explore this option.

Adaptive Resonance Theory (ART)

The Adaptive Resonance Theory (or ART) algorithm of Grossberg (1976) is a neural net based approach which was originally phrased as a model for true biological learning. While based on the ideas in Grossberg (1976), the technique was fully presented in Carpenter and Grossberg (1987b) and has since been refined (see Carpenter and Grossberg, 1987b,a, 1990; Carpenter, Grossberg and Reynolds, 1991a; Grossberg, 1995).

This network stores exemplar values observed in a training data set and initializes a new exemplar when no existing match can be found within a given tolerance. When no new exemplar is created, the nearest match existing exemplar is updated to take into account resemblance to the newly viewed input vector.

Figure 2.4 shows the general scheme of an ART-based neural network. This figure is reproduced from Carpenter and Grossberg (1990), as is Figure 2.5, which shows the sequence of events that occur in ART pattern matching. In Figure 2.5 a), a new pattern is shown to the system, which does not exactly match the previously stored exemplar. In Figure 2.5 part b), this is discovered by the "attentional subsys-

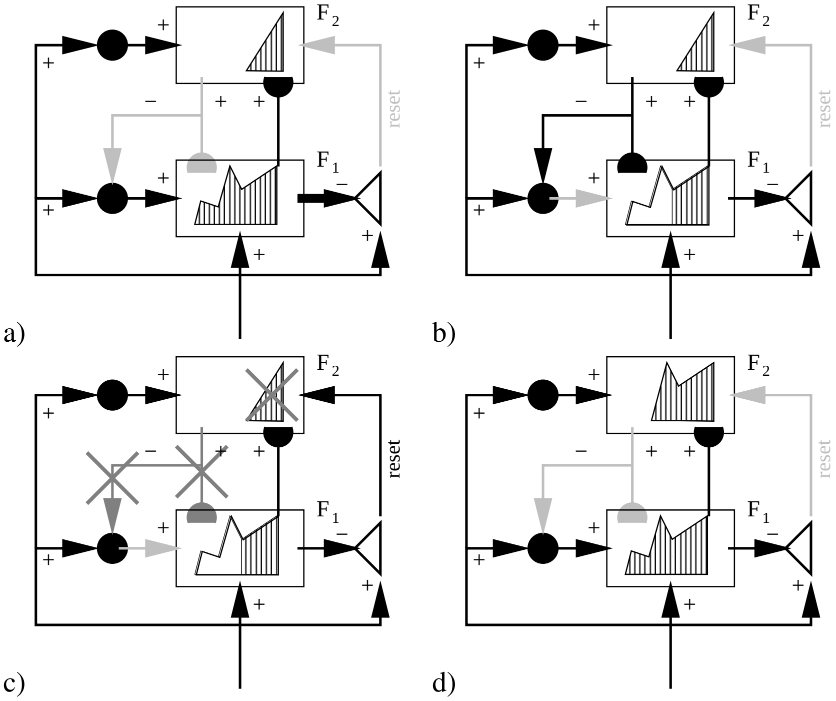


Figure 2.5: Exemplar matching in ART

tem" forming the left-hand side of the network. Figure 2.5 c) shows that the patterns are insufficiently different (determined by a threshold control termed the "vigilance parameter"), and so in Figure 2.5 d), the exemplar is updated to incorporate the new data.

This striking idea has wonderful visual connotations, and is potentially very expressive in pattern matching, assuming the user can transparently understand the metric by which a match occurs.

As originally proposed in Grossberg (1976) and as refined through Carpenter and Grossberg (1987a,b, 1990) ART deals strictly with discrete data values. By adapting the feedback state through use of fuzzy logic techniques, "Fuzzy ART" (Carpenter, Grossberg and Rosen, 1991b) seamlessly deals with continuous-valued data, and provides one of the most elegant refinements of a discrete algorithm through the use of fuzzy systems.

While ART is interesting, in the context of our decision support goals it is not clear that the matching of the exemplars in a generic feature space would be explainable, other than in a manner similar to that used by back-propagation.

Evolutionary Algorithms

Evolutionary algorithms, as a family, take the idea of gradient descent search into a probabilistically-directed search domain.

Within the general family of evolutionary algorithms are such ideas as genetic algorithms and genetic programming (Goldberg, 1989; Goldberg and Deb, 1991; Syswerda, 1991; Koza, 1992; Mitchell, 1996) as well as simulated annealing (Kirkpatrick, Gelatt *et al.*, 1983), and other randomized search techniques such as particle swarm (Kennedy and Eberhart, 1995).

All of these techniques revolve around the central idea that local minima can be avoided by adding a noise element to the search direction. Coupled with a parallel search of the problem space involving many (randomly initialized) points, this powerful technique is tractable even when applied to several difficult problems for which other techniques fail. In particular, the problem space need only be characterized in terms of a cost or benefit function; geometries which are poorly understood can thus be traversed as long as two possible solutions can be evaluated in terms of their merit, even if they cannot be evaluated in any other way.

While quite powerful and requiring extremely small amounts of configuration, the results returned by an evolutionary solution have no guarantee of interpretability other than as a least-cost/best-fit solution.

While the solution itself may perform well with respect to classification, there is no available explanation of the reasons underlying the selection of a given label.

Decision Tree Classifiers

A decision tree classifier functions by establishing a set of decisions that, when made successively, result in the assignment of a label to input data.

Decisions and sub-decisions are always made based on the same initial test, so the decisions themselves can be structured in the form of a tree.

Decision trees form a powerful, simple means of rapidly coming to a classification based on applying a sequence of decisions over an input data vector in order to traverse the tree.

The broad field of decision tree classifiers contains algorithms such as Ross Quinlan's ID3 (Quinlan, 1986), C4.5 (Catlett, 1991; Quinlan, 1993) and FOIL (Quinlan, 1996) algorithms, as well as other popular systems such as WEKA (Witten and Frank, 2000).

The main drawback of a decision tree is the forced direction of traversal. This does not capture how humans actually think, and may not be well supported by the data if there are missing values present.

In order to support human decision making, it is preferable to support the human mode of thought; this

Table 2.1: Mushroom Example Data Set

Size	*Spots*	*Colour*	*Class*
Large	Spotted	Yellow	Poisonous
Large	Striped	Brown	Edible
Small	Spotted	Brown	Edible
Small	Spotted	White	Poisonous
Small	Plain	Brown	Poisonous

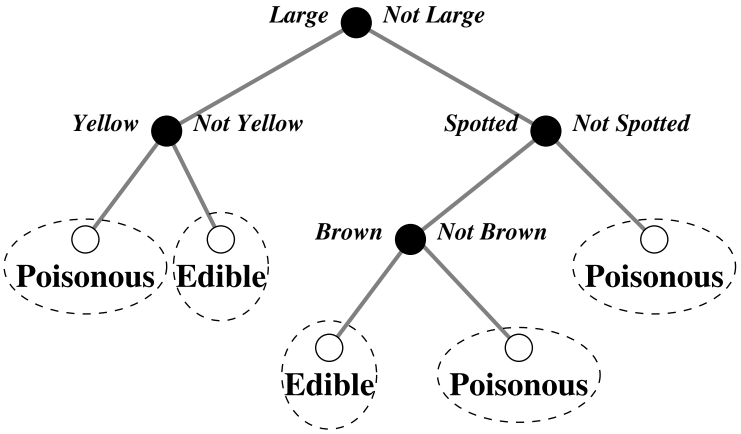


Figure 2.6: Tree Characterizing the Mushroom Data Set

especially includes exploring the results of "what if?" questions, which may involve constructing answers based on partial knowledge which is not encoded in the tree.

In a decision tree, a single test must be isolated as the root of the tree. The PD algorithm allows the presentation of the highest-weighted rule triggered by the match of the data vector as the strongest (and therefore first) source of decision support. In a tree system there is always a fixed first question which one must ask.

Similarly, it is impossible, while navigating a tree, to ask a question based on an unobserved value – the best response which can be made is that the creation of the tree did not necessitate observation of that value, based on traversal from the initial root node. This is completely different both from deciding the value is irrelevant to any decision, and as is the case in PD, from deciding the value cannot answer any questions with statistical confidence.

Consider the example data set shown in Table 2.1, and the decision tree shown in Figure 2.6. While this tree adequately captures the decisions needed to *classify* a new input record, it cannot answer the question:

> "*Are mushrooms which are* LARGE *and* STRIPED *edible?*"

A system presenting a full contingency table would allow exploration of such a question. The partial contingency table of PD supports all questions for which a statistically significant answer is available. To do this, significantly more rules are stored by PD than are stored by C4.5.

The structure of the tree yields an efficient means of classifying data (as the maximum number of data elements examined is equal to the height of the tree).

The efficient classification of a tree algorithm is significantly different from the problem of explanation. In such an algorithm, the "explanation" consists of the presentation of the single rule defining the traversal from root to leaf. This path defines the series of tests by which the tree determined the outcome, ranked from the decision with the highest information gain to achieve any labelling down to the decision which resolved the given label within a small sub-partition of the data space. The decision forming the first branching of the tree therefore only describes what decision globally gives the most information within the decision space, whereas in the PD system the user is presented with the test that contains the most information *specific to the current input data elements*.

For this reason alone, a PD based implementation merits consideration over a tree-based one, as data-specific analysis reflects the way in which decision makers will approach their data. The presence of both positive- and negative-rule inference in PD allows relationships between labels and rules to be explored which are simply not available in a tree based system.

In some other work, such as in Kim, Lee and Min (1999); Boyen and Wehenkel (1999) and Chiang and Hsu (2002), ID3 and C4.5 have been developed into fuzzy inference systems.

Fuzzy Inference Based Classification

Fuzzy inference is a family of techniques, referring only to the means by which logical values are combined in the form of rules, over a set of input membership functions.

Fuzzy inference in general does not speak to how the rules or input membership functions are created.

The work in this thesis is a fuzzy inference system, and suggests one possible means of creating both rules and input membership functions.

Several other papers mention rule generation for fuzzy systems. This literature can be divided into:

- interview of domain experts and construction of rules through human knowledge representation (Mamdani, 1974; Zadeh, 1976, 1978, 1983; Heske and Heske, 1999);
- fuzzy clustering (Krishnapuram and Keller, 1996, 1993; Pal, Bezdek and Hathaway, 1996; Pal, Pal and Bezdek, 1997; Hathaway and Bezdek, 2002);
- neuro-fuzzy systems (Pedrycz, 1995; Labbi and Gauthier, 1997; Pal and Mitra, 1999; Kruse, Gebhardt and Klawonn, 1994; Nauck, Klawonn and Kruse, 1997; Höppner, Klawonn *et al.*, 1999; Mitra and Hayashi, 2000; Gabrys, 2004);
- fuzzy systems designed or configured through evolutionary algorithms (Ishibuchi, Nozaki *et al.*, 1994; Ishibuchi and Murata, 1997; Cordón, Herrera *et al.*, 2001a; Spiegel and Sudkamp, 2003; Hoffmann, 2004);
- use of an extension matrix (Hong and Lee, 1996; Yager and Filev, 1996; Chong, Gedon *et al.*, 2001; Wang, Wang *et al.*, 2001; Xing, Huang and Shi, 2003) or tree generation algorithm such as ID3 or C4.5 (Quinlan, 1986, 1993, 1996);
- approaches using contingency tables based on rough sets (Bean, Kambhampati and Rajasekharan, 2002; Shen and Chouchoulas, 2002; Tsumoto, 2002; Ziarko, 2002); and
- schemes using some kind of statistical clustering technique to create input membership functions, combined with contingency table generation (Chen, Tokuda *et al.*, 2001; Chen, 2002; Kukolj, 2002).

Possibilistic *c*-Means and Fuzzy *c*-Means

Possibilistic *c*-Means (Krishnapuram and Keller, 1993) and fuzzy *c*-Means (Pal *et al.*, 1997) each provide another means of generating rule sets based on examination of the underlying data. Both function by performing a gradient descent evaluation over some (*c*) clusters in the feature space.

Showing some similarity to *k*-nearest neighbour, they both create fuzzy boundaries associating “nearby” values together.

These systems function by performing the same sort of gradient descent search used by back-propagation neural networks, which means that they may be trapped in local minima, and are not guaranteed to contain rules whose construction will be transparent to an end-user.

Evolutionary Fuzzy Classifiers

Evolutionary fuzzy classifiers, such as Fukumi and Akamatsu (1999), while also creating linguistically based rules uses the same type of randomized search used by other evolutionary algorithms. While the rules themselves may be readable by a user, the means by which the rules were constructed is opaque; in

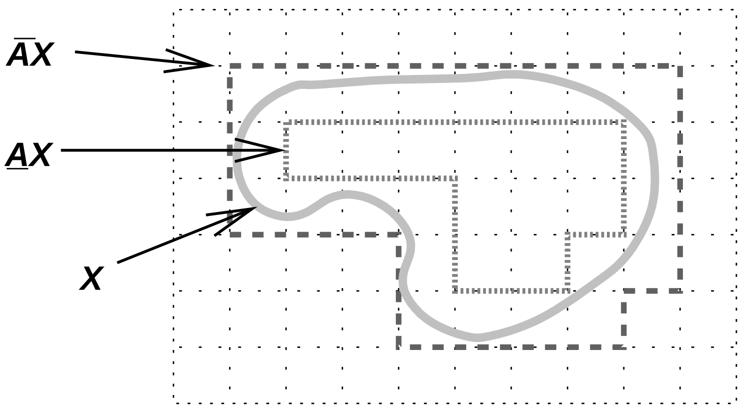


Figure 2.7: Rough Set Example

contrast, the PD system offers a clear statistical basis for rule construction.

Neuro-Fuzzy Classifiers

Similarly, neuro-fuzzy classifiers, while more transparent than their non-fuzzy neural network counterparts have the same problems with respect to the transparency of the rule creation.

These systems function by performing the same sort of gradient descent search used by back-propagation neural networks, which means that they may be trapped in local minima, and are not guaranteed to produce rules whose construction will be transparent to an end-user, but rather involve rules “that produce a good result.”

Rough Set Based Classifiers

Rough sets (Pawlak, 1982, 1992; Lin and Cercone, 1997; Polkowski and Skowron, 1998; Øhrn, 1999; Ziarko, 1999; Grzymala-Busse and Ziarko, 1999; Ziarko, 2000; Grzymala-Busse and Ziarko, 2000; Ziarko, 2001), as with fuzzy logic is not a technique by itself, but rather a means of representing uncertainty.

Rough sets deal in discernibility between equivalence classes rather than partial membership in a class. Figure 2.7 shows a rough set in relation to a set of attributes, A. This figure shows the three regions of discernment relative to A (the attribute universe) and relative to X, the true relation on A:

- the “lower bound” on the set, or $\underline{\mathsf{A}}\mathsf{X}$
- the “upper bound” on the set, or $\overline{\mathsf{A}}\mathsf{X}$

- the region outside of $\overline{\mathsf{A}}\mathsf{X}$

This is referred to as an "indiscernibility relation". The boundaries for $\underline{\mathsf{A}}\mathsf{X}$ and $\overline{\mathsf{A}}\mathsf{X}$ may be the same; if this is true for all boundaries then the set is simply a crisp set.

The task of constructing the relations is an open question, exactly comparable to the construction of input membership functions in fuzzy logic. While techniques to create a rough-set based characterization scheme could just as easily have been produced using a PD basis, there is no *a priori* reason to assume that the performance or reliability would be any better than the FIS which has been implemented.

Classification Comparison

While the PD algorithm must deal with quantization issues, the attractiveness of the robust statistical explanation for its rules, along with the statistical explanation from MME for quantization gives a PD based system such a high degree of transparency that it is a very interesting algorithm in the context of decision support.

2.3 Reliability

There are several means of discussing the reliability of a test. The two most common means of doing so are: receiver operator characteristic (ROC) curve analysis and measurement of decision confidence using probabilistic methods.

2.3.1 Overall Classifier Reliability — ROC Analysis

Consider the situation where there are two candidate classifiers for a given two-class problem. A method to evaluate the two classifiers and provide a means of ranking their quality at discerning between the classes would be advantageous in selecting which classifier to use.

For example, if a test is constructed for hypothyroidism, an analysis can be done to determine that this test is likely to produce an incorrect result 1.25% of the time, and that of these errors, 75% will incorrectly indicate the presence of a disease when there is none.

This type of analysis is done using a receiver operating characteristic (ROC) curve (Metz, 1978) analysis, which provides sensitivity and specificity measures for a given two-outcome classifier.

This type of measure is discussed extensively in the medical informatics literature (see any of: Berner, 1988; Keller and Trendelenburg, 1989; Shortliffe and Perreault, 1990; Kononenko and Bratko, 1991;

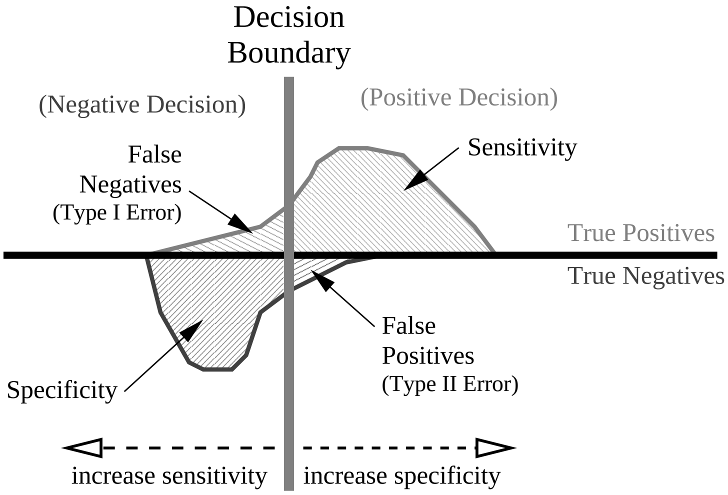


Figure 2.8: Specificity and Sensitivity

López de Mántaras, 1991; Gibb *et al.*, 1994; Harris and Boyd, 1995; de Graaf *et al.*, 1997; Larsson *et al.*, 1997; Friedland, 1998; Kukar *et al.*, 1999; Kononenko, 2001; O' Carroll *et al.*, 2002; Abu-Hanna and de Keizer, 2003; Colombet *et al.*, 2003; Coiera, 2003; Cowan, 2003; Kukar, 2003; Montani *et al.*, 2003; Brillman *et al.*, 2005) as well as in much of the discussion regarding reliability in decision making (see, for example Schniederjans, 1987; Sage, 1991; Sundararajan, 1991; Barlow, Clarotti and Spizzichino, 1993; Cordella *et al.*, 1999; Gurov, 2004, 2005).

The general purpose of an ROC test is to generate a statistic globally characterizing the ability of a two-outcome classifier to separate two distinct outcomes.

Figure 2.8 shows a plot describing typical distributions of desired positive and negative outcomes. The portion of the graph above the central horizontal line indicates a PDF (probability density function) of the occurrence of the values for which we want to see identification as a "positive" outcome. Below the line is a similar PDF describing the occurrences of values for which a "negative" identification is desired. The vertical line in the centre of the figure indicates a decision threshold, dividing the data values between "positive" and "negative" decisions; values from either PDF falling to the right of the decision boundary will be identified as "positive", and all values (from either PDF) falling to the left will be identified as "negative".

The specificity of the test is simply calculated as the fraction of the "positive" decisions that are correctly identified; the sensitivity, conversely is the similar fraction of the "negative" decisions.

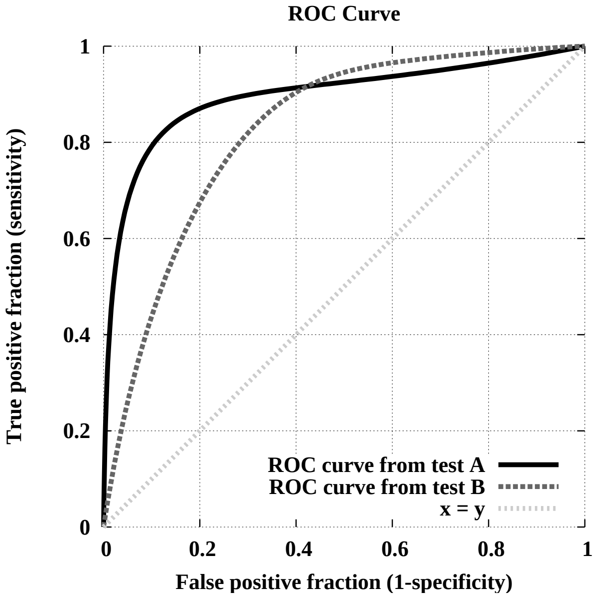


Figure 2.9: Example ROC Curve

As these distributions frequently overlap (as shown in Figure 2.8, it is usually impossible to select a test statistic where a single threshold will avoid making errors in assigning label values.

From ROC analysis, we get the categorization of error types:

Type I Error: a rejection of the null hypothesis (that being a successful test) when we should accept; also a “false negative;”

Type II error: acceptance of the null hypothesis when we should reject it; also a “false positive.”

By plotting the fraction of the errors incurred using each possible threshold value for a given classifier, a curve (the ROC curve) is produced. The area under such a curve provides a metric by which two classifiers can be compared. The curve with the greater area has the greater ability to discriminate between the desired positive and negative outcomes.

The following example is drawn largely from the work by Tape (2005). Two ROC curves are plotted as shown in Figure 2.9, for two different candidate tests “A” and “B.” The fraction of positive-class results which are classified correctly (the sensitivity) is plotted against the fraction of positive-class results which are classified incorrectly (1-specificity).

As can be seen in Figure 2.9, this will produce a curve which is above the line *x*=*y*. The better the test, the more closely this line will approach the axes of this graph. For this reason, the accuracy of the test is measured using a calculation of the area under the ROC curve on the range $[0 \dots 1]$, which produces an area bounded in the domain $[\frac{1}{2} \dots 1]$.

ROC curves which have an area close to $\frac{1}{2}$ are use useless, those whose area is 1 are perfect classifiers between the "positive" and "negative" classes.

As the shape of the distribution of decisions (as shown in Figure 2.8) is dependent upon the test, the ROC curve provides a means of evaluating different tests for the same decision.

As can be seen in Figure 2.9, this provides a simple means of characterizing the two tests, as it is not otherwise obvious which of tests "A" and "B" would be superior, as at some points the sensitivity of "A" exceeds that of "B", and at some points the reverse is true.

Given the calculations of the areas under the two curves $\text{Area}(A) = 0.74$ and $\text{Area}(B) = 0.81$, we see the "A" test is apparently superior by this measure, however it is important to note that statistical tests for significance between the curves still apply, incorporating measures such as the number of points, *etc*.

While ROC analysis can provide a measure of performance of a classifier in a two-outcome test, this measure is a global one, reporting the reliability as a mean chance of failure of the classification system. What is truly desired in a decision support system is a means of inferring the probability of failure of a particular decision, not the class of all decisions made by a system. For that reason, we turn to the discussion of decision confidence within a system when knowledge of particular input values is available.

2.3.2 Input Specific Reliability

System reliability can be measured in terms of the probability of a correct response. Specifically, the "reliability" of a system is simply the inverse of the probability of failure, or

$$\begin{aligned} C &= \mathcal{E}(\mathcal{R}) \\ \mathcal{R} &= 1 - \Pr(\text{failure}) \equiv 1 - \Pr(\text{Incorrect}) \end{aligned} \tag{2.6}$$

Generalized Probabilistic Measurement of Decision Confidence

Frequently, this is calculated as the compound probability of both measurement and inference error, combined using Bayesian logic.

Bayes's (1763) theorem provides the well-known and convenient mechanism of relating a prior distribution of a particular label $\Pr(\Psi)$ with the probability of the occurrence of a particular input vector given the occurrence of the label $\Pr(\mathbf{x}|\Psi)$ providing the probability of occurrence of the label, given the input vector, as

$$\Pr(\Psi|\mathbf{x}) = \frac{\Pr(\Psi)\ \ \Pr(\mathbf{x}|\Psi)}{\Pr(\mathbf{x})},$$

using the relation

$$\Pr(\Psi|\mathbf{x})\ \Pr(\mathbf{x}) \;=\; \Pr(\Psi, \mathbf{x}) \;=\; \Pr(\mathbf{x}|\Psi)\ \Pr(\Psi).$$

Using these relations, one can calculate the probability of an incorrect assignment for any suggested labelling based on the observed probabilities calculated on training data, assuming a direct probabilistic path of inference exists from input value measurement to output characterization. If this path exists, this allows a software system to report the probability of incorrect support at the time of a decision presentation.

By using Dempster-Shafer theory, this technique can be extended through the postulated limits of “belief”, however the underlying assumption in all Bayesian inference is that the mapping of uncertainty can be done in probabilistic terms.

Such a technique cannot be used when the mechanism of inference is not probabilistic, such as in possibilistic fuzzy inference systems. The use of a derived probabilistic model is frequently used in such contexts.

Derived Probabilistic Methods

Generalized indices of reliability which only approximate probabilistic models can also be used, such as the *certainty factor* of the MYCIN system (Shortliffe, 1976; Buchanan and Shortliffe, 1984).

The MYCIN index simply used a minimum operation at each logical join, rather than computing true joint or conditional probabilities.

While not directly based on probabilistic methods, it has been proven by Heckerman (1986) (and summarized by Ginsberg (1993)) that there is an underlying probabilistic basis for this method.

When tested against a true probabilistic model using the same inference, there was no measurable difference in MYCIN’s performance (see, for instance, Ginsberg, 1993).

Much of the discussion regarding the “usefulness” of fuzzy systems involves a variation of this discussion (for example Klir, 2005a,b), and much of this work is in the form of a response to the famous paper by Cox (1946), which claims that any non-probabilistic system is without merit.

The large volume of research continuing in the many-sided field of approximate reasoning refutes the validity of Cox’s claim, as do specific responses such as those by Klir, above.

Measured Reliability

The overall focus of all reliability modelling is to predict the true probability of failure of a specific classification or suggestion. It is therefore desirable to determine reliability by calculating the probability of failure of the labelling process as a function of some system parameter (usually input values or some internal state variable).

Such a calculation will provide an estimate of the true system reliability. The quality of the estimate can be assessed by evaluating how well estimated values correlate with measured reliability, as calculated for a population of test values.

In the work in this thesis, such a reliability metric will be used. This technique will be introduced, discussed and evaluated in Chapter 9, “Confidence and Reliability.”

2.4 Summary

Several classification schemes have been presented. The common thread of all of the systems discussed is that they do not have complete transparency in the creation of their decisions, and they do not necessarily support the exploration of a suggestion presented to the user.

What is desired is a system which has a sound basis for rule generation, provides a simple means of producing suggestions which will be explainable, provides a good estimate of the reliability of any suggestion made, and, finally, presents its suggestions in a framework that allows the exploration of both suggestion itself and the decision space from which the suggestion was drawn.

A pattern discovery based characterization system has all of these benefits:

- there is a statistical basis for both the input quanta and rules (patterns) generated from them;
- transparency is provided both through framing inference in terms of rules (or patterns) and from the statistical basis of the system and
- the adaptation of the PD algorithm to fuzzy analysis provides a linguistic mechanism for explanation and inference.

What remains is to consider the performance of a PD based FIS. To do this, we must first understand how the PD algorithm works.

Chapter 3

(Non-Fuzzy) Pattern Discovery

The Pattern Discovery (PD) algorithm was originally described in the doctoral work of Wang (1997) and later published in the journal literature (see Wong and Wang, 1997, 2003; Wang and Wong, 2003). It functions as a rule based classifier constructed through statistical inference.

3.1 Pattern Discovery Algorithm

Consider a set of discrete training data presented as an array of N rows of length M+1. Each row or input vector contains M input feature values and a single class label, Y=y_k, from a set of K possible class labels.[*] Every input vector can be considered to be an event of order M+1 in input space. Each element of an M-dimensional input feature vector, x_j, $j \in \{1, 2, \cdots, M\}$, can have one of v_j discrete observed values drawn from the set of possible values or primary events describing feature j. Each possible combination of m primary events selected from *within* a vector can be considered a sub-event of order m, $m \in \mathbb{I}, 1 \leq m \leq M+1$.

Primary (or first order) events are represented as $\mathbf{x}_l^1$, while in general an event of order m is represented as $\mathbf{x}_l^m$, with l indicating a particular sub-event within the list of all sub-events of order m occurring in a particular input event $\mathbf{x}$. Events of interest with respect to classification must be of order 2 or greater and be an association of at least one input feature value (a primary event) and a specific class label.

PD analysis begins by counting the number of occurrences of all observed events among the N vectors forming the training data. Statistically significant events (or "patterns") within this set are then discovered

[*] A summary of the variable definitions used throughout this thesis is supplied in Appendix A at the end of the document. This is intended to form a general reference for the notation throughout the development of the discussion.

by using a residual analysis technique.

3.2 Definitions for Residual Analysis

Definition 3.1 (Standardized Residual)**:**

The standardized residual is defined as the ratio of the simple residual (*i.e.*, $o_{\mathbf{x}_l^m} - e_{\mathbf{x}_l^m}$) to the square root of its expectation (Haberman, 1973):

$$z_{\mathbf{x}_l^m} = \frac{o_{\mathbf{x}_l^m} - e_{\mathbf{x}_l^m}}{\sqrt{e_{\mathbf{x}_l^m}}} \tag{3.1}$$

where

$e_{\mathbf{x}_l^m}$ indicates the number of occurrences of $\mathbf{x}_l^m$ expected from observation of a training set of known size under an assumed model of uniform random chance and

$o_{\mathbf{x}_l^m}$ is the observed number of occurrences of $\mathbf{x}_l^m$ in a training data set.

In equation (3.1) it is important to note that the expectation value $e_{\mathbf{x}_l^m}$ cannot fall to zero under the assumed model used here. This is because $e_{\mathbf{x}_l^m}$ is equivalent to a linear scale of the number of available training examples, as it is produced by dividing the available training examples among the number of quanta. The only way a zero value could therefore be produced is by having no observed values for some feature. If this were to occur (*i.e.*, if all the values for a given column in the training data were missing) the adjusted residual would be undefined, however in this pathological case there is no discovery system that would be able to proceed. It suffices to proceed under the assumption that there will be a non-zero number of exemplars available for every input feature.

The standardized residual provides a normalization of the difference $o_{\mathbf{x}_l^m} - e_{\mathbf{x}_l^m}$ such that $z_{\mathbf{x}_l^m}$ has an asymptotic Normal distribution (for a proof, see Haberman, 1973, 1979). The standardized residual does not, however, have a unit standard deviation; for this reason we proceed to further scale $z_{\mathbf{x}_l^m}$ so that the distribution is $\mathcal{N}(0, 1)$ by considering the adjusted residual.

Definition 3.2 (Adjusted Residual)**:**

The adjusted residual is a normalization of the standardized residual (also Haberman, 1973, 1979) that achieves a $\mathcal{N}(0, 1)$ distribution by adjusting the variance of the previously zero-mean Normally distributed deviate of equation (3.1):

$$r_{\mathbf{x}_l^m} = \frac{z_{\mathbf{x}_l^m}}{\sqrt{v_{\mathbf{x}_l^m}}} \tag{3.2}$$

where $v_{\mathbf{x}_l^m}$ is the maximum likelihood estimate of the variance of the $z_{\mathbf{x}_l^m}$ value of (3.1); as given by Wong and Wang (1997), this is:

$$v_{\mathbf{x}_l^m} = var\left(z_{\mathbf{x}_l^m}\right) = var\left(\frac{o_{\mathbf{x}_l^m} - e_{\mathbf{x}_l^m}}{\sqrt{e_{\mathbf{x}_l^m}}}\right) = 1 - \prod_{\substack{x_{li} \in \mathbf{x}_l^m \\ j=1}}^{m} \left(\frac{o_{x_{li}}}{N}\right) \tag{3.3}$$

where $o_{x_{li}}$ is the number of occurrences of the primary event $x_{li} \in \mathbf{x}_l^m$ ($\mathbf{x}_l^m$ is the current event being examined) and N is the total number of observations made (*i.e.*, the number of rows in the training data set).

The benefit of the adjusted residual is that it scales the space of the standardized residual into that of a Normal deviate with unit standard deviation. Using the resulting $\mathcal{N}(0, 1)$ space, we can easily calculate observed deviation from expectation.

3.3 Pattern Identification Using Residual Analysis

The test performed on each $\mathbf{x}_l^m$ event to determine whether it is "significant" simply compares the observed number of occurrences of the event with the expected number of occurrences under the null hypothesis that the probability of the occurrence of each component primary event is random and independent.

The observed number of occurrences of $\mathbf{x}_l^m$ is represented as $o_{\mathbf{x}_l^m}$ and the expected number of occurrences, $e_{\mathbf{x}_l^m}$ is

$$e_{\mathbf{x}_l^m} = N \prod_{\substack{x_{li} \in \mathbf{x}_l^m \\ i=1}}^{m} \left(\frac{o_{x_{li}}}{N}\right), \tag{3.4}$$

where $o_{x_{li}}$ is the number of occurrences of x_{li}, itself a primary event drawn from the event $\mathbf{x}_l^m$.

To select significant events, the adjusted residual $r_{\mathbf{x}_l^m}$ defined in (3.2) is used as it provides a $\mathcal{N}(0, 1)$ distributed z statistic (*i.e.*, a statistic drawn from a normal distribution with zero mean and unit standard deviation). The value $r_{\mathbf{x}_l^m}$ defines the relative significance of the associated event $\mathbf{x}_l^m$. The PD algorithm deems an event to be significant if $|r_{\mathbf{x}_l^m}|$ exceeds 1.96, defeating the null-hypothesis with 95% confidence.

Events capturing significant relationships between feature values in the training data are termed "patterns." Patterns are used to suggest the class labels of new input feature vectors. Patterns containing a value for the label column are termed "rules."

Significance is calculated in absolute terms because combinations of events which occur significantly less frequently than would be expected by the null-hypothesis (patterns with a negative $r_{\mathbf{x}_l^m}$) are just as significant and potentially discriminative as those that occur more frequently. Such patterns may be used to contra-indicate a specific class label.

3.4 Classification

Classification in PD functions by combining the implications of extracted patterns to indicate a class label. The patterns used are chosen based on their discriminative ability.

3.4.1 Weight of Evidence Weighted Patterns

In order to measure the discriminative power of a pattern, Wang (1997) suggests the use of the "weight of evidence" statistic or WOE.

Definition 3.3 (Weight of Evidence)**:**

Letting $(Y{=}y_k)$ represent the label portion of a given pattern $\mathbf{x}_l^m$, the remaining portion (consisting of the input feature values) is referred to as $\mathbf{x}_l^\star$. The mutual information between these two components can be calculated (Wong and Wang, 1997) using:

$$I(Y{=}y_k : \mathbf{x}_l^\star) = \ln \frac{\Pr(Y{=}y_k|\mathbf{x}_l^\star)}{\Pr(Y{=}y_k)} \tag{3.5}$$

A WOE in favour of or against a particular labelling $y_k \in Y$ can be calculated as

$$\mathrm{WOE}\left(\frac{Y{=}y_k}{Y{\neq}y_k}\middle|\mathbf{x}_l^\star\right) = I(Y{=}y_k : \mathbf{x}_l^\star) - I(Y{\neq}y_k : \mathbf{x}_l^\star) \tag{3.6}$$

or

$$\mathrm{WOE} = \ln \frac{\Pr(\mathbf{x}_l^\star, Y{=}y_k)\ \ \Pr(Y{\neq}y_k)}{\Pr(Y{=}y_k)\ \ \Pr(\mathbf{x}_l^\star, Y{\neq}y_k)}. \tag{3.7}$$

WOE thereby provides a measure of how discriminative a pattern $\mathbf{x}_l^\star$ is in relation to a label y_k and gives us a measure of the relative probability of the co-occurrence of $\mathbf{x}_l^\star$ and y_k (*i.e.*, the "odds" of labelling correctly).

The domain of WOE values is $[-\infty \cdots \infty]$, where $-\infty$ indicates those patterns ($\mathbf{x}_l^\star$) that never occur in the training data with the specific class label y_k; ∞ indicates patterns which only occur with the specific class label y_k. These $\pm\infty$ valued WOE patterns are the most descriptive relationships found in the training data set as any non-infinite WOE indicates a pattern for which conflicting labels have been observed.

WOE-based support for each y_k (possible class label) is evaluated in turn by considering the highest-order pattern with the greatest adjusted residual from the set of all patterns occurring in an input data vector to be classified, and accumulating the WOE of this pattern in support of the associated label. All features of the input data vector matching this pattern are then excluded from further consideration for this y_k, and the next-highest order occurring pattern is considered. This continues until no patterns match the remaining input data vector, or all the features in the input data vector are excluded. This "independent" method of selecting patterns attempts to accumulate their WOE in way which estimates the accumulation of the probabilities of independent random variables. Once this is repeated to consider each y_k, the label with the highest accrued WOE is assumed as the highest likelihood match and this class label is assigned to the input feature vector.

3.4.2 Pattern Absence

It is possible that an input vector may contain only primary events which are not associated with any pattern. This is most likely to occur when an input event matches a hyper-cell that, in the training set, occurred with a probability similar to that of the null hypothesis. In this case, no information about the possible classification of this event is available, and no pattern is matched, as described in Figure 2.1 on page 12. When this occurs, the classifier leaves an input vector unclassified; this will be a distinct outcome in addition to the set of assigned labels through the addition of an extra "UNCERTAIN" label. This behaviour provides a strength not commonly seen in classifiers; a possibility of a graceful "no decision" in scenarios when insufficient information is available to make a robust decision.

The alternative action in this case would be to choose the class with the greatest overall probability of occurrence, irrespective of **x**, *i.e.*,

$$Y = y_k \quad \text{such that} \quad k = \operatorname*{argmax}_{k=1}^{K} \ \Pr(y_k) \tag{3.8}$$

however for DSS purposes these cases are intentionally flagged separately in the work described here.

3.5 Quantization: Analysis of Continuous Values

Originally, PD was developed to deal with integer or nominal data, terming the observation of each discrete datum as an event.

In most real-world problems data is continuous-valued and must be quantized to be used by the PD

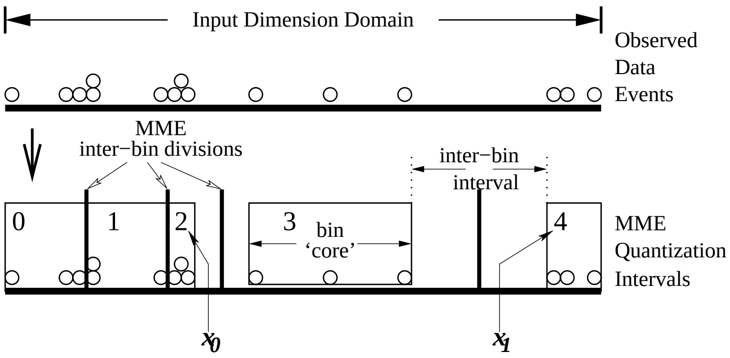


Figure 3.1: MME Partitioning

algorithm. In order to define events over continuous-valued input data the feature values must be first discretized. In this work, a marginal maximum entropy (MME) partitioning scheme as described by (Gokhale, 1999; Chau, 2001) is used, this divides the continuous data into bins with Q quantization intervals per feature. The general idea is that for a specific feature the data values assigned to each bin have a local similarity, and that each bin contains the same number of assigned feature data values.

This is achieved over the set of observed values by:

- sorting all the values for a given feature j, $j \in \{1, 2, \cdots, M\}$;
- dividing the sorted list into q_j bins of $\frac{N}{q_j}$ values each;
- calculating a minimum cover or "core" of each bin;
- covering gaps between the calculated cores of adjacent bins by extending the bin boundaries to the midpoint of the gaps.

The creation of MME bounded divisions based on grouping of input data values is described in Figure 3.1 where the first two rows illustrate the construction of MME based input space quantization intervals. In the top row, an input space is shown with training feature data shown as circles. The distribution of the circles along the axis indicates the values of the features observed; where feature values occur very close together, the values are "piled up" on top of each other.

Using this data, the second row shows the division of the sorted data points into MME quanta of three points each, creating five MME "cells" numbered 0 to 4. These MME cells have differing sizes, as the distribution density of the continuous values is not constant; this has also caused variability in the widths of the gaps between MME cell cores. There is no discernible gap between quanta 0 and 1 as there is no gap between the third and fourth data point as counted from the left. In the case of the cores of cells 3 and

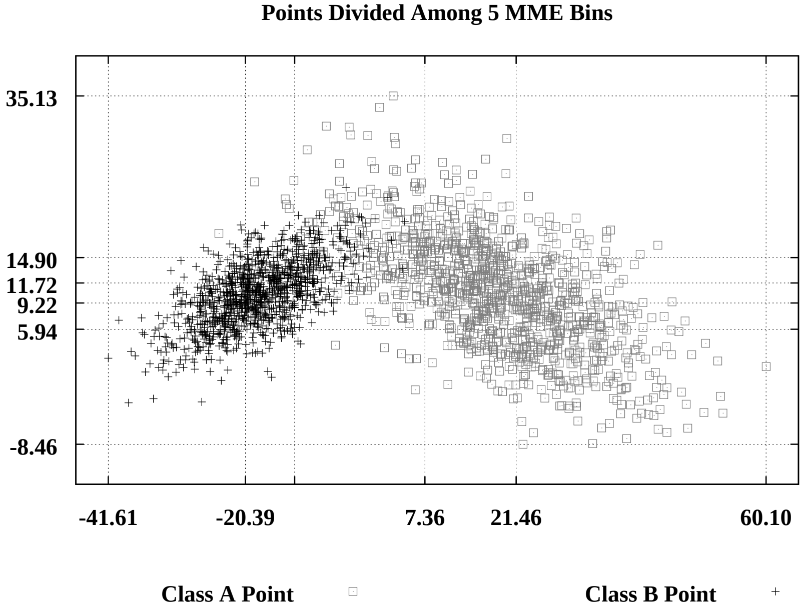


Figure 3.2: Example MME Division of 2-class Unimodal Normal Data

4, there is a substantial gap in which no points were recorded in the training set shown here.

These gaps are closed by expanding the bin and moving the boundaries to the midpoints of the gaps, covering the input domain completely. New data points are then assigned to the bin whose interval includes their value as shown in Figure 3.1 on the second row with the assignment of x_0 and x_1 to MME quantization intervals 2 and 4, respectively. This assignment is not necessarily what is desired, as when there is a large gap in the discretization of the training data there is considerable vagueness regarding the exact boundary of the MME cells. Decisions made based on this type of data are more uncertain than decisions made if the data falls into the core area of a bin.

There are two types of imperfection to be captured: the vagueness based on the location of the bin boundaries, and uncertainty in the measurement of both the training and testing data points.

Figure 3.2 shows the MME divisions constructed for the two displayed classes, which are unimodal, normally distributed data with some covariance. The training data used for the bin construction is overplotted on top of the bin boundaries. The data have been divided among five MME bins along each axis. As can be seen, the MME divisions are constructed without regard to class association — the quantization is done by counting points in both classes together (*i.e.*, class-independent quantization). In regions of the graph where the data points are sparse, the MME divisions are wider. Note that in the centre of the y-axis where the density of the data points is high, a narrower bin will accommodate the same number of points; along the x-axis, the bins are wider throughout as the separation of the distributions along this axis means that the density is more constant than is the case in y.

3.6 Expectation Limits and Data Quantization

The relationship between decisions made by the PD algorithm in a continuous-valued domain and the configuration of the system is dependent upon two major factors: the number of training records available for PD and the quantization resolutions for each feature q_j that govern how many different MME bins each feature is divided into.

Given a fixed number of training data records, it is important to choose the q_j to ensure that enough records are present to allow a statistically sound decision to be made regarding the discovery of each possible high order pattern. As the occurrence of a high order event is simply the product of the occurrence of the primary events (which MME attempts to keep equal across all bins), we can calculate an estimate of this value by simply calculating a product relating the number of rows of training data available (N), the quantization resolutions used for MME (q_j) and the number of features used to represent the class distribution (M), which defines the highest order observable event. This relation is

$$\mathcal{E}_{\mathbf{x}_l^m} = N \prod_{\substack{x_{li} \in \mathbf{x}_l^m \\ i=1}}^{m} \frac{1}{q_j}, \quad j = \text{index}(i, \mathbf{x}_l^m), \tag{3.9}$$

in which the function "index($i, \mathbf{x}_l^m$)" selects the column index of primary event i within the poly-order event $\mathbf{x}_l^m$. This function is required as pattern $\mathbf{x}_l^m$ may well be constructed via a subset of the M input features available in $\mathbf{x}$ and therefore j may not be valid for all values in the interval $\{1, \ldots, M\}$.

In equation (3.9) an increase in any q_j decreases $\mathcal{E}_{\mathbf{x}_l^m}$; an increase in m (the order of the event) also decreases $\mathcal{E}_{\mathbf{x}_l^m}$. Essentially, given a fixed amount of training data (or a fixed amount of information), dividing the data into a greater number subdivisions reduces the amount of information in each. This is the well-known "curse of dimensionality" (Bellman, 1961) which affects any multi-variate data inference technique. To test high order events for their possible significance as patterns we are therefore obliged to use a lower q_j, given the same N; effectively consolidating our data at a very coarse resolution to ensure that each division has enough data to be able to support a statistical inference procedure.

As a precaution against recording spurious patterns when $\mathcal{E}_{\mathbf{x}_l^m}$ is low, the PD classifier implemented will not consider as patterns any events for which the expected number of occurrences is less than 5. This is desired in order to prevent the discovery of high-order patterns caused by the occurrence of only one or two instances of a high-order event (instances that were possibly generated by chance). Such a pattern, if accepted, will in all likelihood have an infinite WOE, as the chances of observing the same randomly-generated pattern while observing a different label is quite small.

The existence of such patterns may diminish classifier performance, especially when the total number of features, M, is quite large. The use of an infinite weighted but spurious pattern would then corrupt the evaluation of any remaining features in the induction of the correct class label value.

Referring back to the introduction of PD in Figure 2.2 on page 13, we see all cells in this figure will be discarded as the expectation at this quantization places only one element into each cell. While first inspection may indicate that this restriction is unduly harsh, a small degree of consideration shows that in fact there is far too little information to proceed to any sound conclusion, and indeed the data pictured in Figure 2.2 could easily have occurred by random chance.

The purpose of equation (3.9) is to ensure that we do not extend the level of inference discovered in a data set beyond the amount of information actually present; instead we restrict ourselves to the patterns which can be discovered with confidence, omitting patterns whose rationale cannot be rigorously defended.

Using equation (3.9) it is therefore possible to choose $Q = q_1 = q_2 = \cdots = q_m$ based on a knowledge of the maximum order event expected and the number of training examples available.

3.7 Summary

The PD algorithm provides a pattern (rule) selection mechanism based on a simple analysis of residuals, which are simple to calculate and straight-forward to explain.

In order to adapt PD to a continuous domain, a class-independent quantization scheme was used. This MME quantization is transparent and simple to explain, however any relatively coarse quantization has the potential to raise problems when the resolution of the data is high.

The primary advantage of these two techniques is the simple statistical basis, allowing excellent explainability.

Part II

Hybrid System Design and Validation on Synthetic Data

Chapter 4

Construction of Fuzzy Pattern Discovery

Research is what I'm doing when I don't know what I'm doing.

— Wernher von Braun

A scheme to implement a "fuzzified" version of the PD algorithm is introduced in this section. Two main factors within the PD system are considered: the crisp nature of the quantization bins, and the lack of any fuzzy framework for the PD patterns.

4.1 Rationale

One of the primary advantages of a fuzzy inference system (FIS) is the linguistic framework supporting the inference and resulting transparency of the decisions made. This framework allows the robust exploration of the knowledge structure necessary for use in decision support. A new fuzzy inference system (FIS) is introduced here performing continuous and mixed-mode data classification based on rules recognized as patterns by the PD method of statistically based pattern extraction. The classification performance of this new FIS across a series of different class distributions is examined and discussed.

The rules utilized in this classification based FIS are created by the PD algorithm, based on an MME quantization of the input feature space.

The creation of an FIS based on the PD framework will ameliorate the high cost of quantization expressed by an MME based PD as well as provide a linguistic basis for a DSS.

The main advantage of the rules exported from the PD algorithm is the transparency and statistical validity provided by the analytical techniques used to create them. The main problem with the use of these rules is the characterization of the weights associated with the rules by the PD algorithm; these weights do not fall into the normalized fuzzy membership domain, but rather in the domain of the relative likelihood of contrary observation, and are bounded only by $[-\infty \cdots \infty]$.

What is being proposed in this work, therefore, is a marriage between a probabilistically based rule extraction system and a fuzzy inference methodology for rule evaluation and expression. It may seem that this conjunction is a poor fit, as every fuzzy researcher has pointed out that fuzzy inference and probability are different things, beginning with Zadeh (1968). What is proposed in this work is not a mixing of the metaphors of fuzzy and probability, but rather a co-operation between the two, using fuzzy sets to represent vagueness where appropriate and using probability theory to explain observation. By using the strengths of the two paradigms together, the hybrid approach suggested here provides a means of providing a context for decision making superior to either alone.

4.2 Fuzzification of Input Space Data Divisions

One of the advantages of fuzzy inference is the blurring of bin boundaries between adjacent input ranges and concordant reduction in quantization costs. The construction of fuzzy input membership functions for this purpose is described here.

Figure 4.1 extends Figure 3.1 from page 39, indicating the production of FIS input membership functions from the vagueness in the constructions of the MME quanta. When examining Figure 4.1, we see there are two types of imperfection to be captured: the vagueness based on the location of the bin boundaries, and uncertainty in the measurement of both the training and testing data points.

Consider the boundaries of bin 0. When there is no gap between cells the cell-boundary vagueness is low. The crisp, well-defined bin bounds of this cell allow the calculation of WOE to be performed with a higher certainty than is possible when vagueness about the boundary location exists. As bin 1 (which will define the "core" of a fuzzy set) is crisply defined by its borders with the "cores" of bins 0 and 2, any new measured values in this feature in the vicinity of the core of bin 1 can be crisply assigned based on these bounds.

In contrast, when there are significant gaps between the bin cores, such as those bordering bin 3, the fuzzy concept of "vagueness" captures the imperfection present in the bin-boundary problem. As shown on the lower-most line of Figure 4.1, this vagueness is captured when fuzzy input membership functions are constructed with a flat central area (μ=1) corresponding to the defined region of the MME bin core, and with extension of the support of the set into trapezoidal ramps projecting into the area of vagueness between the quantized MME bins.

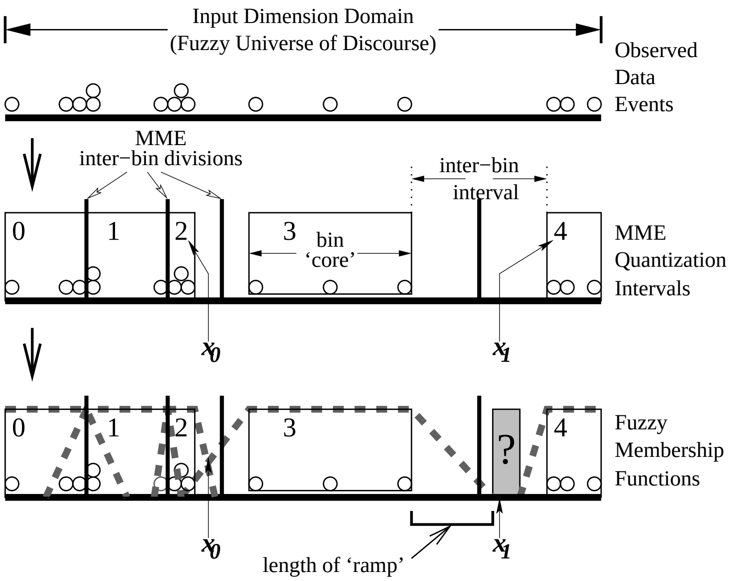


Figure 4.1: Fuzzy Mapping of MME Partitioning

4.2.1 Creation of Fuzzy Membership Functions

Using the following rationale, we create fuzzy membership functions based on the location of the bin core regions:

- the degree of uncertainty in the boundary of a bin is related to both the width of adjacent bin cores and the distance between them. Cores which abut have less uncertainty than those with a large inter-core gap; similarly, the degree to which an inter-bin gap is "large" cannot be evaluated without a knowledge of the width of the bin cores themselves;
- limiting the extent of the support ensures that the locality of inference of information regarding MME assertions is maintained (*i.e.*, we maintain the assumption that as a measurement deviates from some fixed constant, assertions made on the measured value will begin to differ from those made on the constant);
- if the point is farther from known data than we can reasonably extend the locality of inference, the preferred behaviour is to discard decisions altogether rather than make a classification based on extrapolating behaviour trained using distant exemplars.

The support of a trapezoidal fuzzy set is therefore extended in each direction from the bin core in order to expand the domain covered by the bin. The length of the extension of the support in each direction (the length of the trapezoidal ramp) is set to the minimum of:

- the distance from the edge of the current bin core to the midpoint of the neighbouring bin core (to limit the number of fuzzy membership functions which overlap);
- 1/2 the width of the current bin core (to restrict the maximum distance to which a bin may have influence, based on its width).

4.2.2 Fuzzy Membership Function Attributes

The resulting fuzzy membership functions may abut in one of three ways:

1. the overlap may be complete, as in the boundary between bins 0 and 1—in this case the support of the set projects into the adjacent bin and the membership across the inter-core boundary never drops below 1.0;
2. the regions of support of adjacent sets may overlap and cover the inter-core region, as in Figure 4.1 at point x_0 which will be assigned to both fuzzy input membership sets 2 and 3 with non-zero membership;
3. the regions of support may not meet, as is the case at point x_1, which is sufficiently far from both neighbouring core areas that it is outside the extent of the support for both sets.

This third option causes the creation of regions in the input space which would have no assignment in the fuzzy logic system; in order to avoid the instability which this would otherwise cause, crisp membership functions are inserted into these areas which will cause the output label "UNCERTAIN" to be assigned if any input values fall into these regions.

The above scheme allows the point x_1 (which was assigned to a distant MME bin in the PD row) to be discarded, producing a classification of "UNCERTAIN" for x_1, which allows the system to avoid performing a classification with insufficient confidence. This strategy maintains the traditional transparency of fuzzy systems and extends the ability of a user to inspect the rationale of the decision through to the input domain. The user can rely upon the statistical validity of the MME quantization (see Gokhale, 1999; Chau, 2001) and further see that this quantization is not disturbed by unduly extending the support of the fuzzy membership functions away from the training-defined core regions. This will allow the user to "drill down" through the final rule firings to determine the input value matches and maintain a high degree of confidence in the conservatism of the overall system. The mechanism supporting this will be discussed in Chapter 10.

4.3 Use of Pattern Discovery Based Rules

Using the patterns provided by the PD algorithm as rules for a FIS also requires some thought for two reasons: the pattern weightings used within the PD algorithm are not $[0 \dots 1]$ bounded, and the PD logic for pattern selection and use differs greatly from the firing of rules in a general FIS.

Aside from weighting and selection, the patterns created by the PD system are very similar to fuzzy rules, as they enumerate the associations observed between input events and output labels. These can be easily mapped into the association of membership in (collections of) input fuzzy sets with membership in output universes describing the degree of association with each label.

The PD algorithm generates, however, both positive logic patterns which support a given class, and negative logic patterns which indicate that the associated class is unlikely, given the input events observed. Each positive logic pattern provides a vote supporting a single class. Negative logic patterns may also exist, decreasing the support for the indicated class, and other positive logic patterns may exist increasing the support for a conflicting conclusion. Each possible class may therefore have several assertions in support or in refutation of its candidacy as a class label. The FIS must therefore consider assertions of both support and refutation for each possible class; these assertions must be combined into one overall assertion value, $\mathcal{A}_k$, describing the support or refutation for each candidate label $k \in K$.

The WOE values associated with PD patterns provide a description of their discriminative power. In order to process these as rules weighted using WOE, a provision for a system to accumulate assertions from rules that have infinite weights within the structure of the hybrid FIS is introduced.

In order to facilitate this accumulation, the consequents of all rules are examined as hypotheses supporting or rejecting each output class.

Three schemes to implement this mapping are described in the following sections:

- a Mamdani (1974) based inference using WOE weighted rules;
- a mapped WOE weighting scheme using fuzzy inference and
- a simpler occurrence based weighting scheme with fuzzy inference.

4.3.1 Mamdani Inference Using WOE Weighted Rules

This strategy is supplied to give a baseline comparison in performance using a very simplistic implementation using a fuzzy set method based on Mamdani's (1974) initial implementation of a fuzzy logic system.

Each rule (pattern) will fire and generate an assertion $a_{\mathbf{x}_l^m K}$ in an output space for the associated class K, based on the WOE of the rule using the ranges and assertions shown in Table 4.1.

Table 4.1: Mamdani Output Values for WOE

WOE Value	**Output Class Name**	**Centre**
$[\ \frac{1}{2}\omega \ldots \quad \infty]$	SUPPORTED	1
$(\quad 0 \ldots \ \frac{1}{2}\omega)$	SOMEWHAT-SUPPORTED	0.5
$(-\frac{1}{2}\omega \ldots \quad 0]$	SOMEWHAT-REFUTED	-0.5
$[\ -\infty \ldots -\frac{1}{2}\omega]$	REFUTED	-1

The value ω in the table is calculated as:

$$\omega = \max(|WOE_i|) \quad \forall\ i \quad |WOE_i| \neq \infty, \tag{4.1}$$

where i indexes the list of all rules, and $|\cdot|$ indicates absolute value.

Each output set described in Table 4.1 is defined as an impulse based set or fuzzy singleton; a set whose support is a single value

$$\mathcal{V}_{\mathbf{x}_l^m K} = \frac{\mu(a_{\mathbf{x}_l^m K})}{a_{\mathbf{x}_l^m K}}, \tag{4.2}$$

where

$a_{\mathbf{x}_l^m K}$ is an the value created by the firing of a fuzzy rule through the use of Table 4.1 based on pattern $\mathbf{x}_l^m$ supporting class K and

μ is the fuzzy membership match of the rule firing.

Firing of rules using this Mamdani-style assertion logic creates a collection of singletons in one of K defuzzification universes, each of which describes the degree of support for a particular classification.

The degree of the rule firing is used to scale the membership of the impulse set, combining the degree of input membership match of the rule (pattern) with the assertion value provided by the WOE value of the rule.

After all rules have been processed, a centroid-based calculation is applied to the universe of discourse to achieve a scalar output in the usual manner. The value of this scalar $\mathcal{A}_k$ is then taken as an assertion of support or refutation of this classification. This assertion has the range $[-1 \ldots 1]$ where -1 is total refutation, 1 is total support and 0 indicates no opinion.

This is similar to the "certainty factor" introduced by the MYCIN system of Shortliffe (1976); Buchanan and Shortliffe (1984).

This procedure will be termed "MAMDANI" in the discussion section.

4.3.2 Using WOE Directly With Fuzzy Rules

The main problem with the use of WOE weighted rules is the fact that the rules have possibly infinite weights. The range $[-\infty \cdots \infty]$ cannot be applied directly within the standard *t*-norm/*t*-conorm proposed by Zadeh (Yager, Ovchinnikov *et al.*, 1987) as any conflicting $\pm\infty$ values will leave the results of the calculation undefined. A further consideration is that WOE values are measured in "relative likelihood" and are therefore not $[0, 1]$ bounded; instead arbitrarily large finite values may be observed, along with any infinities.

The inconveniently bounded WOE based assertion for a particular classification provides a mapping of the "degree of support" by which the given classification is supported by the rule, where larger positive values indicate higher degrees of support, and larger negative values indicate higher degrees of refutation of the classification.

To convert WOE values into a space bounded by $[-1 \ldots 1]$ in the defuzzification universe the WOE based assertions are normalized by the maximum finite WOE value recorded in the PD rule set, independent of class. This ensures that all the finite WOE assertions will fall into the range $[-1 \ldots 1]$. The net effect of this normalization is to cause all $a_{\mathbf{x}_l^m K}$ assertions to fall into three groups:

1. assertions which fall into the range $[-1 \ldots 1]$;
2. assertions whose value $= \infty$ and
3. assertions whose value $= -\infty$.

Adjusted Centroid Calculation for Infinite WOE

Assertions in the $[-1 \ldots 1]$ range can be considered using a standard centroid calculation. In the infinite cases, the support/refutation value is simply calculated by counting the number of infinities. If more $+\infty$ values are observed than $-\infty$ values, an assertion of 1 is produced. If more $-\infty$ values are present, an assertion of -1 is produced. If there are an equal number of positive and negative infinite assertions, the output is marked "UNCERTAIN," regardless of the outcome of the assertions for the other class possibilities. As each rule (each $\mathbf{x}_l^m$) asserts a value ($\mathcal{A}_k$) in the range $[-\infty, -1 \ldots 1, \infty]$, we term this "WOE" weighting.

4.3.3 Occurrence Weighted Fuzzified PD Rules

A drawback of the Mamdani-style system is that fact that the granularity of the weight is lost in the assignment to one of a discrete number of output sets (assertion values). A similar potential problem with pure WOE is the potential to lose resolution through the combination of infinity values in WOE weighting. It is therefore of interest to construct a means of providing a functional mapping for rule weights into output assertions.

Definition 4.1 (Occurrence Based Weighting)**:**
A relative measure ($\mathcal{W}_{\mathbf{x}_l^m}$) of the discriminative power of the rules ($\mathbf{x}_l^m$) is created by using a weighting based on the number of occurrences of the supporting rule in the training data:

$$\mathcal{W}_{\mathbf{x}_l^m} = \begin{cases} \frac{o_{\mathbf{x}_l^m}}{o_{\mathbf{x}_l^\star}} & \text{if } r_{\mathbf{x}_l^m} \geq 1.96, \\ \\ \frac{o_{\mathbf{x}_l^m} - e_{\mathbf{x}_l^m}}{e_{\mathbf{x}_l^m}} & \text{if } r_{\mathbf{x}_l^m} \leq -1.96, \end{cases} \tag{4.3}$$

where

$o_{\mathbf{x}_l^m}$ indicates the number of occurrences of the event defining rule l, or $\mathbf{x}_l^m$, including the class label, as used in equation (3.2);

$e_{\mathbf{x}_l^m}$ is the expected number of occurrences of the event defining rule l, ($\mathbf{x}_l^m$) and

$o_{\mathbf{x}_l^\star}$ indicates the number of occurrences of only the input value portion of the event (*i.e.*, event $\mathbf{x}_l^m$ with the class label column excluded), noting that this input event may also occur with other class labels.

Note that in (4.3), the first proposition produces values in the range $[0 \ldots 1]$, and the second values in the range $[-1 \ldots 0]$. Values of $r_{\mathbf{x}_l^m} \in (-1.96 \ldots 1.96)$ will not be observed for significant patterns (or rules).

This method can be considered an implementation of the techniques of Takagi and Sugeno (1985) and Sugeno and Kang (1988) as it provides a functional representation which combines the membership value of the rule firing and a $\mathcal{W}_{\mathbf{x}_l^m}$ value for the rule weighting into the output of some function producing a set of fuzzy singletons which are collected together and defuzzified using a centroid defuzzification algorithm.

Each rule $\mathbf{x}_l^m$ is allowed to assert a value $a_{\mathbf{x}_l^m k} \in [-1 \ldots 1]$, supporting or refuting some possible classification k of input value $\mathbf{x}$. Overall support for each class, $\mathcal{A}_k$, is thereby based on a possibilistic scheme and is independent of the support for any other classes.

As this algorithm creates assertions supporting or refuting an output classification weighted purely on the observed occurrence of sub-events, we term this "Occurrence" weighting.

4.3.4 Selection of a Classification Label

For all the above schemes, once an assertion is calculated for each class for which any rule fires, the label is selected as follows:

1. locate the class containing the centroid which asserts the maximum value:

$$k^{\star} = \operatorname*{argmax}_{k=1}^{K} \mathcal{A}_k \tag{4.4}$$

2. if $\mathcal{A}_{k^{\star}} > 0$ then $Y{=}y_{k^{\star}}$
3. otherwise $Y{=}y_{\text{UNCERTAIN}}$

In the case where $Y{=}y_{\text{UNCERTAIN}}$, the classifier has produced a "soft failure"; that is, the classifier has decided that the given input data does not provide sufficient discriminative information to produce a reliable decision, and therefore no decision will be made. In a decision support context this characteristic is a welcome one, as it will enhance the reliability of the overall system.

4.4 Selection and Firing of PD Generated Rules

In the PD algorithm, the occurrence of a pattern is assumed to consume the information associated with the primary events describing the pattern. For this reason, each input feature can support at most one rule firing, in order to maintain the assumption of statistical independence of the input features. The rule to be fired is selected by evaluating the order and adjusted residual of all rules matching the input vector. Conversely, the fuzzy methodology allows all rules to fire, re-using the information in the associated input values, and letting the rule and membership weights govern to what extent each rule contributes to the final solution.

An implementation of the PD based independent selection is presented and evaluated against the standard fuzzy rule firing scheme, with the results included as Chapter 7. This comparison allows us to evaluate the "fuzzification" of the PD system by stages, comparing the performance as each feature of the PD algorithm is adapted to the fuzzy framework.

4.4.1 Fuzzy Rule Firing

This strategy, termed "ALL" in the results, refers to the standard fuzzy logic evaluation where all rules are fired. In the case of rules for which the input value falls outside the support of their defining membership functions, a zero membership value is attached to the assertion provided by the rule.

4.4.2 Independent Rule Firing

This strategy provides a rule evaluation procedure similar to that used in the PD system. This strategy is termed “Ind” in the results chapters.

The algorithm proceeds through the following steps:

1. produce a list $\mathcal{L}$ of all fuzzy variables provided through fuzzification of the input values which have a non-zero membership;
2. set search order o to be N, the number of inputs to the system;
3. place all rules of order o whose precedents exist in the list $\mathcal{L}$ into a list of matches, $\mathcal{M}$. If this search fails, repeat after setting $o := o - 1$;
4. If $o = 1$ has been reached, and no matches have been found, then stop;
5. find the rule $\rho_{\max}$ in $\mathcal{M}$ which has the highest adjusted residual;
6. fire rule $\rho_{\max}$, generating consequents as described in Section 4.4.1;
7. remove from $\mathcal{L}$ all the variables matching the precedents of rule $\rho_{\max}$;
8. if $\mathcal{L}$ is empty, then stop
9. repeat, starting from step 3, noting that the value of o continues to decrease with each iteration.

4.5 Summary

This chapter has outlined the extension of PD rules into a FIS, along with some changes to the inference required to evaluate the PD based rules. Three rule weightings and two rule firing techniques have been presented.

Rule Weighting :

- Mamdani rule weighting using a lookup table for fixed output assertions with rule-based membership
- WOE rule weighting with adapted centroid calculation to manage possible infinite values
- Occurrence rule weighting with simplified, non-infinite rule weights

Rule Firing :

- FZ – standard fuzzy firing of all rules
- Ind – firing of PD style rules, within a fuzzy context

The next chapters will now evaluate the performance of these various strategies.

Chapter 5

Synthetic Class Distribution Data and Analysis Tools

What is research but a blind date with knowledge?

— Will Harvey

In order to evaluate the hybrid system, several synthetic data distributions have been created. Synthetic data is used for analysis of this part of the work as the properties of the data are known in advance, allowing a deeper insight into both the performance triumphs and failures of the hybrid system with respect to attributes due to the construction of the data sets.

Four main types of synthetic data have been produced: unimodal covaried data, log-Normal covaried data, bimodal covaried data and spiral data. The construction of each of these types will now be explained.

5.1 Covaried Class Distributions

Covariance matrices for 4-feature $\mathcal{N}(0, 1)$ data were generated by using the variance values

$$\mathbf{V} = \{160, 48, 16, 90\} \tag{5.1}$$

arranged along the diagonal of a generating covariance matrix ($\mathbf{Cov}_{ii}$=$\mathbf{V}_i$). Each off-diagonal element $\mathbf{Cov}_{ij}$ was calculated through

$$\mathbf{Cov}_{ij} = \kappa \sqrt{(\mathbf{Cov}_{ii})(\mathbf{Cov}_{jj})} \tag{5.2}$$

$$
\begin{aligned}
\mathbf{Cov}_A &= \begin{bmatrix} 160 & -52.58 & -30.36 & -72 \\ -52.58 & 48 & 16.63 & 39.44 \\ -30.36 & 16.63 & 16 & 22.77 \\ -72 & 39.44 & 22.77 & 90 \end{bmatrix} \\
\mathbf{Cov}_B &= \begin{bmatrix} 48 & 16.63 & 39.44 & -52.58 \\ 16.63 & 16 & 22.77 & -30.36 \\ 39.44 & 22.77 & 90 & -72 \\ -52.58 & -30.36 & -72 & 160 \end{bmatrix} \\
\mathbf{Cov}_C &= \begin{bmatrix} 16 & 22.77 & -30.36 & 16.63 \\ 22.77 & 90 & -72 & 39.44 \\ -30.36 & -72 & 160 & -52.58 \\ 16.63 & 39.44 & -52.58 & 48 \end{bmatrix} \\
\mathbf{Cov}_D &= \begin{bmatrix} 90 & -72 & 39.44 & -22.77 \\ -72 & 160 & -52.58 & 30.36 \\ 39.44 & -52.58 & 48 & -16.63 \\ -22.77 & 30.36 & -16.63 & 16 \end{bmatrix}
\end{aligned}
$$

Table 5.1: Covariance Matrices for 4 Classes

using a κ value of 0.6.

This produced the covariance matrix for class A. The covariance matrix for class B was produced by setting

$$\mathbf{Cov}^B_{ii} = \mathbf{V}_{(i+1) \bmod 4}. \tag{5.3}$$

Class C and D were produced by substituting i+2 and i+3 respectively into (5.3). The resulting covariance matrices used to generate the 4-class data are shown in Table 5.1 for reference.

To use a covariance matrix to create data for some class k, a $\mathcal{N}(0, 1)$ data set with zero covariance was randomly generated, and a transform $\mathcal{T}$ was calculated to transform the data to have the desired covariance:

$$
\begin{aligned}
\boldsymbol{\mathcal{T}}_k &= \boldsymbol{\Phi}_k \boldsymbol{\lambda}_k \\
\boldsymbol{\mathcal{P}}_k &= (\boldsymbol{\mathcal{T}}_k \boldsymbol{\mathcal{P}}_{\mathcal{N}(0,1)}{}^T)^T + \boldsymbol{\mu}_k
\end{aligned}
\tag{5.4}
$$

where

$\boldsymbol{M}^T$ is the transpose of some matrix $\boldsymbol{M}$;
$\boldsymbol{\Phi}_k$ is the matrix of eigenvectors derived from the covariance matrix for this class ($\mathbf{Cov}_k$);
$\boldsymbol{\lambda}_k$ is the vector of eigenvalues for the covariance matrix;
$\mathcal{P}_{\mathcal{N}(0,1)}$ are the $\mathcal{N}(0, 1)$ uncoloured points;
$\mathcal{T}_k$ is the "colouring" transform to be applied;
$\boldsymbol{\mu}_k$ is the mean vector; and
$\mathcal{P}_k$ are the final coloured points.

These matrices generate clouds of data which intersect non-orthogonally and which have differing variances and covariances in each dimension, and in each class. The covariance values themselves have been chosen to provide both strong and weak covariances, across the various dimensions.

Class separations were produced using a combination of the variances of classes A and B within the set of covariance matrices, where the separation vector $\mathbf{c}$ was calculated using

$$\mathbf{c}_i = \sqrt[4]{(\mathbf{Cov}^A_{ii})(\mathbf{Cov}^B_{ii})}. \tag{5.5}$$

The four classes were separated into different quadrants in Euclidean space by projecting the mean vector of each class away from the origin by separation factors of

$$s_i \in \mathbf{S}, \quad \mathbf{S} = \left\{ \frac{1}{8},\ \frac{1}{4},\ \frac{1}{2},\ 1,\ 2,\ 3,\ 4 \right\}, \tag{5.6}$$

and combining s_i with $\mathbf{c}_i$ from (5.5) to produce centres for each factor of s located at $(+\frac{1}{2}s_i\mathbf{c}_i, +\frac{1}{2}s_i\mathbf{c}_i)$, $(+\frac{1}{2}s_i\mathbf{c}_i, -\frac{1}{2}s_i\mathbf{c}_i)$, $(-\frac{1}{2}s_i\mathbf{c}_i, +\frac{1}{2}s_i\mathbf{c}_i)$ and $(-\frac{1}{2}s_i\mathbf{c}_i, -\frac{1}{2}s_i\mathbf{c}_i)$, respectively.

5.2 Covaried Log-Normal Class Distributions

In order to evaluate the performance of the classifiers on data which is not $\mathcal{N}(0, 1)$ distributed, data was generated with a log-Normal distribution. This data was produced by taking each point generated in a $\mathcal{N}(0, 1)$ distribution using the method just described and calculating

$$\boldsymbol{p}' = e^{\boldsymbol{p}\xi}, \quad \boldsymbol{p} \in \mathcal{N}(0, 1) \tag{5.7}$$

to transform each point $\boldsymbol{p}$ to be log-Normally distributed, controlled by the shape parameter ξ. Each point in the resulting log-Normal distribution is then transformed using (5.4), using the covariance matrix. Sepa-

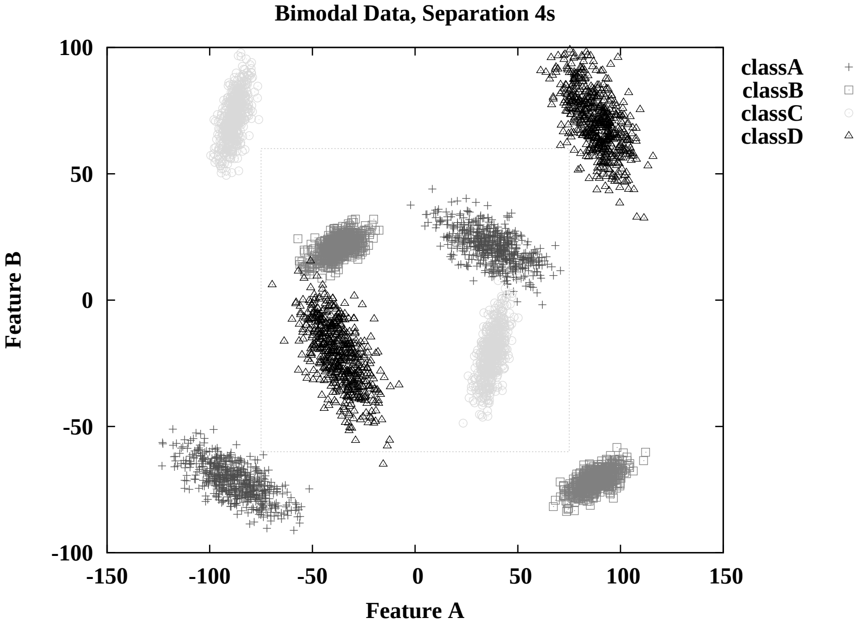


Figure 5.1: Bimodal Data Points

rate experiments were done with $\xi \in \{1.0, 1.5, 2.0\}$. The skewness value of the resulting class distributions was on average 4.58 when ξ=1.0; was 13.80 when ξ=1.5 and was 27.33 when ξ=2.0.

5.3 Covaried Bimodal Class Distributions

The bimodal data was created by starting with the s_i cluster locations of the linearly separable class distributions and then adding a second set of points for each class. The location of the second cluster is set by projecting the mean away from the origin in a diametrically opposite direction, with an extra translation of $4\sqrt{v_{\max}}$, where $v_{\max}$ is the maximum variance value specified in the set of variances, $\mathbf{V}$ (*i.e.*, 160.0). Thus, along with a cluster of points centred at (s, s), a second cluster would be placed at $(-4s\sqrt{v_{\max}}, -4s\sqrt{v_{\max}})$. Each cluster contained half the total number of points.

This algorithm was repeated for all sets, generating a layout of pairs of clusters around the origin, shown in the sample data illustrated in Figure 5.1 for s=8, which displays the first two dimensions of the four-feature data and shows clearly that no line can be drawn across the field to separate any one class from any other.

In Figure 5.1, the four modes shown in the centre (within the dotted square) indicate the distribution of points for Unimodal data with s=8.

5.4 Spiral Class Distributions

2-class data was produced by using the spiral equation

$$r = \rho(2\pi\theta) + r_0, \tag{5.8}$$

which relates the input variable θ to a radius where r_0 is the base radius at which the spiral begins and ρ is the scale, or acceleration of the spiral.

The spiral defined in (5.8) accelerates out from $(0, 0)$. For each (r, θ) point chosen on this spiral, a value from a $\mathcal{N}(0, 1)$ distribution was chosen to apply a scatter in radians to the θ value of each generated point, while maintaining the same radius. To generate 4 feature points, a second scatter is chosen, to perturb the data in the third and fourth dimensions also. The second class was generated by choosing a similar set of points. Separation was introduced by rotating the entire set around the origin by a specified amount in units of π radians. Maximum separation therefore is 1.0. Data was generated using σ=1.0, ρ=0.5 and r_0=0.125. Two dimensions of a sample pair of class distributions with a separation of 1.0 (a half-turn, or 1.0π) is shown for 2-dimensional data in Figure 5.2.

5.5 Training Error

For each class distribution studied the effect of training error $\mathcal{T}_{\text{err}}$ on the performance of the classifiers was examined. In each summary table a column indicating $\mathcal{T}_{\text{err}}$=0.1 is included to indicate that 10% of the records for each class have had their true label value replaced with a value chosen randomly from the other class labels.

5.6 Jackknife Data Set Construction

For each class distribution 11 sets of randomly-generated data from each class were produced to create training and testing data sets. The data were then combined using a jackknife procedure. This popular performance measurement methodology is discussed in Duda *et al.* (2001) and is widely used in the

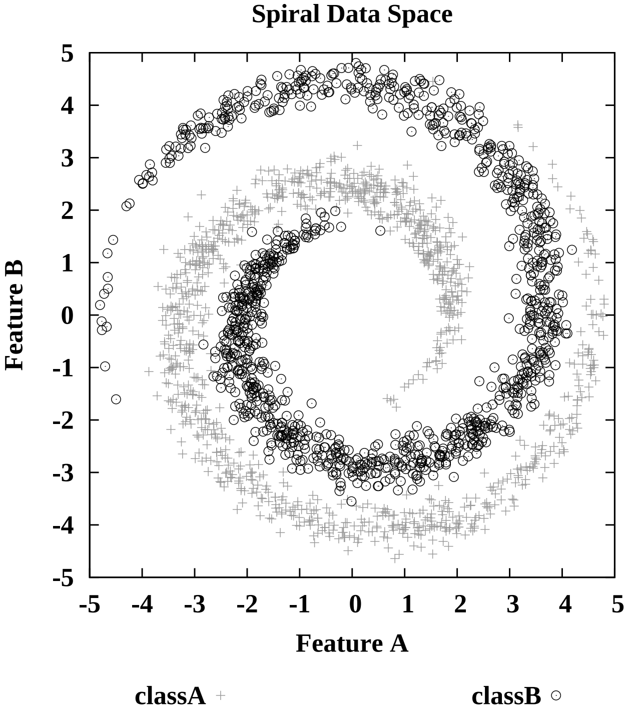


Figure 5.2: Spiral Data Points

literature. Essentially, the total set of Ω available data records is divided into a set of G groups, with N records per group. Each group $\mathcal{G}_g$, $g \in \{1, \ldots, G\}$ will be used once for testing; when group g is being tested, the training data is formed by concatenating all other records together

$$T_g = \bigcup_{\substack{i=1 \\ i \neq g}}^{G} \mathcal{G}_i. \tag{5.9}$$

In this way, each record is used for testing once, and used for training G times.

In each experiments described here G=11 separate jackknife runs have been created. The number of training records differs among experiments, however in each case the term N will be used to indicate the number of training vectors used for each class; in each case $\frac{1}{10}N$ will indicate the number of testing records. Specifically, when N=1000 training vectors per class, there are 100 testing points per class. This amount of training data reflects the quantity of data available for several real-world problems.

5.7 Classifiers Used in Comparison

Classification accuracies of the original PD and the hybrid PD/FIS were compared with those of back-propagation (BP) and a minimum inter-class distance (MICD) classifier in a series of experiments involving both linearly and non-linearly separable class distributions. The MICD classifier is simply a simple Naïve Bayesian classifier, and will be defined shortly.

In this work q_j was constant for all M input features and was set by the quantization resolution value Q.

5.7.1 Back-Propagation Classifier Construction

A simple BP network was constructed using the algorithms provided in Simpson (1991, pp. 112–120) and in Hertz *et al.* (1991, pp. 115–120). The learning rate and momentum for the classifiers were fixed across all BP configurations using the typical values of 0.0125 and 0.5, respectively. Training for each experiment was run for a maximum of 10^5 epochs, until the overall error dropped below 2.5×10^{-3}, or until the derivative of the error dropped below 1.25×10^{-3}. Several choices for the number of hidden nodes were studied (H=$\{5, 10, 20\}$).

5.7.2 Minimum Inter-Class Distance Classifier Construction

The MICD classifier takes an input record and applies a whitening transform transform calculated from the observed covariance. The smallest distance to a class mean in the whitened space is then the optimal maximum likelihood match, *if* the true class distribution is Normally distributed, linearly separable and sufficient points are available for the estimate of the covariance.

The transform for class k is given by Duda *et al.* (2001, pp. 41) as

$$\delta_k = \mathbf{x}^T \boldsymbol{W}_k \mathbf{x} + \mathbf{w}_k \mathbf{x} + w_k, \tag{5.10}$$

using transform components calculated by

$$\begin{aligned} \boldsymbol{W}_k &= -\frac{1}{2}\mathbf{Cov}_k^{\star\,-1}, \\ \mathbf{w}_k &= \mathbf{Cov}_k^{\star\,-1} + \boldsymbol{\mu}_k, \\ \text{and} \qquad & \\ w_k &= -\frac{1}{2}\left(\boldsymbol{\mu}_k^T \mathbf{Cov}_k^{\star\,-1} \boldsymbol{\mu}_k\right) - \frac{1}{2}\ln\left|\mathbf{Cov}_k^{\star\,-1}\right| + \Pr(k). \end{aligned} \tag{5.11}$$

In equation (5.11),

M^{-1} represents the inverse of matrix M;

$\mathbf{x}$ represents the vector of feature values, as in the discussion of PD;

$\mathbf{Cov}_k^{\star}$ is the covariance matrix calculated for class k;

$\boldsymbol{\mu}_k$ contains the mean vector for class k;

$\Pr(k)$ is the overall probability of occurrence of class k; and

$|\cdot|$ is the matrix determinant operation.

5.8 Weighted Performance Measure

To compare performance across classifiers, a statistic was constructed to summarize results across all separations into a single scalar value.

Definition 5.1 (Weighted Performance Summary Measure)**:**
Summarizing over all separations with greater emphasis placed on data from lower separations is performed using

$$P = \frac{\sum_{i=0}^{|\mathbf{S}|} \frac{p_i^{\text{classified}}}{s_i}}{\sum_{i=0}^{|\mathbf{S}|} \frac{1}{s_i}} \tag{5.12}$$

in which s_i is a separation value as defined in equation (5.6), the value $|\mathbf{S}|$ indicates the number of separations tested and

$$p^{\text{classified}} = \left(\frac{n^{\text{classified}}}{n^{\text{total}}}\right)\left(1 - \frac{n^{\text{error}}}{n^{\text{classified}}}\right), \tag{5.13}$$

where

$n^{\text{classified}}$ represents the number of records classified;

n^{error} is the number of records incorrectly classified; and

n^{total} is the total number of records processed (the sum of those classified and those left unclassified through assignment to the UNCERTAIN class label).

This generates a single (scalar) value representing the overall performance of the classifier weighed across all separations, such that performance at lower separations (where the problem is harder) is given more weight.

5.9 Summary

This chapter has described a series of synthetic data type distributions to be used for analysis. The motivation for evaluation on synthetic data is the available knowledge of the problem domain and the relationship this will have to the resulting performance.

Once the PD/FIS system is evaluated on synthetic data, real-world examples will be used to further test the best of the proposed FIS classifiers in later chapters.

Natively continuous classifiers (BP, MICD) have been chosen as comparative test cases, as on the continuous data type distributions described here their expected performance will be very high, providing a real and fair test for the PD/FIS algorithm.

Chapter 6

Synthetic Data Analysis of Pattern Discovery

You cannot have a science without measurement.

— Richard W. Hamming

Results evaluating PD are presented in the following sections, organized by the type of class distribution. The purpose of this chapter is to establish a performance baseline for the MME quantized PD system when functioning as a classifier in the continuous domain.

6.1 Covaried Class Distribution Results

Table 6.1 displays the weighted performance results calculated using equation (5.12) for the covaried Normal class distributions. Figure 6.1 displays the results over all separations tested for N=1000 and Figure 6.2 for N=10,000 training examples. The graphs in these figures are clipped at 0.5 to allow the lines to be more clearly visible, and the x-axis is plotted in $\log_2$ for clear separation of the data points.

In all of these displays, the PD classifier implementation suggested by Wang (1997) using independent pattern selection and WOE weightings is termed PD(WOE/IND) The modified OCCURRENCE weighting using all existing patterns is termed PD(OCCURRENCE/ALL). BP is shown only with H=10 hidden nodes as this was the highest performance configuration tested for both values of N. The covaried class distributions are quite rich in high-order information, and were easily separated by the MICD classifier, which

Table 6.1: Weighted Performance: Covaried Class Distribution

	Covaried				
Classifier	N**=100**	N**=500**	N**=1000**	N**=1000** $\mathcal{T}_{\text{err}} = 0.1$	N**=10,000**
PD(WOE/Indep);Q=5	0.51±0.07	0.61±0.03	0.66±0.03	0.65±0.02	0.69±0.01
PD(WOE/Indep);Q=10	0.52±0.07	0.64±0.03	0.63±0.02	0.63±0.02	0.69±0.01
PD(WOE/Indep);Q=20	0.48±0.05	0.60±0.03	0.61±0.02	0.60±0.02	0.66±0.01
PD(Occurrence/All);Q=5	0.50±0.07	0.65±0.03	0.66±0.03	0.65±0.02	0.68±0.01
PD(Occurrence/All);Q=10	0.51±0.07	0.64±0.03	0.66±0.02	0.65±0.02	0.69±0.01
PD(Occurrence/All);Q=20	0.44±0.04	0.59±0.03	0.60±0.02	0.59±0.02	0.68±0.01
BP;H=5	0.56±0.04	0.63±0.02	0.61±0.03	0.62±0.02	0.59±0.01
BP;H=10	0.60±0.04	0.69±0.02	0.71±0.02	0.70±0.02	0.70±0.01
BP;H=20	0.56±0.07	0.69±0.01	0.72±0.02	0.69±0.01	0.72±0.01
MICD	0.74±0.07	0.74±0.03	0.74±0.02	0.73±0.02	0.74±0.01

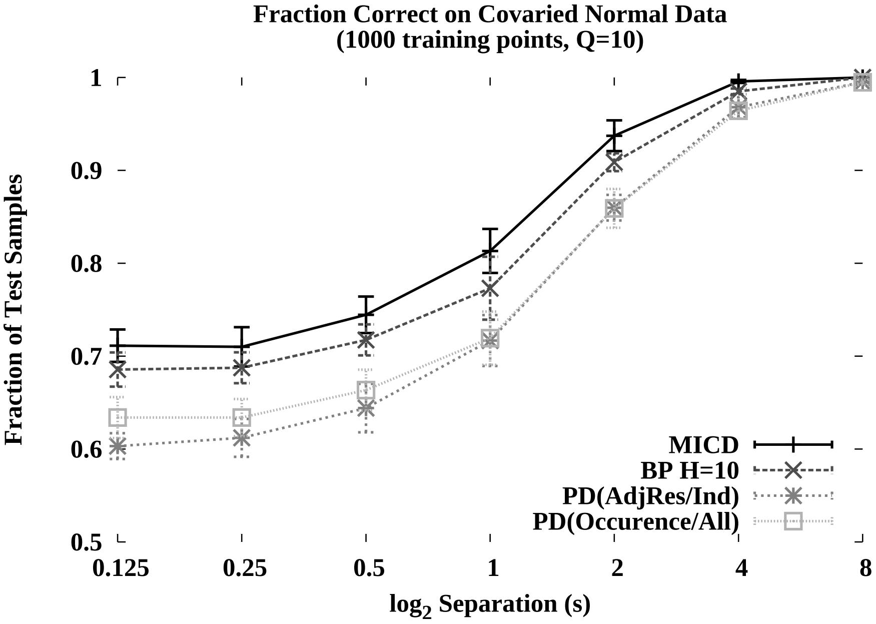


Figure 6.1: Covaried Data N=1000 Results

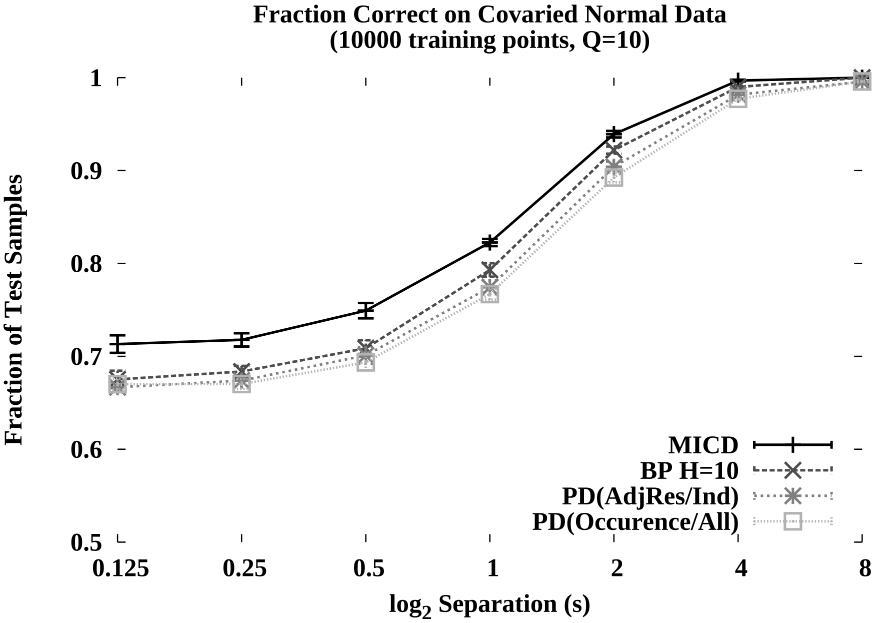


Figure 6.2: Covaried Data N=10,000 Results

can be seen providing an upper bound in each of Figures 6.1 and 6.2.

The performance of the PD classifier was somewhat lower than that of BP and MICD, though some small difference in performance between the PD(Occurrence/All) and PD(WOE/Ind) performance is visible in Figure 6.1, with the PD(Occurrence/All) performance being noticeably higher at lower separations. Once a separation of 1σ is reached, there is no longer a difference between the different PD classifier implementations.

Figure 6.2 shows no real difference in performance between either of the PD classifiers or the BP classifier, though the MICD classifier still shows that a marked difference exists in all these cases between the recorded performance and optimality in this case.

These analyses are supported by an examination of Table 6.1, which reports a generally stronger performance for PD(Occurrence/All) versus PD(WOE/Ind) (but both being somewhat weaker than that of BP classifier) and significant improvement in this linearly separable case when using the MICD classifier.

Considering the $\mathcal{T}_{\text{err}}$=0.1 column in Table 6.1, we can see that the PD classifiers can tolerate moderate training data error.

Table 6.2: Weighted Performance: Log-Normal Class Distribution

Classifier	**N=10,000 4c/4f** **LogNormal s=1**	 **LogNormal s=1.5**	 **LogNormal s=2**
PD(WOE/Indep);*Q*=5	0.93±0.00	0.86±0.01	0.82±0.01
PD(WOE/Indep);*Q*=10	0.94±0.00	0.87±0.01	0.82±0.01
PD(WOE/Indep);*Q*=20	0.93±0.00	0.84±0.01	0.77±0.01
PD(OCCURRENCE/ALL);*Q*=5	0.92±0.01	0.83±0.01	0.79±0.01
PD(OCCURRENCE/ALL);*Q*=10	0.93±0.00	0.86±0.01	0.82±0.01
PD(OCCURRENCE/ALL);*Q*=20	0.91±0.00	0.83±0.01	0.80±0.01
BP;*H*=5	0.95±0.00	0.86±0.01	0.78±0.02
BP;*H*=10	0.96±0.00	0.92±0.01	0.86±0.01
BP;*H*=20	0.97±0.00	0.94±0.01	0.90±0.00
MICD	0.91±0.01	0.61±0.03	0.36±0.02

Overall for these class distributions, it is apparent that Q=10 performs the strongest for the PD based classifiers; there is a notable performance decrease when comparing any of the tests using Q=10 with those using Q=5 or Q=20.

6.2 Covaried Log-Normal Class Distribution Results

As shown in Figure 6.3, the performance of the PD classifier remained high as the data distribution deviated from Normal, while the MICD classifier performance dropped remarkably as the skewness of the data increased. This is summarized across all classifiers in Table 6.2, in which it can be seen that the BP classifier responds to changes in skewness with a stability comparable to that of the PD classifiers.

6.3 Covaried Bimodal Class Distributions Results

As there is no single hyper-plane which can linearly divide any two classes for the bimodal and spiral data, MICD is no longer really useful as a classifier, and the comparison between PD and BP classifiers becomes much more important.

The results for the bimodal class distributions are shown across all distributions for N=1000 in Figure 6.4, and are summarized for all bimodal class distributions studied in Table 6.3.

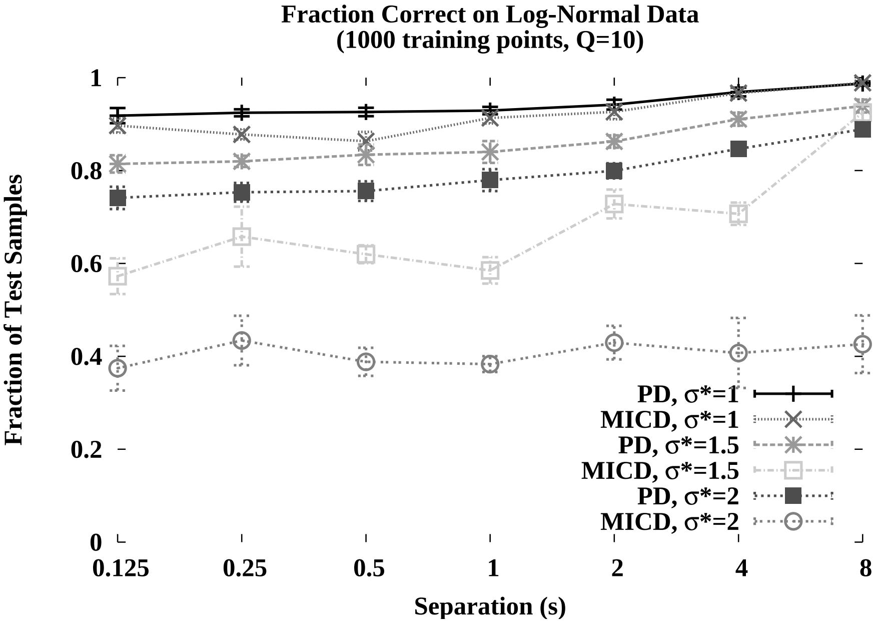


Figure 6.3: Log-Normal Data N=1000 Results

The PD classifiers with N=1000, with Q=20 and with Q=30 again showed poor performance, however with lower Q or greater N, the PD classifier performance matched that of the other classifiers. Note that the BP classifier performance suddenly decreases at high separation for these class distributions.

As seen in Table 6.3, training data error has little effect on the performance of the PD classifiers, as shown in the $\mathcal{T}_{\mathrm{err}}$=0.1 column. While this is largely true for the BP algorithm as well, the MICD classifier shows a strong sensitivity to this form of error, due to the inaccuracies this error will introduce into the estimates of the mean and covariance values.

6.4 Spiral Class Distributions Results

The results of experiments using spiral class distributions are summarized in Table 6.4.

We see in these results that the performance of the PD classifiers remain close to that of BP classifiers, and that Q=10 provides good performance in all of the cases examined. The performance of Q=20 is high when N=10, 000, however when N=1000 the performance is much lower.

Table 6.3: Weighted Performance: Bimodal Class Distribution

	Bimodal				
Classifier	N=100	N=500	N=1000	N=1000 $\mathcal{T}_{\text{err}} = 0.1$	N=10,000
PD(WOE/Indep);Q=5	0.69±0.05	0.73±0.02	0.78±0.02	0.78±0.02	0.82±0.01
PD(WOE/Indep);Q=10	0.73±0.05	0.79±0.02	0.79±0.01	0.79±0.02	0.83±0.00
PD(WOE/Indep);Q=20	0.73±0.05	0.78±0.02	0.79±0.02	0.78±0.02	0.82±0.00
PD(Occurrence/All);Q=5	0.70±0.05	0.77±0.03	0.80±0.01	0.80±0.01	0.81±0.01
PD(Occurrence/All);Q=10	0.72±0.05	0.81±0.02	0.82±0.01	0.81±0.01	0.84±0.01
PD(Occurrence/All);Q=20	0.71±0.06	0.77±0.02	0.78±0.02	0.78±0.01	0.83±0.00
BP;H=5	0.70±0.03	0.73±0.04	0.74±0.01	0.75±0.03	0.74±0.02
BP;H=10	0.73±0.03	0.82±0.02	0.82±0.01	0.81±0.03	0.82±0.01
BP;H=20	0.70±0.05	0.83±0.03	0.85±0.01	0.82±0.02	0.85±0.01
MICD	0.73±0.04	0.73±0.02	0.73±0.01	0.67±0.01	0.73±0.01

Table 6.4: Weighted Performance: Spiral Class Distribution

	Spiral				
Classifier	N=100	N=500	N=1000	N=1000 $\mathcal{T}_{\text{err}} = 0.1$	N=10,000
PD(WOE/Indep);Q=5	0.17±0.07	0.61±0.04	0.65±0.03	0.64±0.03	0.67±0.01
PD(WOE/Indep);Q=10	0.14±0.04	0.58±0.05	0.70±0.03	0.68±0.02	0.74±0.01
PD(WOE/Indep);Q=20	0.29±0.02	0.59±0.02	0.63±0.02	0.61±0.02	0.74±0.01
PD(Occurrence/All);Q=5	0.17±0.02	0.62±0.04	0.65±0.03	0.64±0.03	0.64±0.01
PD(Occurrence/All);Q=10	0.14±0.04	0.57±0.04	0.70±0.03	0.69±0.02	0.73±0.01
PD(Occurrence/All);Q=20	0.07±0.02	0.32±0.04	0.43±0.03	0.37±0.04	0.73±0.01
BP;H=5	0.60±0.06	0.71±0.03	0.73±0.02	0.67±0.05	0.71±0.01
BP;H=10	0.62±0.08	0.71±0.05	0.75±0.03	0.74±0.03	0.79±0.02
BP;H=20	0.63±0.09	0.76±0.01	0.77±0.03	0.73±0.02	0.81±0.01
MICD	0.50±0.07	0.49±0.03	0.49±0.03	0.50±0.02	0.50±0.01

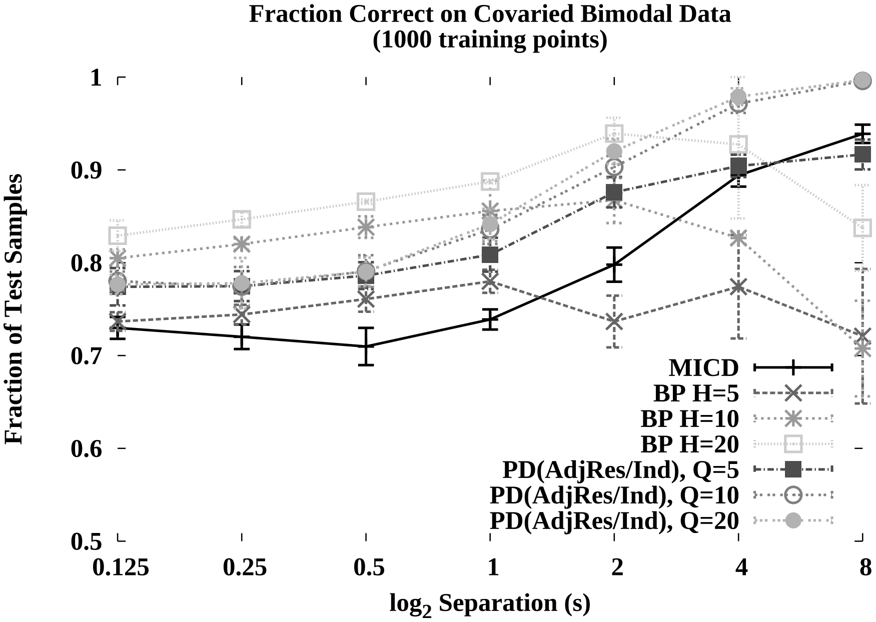


Figure 6.4: Covaried Bimodal Data N=1000 Results

The relative performance of the PD classifiers shows the PD(Occurrence/All) classifier performed more strongly than the PD(WOE/Ind) classifier, however note the extremely poor performance for Q=20, and N=100 tests of the PD(Occurrence/All) classifier in this case.

Here too, the $\mathcal{T}_{\mathrm{err}}$=0.1 column shows a stability against training data error with the PD classifiers being less affected than the BP classifiers.

6.5 Discussion

The results of these evaluations clearly show that the PD algorithm is an effective data mining tool. Furthermore, these results demonstrate that PD classifiers can be effective components within a higher level decision support system. These results show that the PD classifiers are sensitive to having sufficient training data, though their requirements do not exceed those of other popular classifiers, such as BP. Both PD and BP classifiers (Rumelhart *et al.*, 1986; Minsky and Papert, 1988; Simpson, 1991; Hertz *et al.*, 1991) are trained in a similar way; a set of training data is examined and the essential relationships describing the data are extracted. These relationships or patterns are then used in classification to provide labels for new test data. PD and BP classifiers can be applied to linearly and non-linearly separable class-distributions.

The primary difference between the two classifiers lies in the structure of the decision space. The PD classifiers construct a contingency table from discretized training values forming a hyper-cell division of the input space and make classification decisions using a nonlinear-weighted information-theory or occurrence based estimation of the most likely class calculated using the patterns occurring in the input data vector. BP classifiers make a decision by selecting several optimal hyper-planes and making a classification by performing a regression on multiple weighted hyper-planes within the subdivided space.

Both PD and BP classifiers benefit from large amounts of representative, labelled training data and have configuration parameters that affect their classification performance. When PD is applied to continuous-valued data, the number of intervals (*i.e.*, the resolution) used to quantize the data relative to the number of features and the amount of training data available is important. The number of hidden nodes and the learning rate and momentum are important factors for BP. Therefore, the performance of various configurations of these two classification schemes was compared so that some insight into the impact of these factors on the practical use of PD and BP classifiers could be obtained.

6.5.1 Covaried Class Distributions

The results for the covaried data demonstrate that when the value for Q rises, the performance may fall, as shown in Table 6.1. This behaviour is a result of the need for sufficient data to reach the expectation limit of (3.9) within the PD classifier to reliably discover high-order patterns. Without high-order pattern (maximum 5th order for this data set), performance suffers.

We also see that the performance of the PD classifiers is reasonably close to the optimal performance of an MICD classifier for these simple, linearly separable, class distributions. This implys that a PD classifier can reach optimal classification performance when the quantization adequately represents the underlying data set and sufficient N allows optimal pattern discovery (*i.e.*, all existing high order patterns are found).

With a high value for Q and a small training data set size, an insufficient number of observed events will occur for the highest order events to be confidently observed and discovered as patterns. As a result, events which may form patterns in the underlying data set are not discovered during training, in turn providing a poorer pattern space in which to make decisions during classification. This is the process responsible for the low performance for Q=20 and N=100 or 1000. Increasing the size of the training data set will overcome this as the number of occurrences of each high order event will rise; this is not a satisfactory solution in all cases however, as often sufficient training data is simply not available. In such cases, where reliable training data is difficult to produce, lowering the value of Q to produce a coarser division of the feature space may provide a more viable alternative. Considering the accuracy of the classification when Q=10 in Table 6.1 we see that choosing a lower value of Q does not necessarily penalize the performance of the PD classifier; instead, for these class distributions, the strong ability to generalize arising from the patterns discovered allows correct classification decisions to be made while discretizing the features at a lower resolution.

It is notable that PD classifiers provide correct decisions even when the separation is very low; decisions which approach the optimal MICD bound in this linearly-separable decision space.

The difference between the performance of the PD(WOE/IND) and PD(OCCURRENCE/ALL) classifiers shows that using the infinite weighting of the PD(WOE/IND) scheme may be over-weighting some of the pattern assertions, and that a more generalized "centre of mass" approach such as that produced by a more linear weighting and firing of all patterns may avoid this over-weighting problem.

6.5.2 Log-Normal Class Distributions

When examining the log-Normal data in Figure 6.3 and Tables 6.2 it is apparent that the performance of the PD classifier is independent of the distribution of the underlying random elements of the data, while the assumption made by the MICD classifier that the distribution is Normal penalizes its performance. As was mentioned when describing the log-Normal distributions in Section 5.2, as the shape parameter ξ is increased the skewness of the distributions increase. Figure 6.3 indicates that as ξ increases and the distributions become more skewed, the performance of the MICD classifier drops, until by ξ=2, the MICD classifier is essentially guessing.

In contrast, the PD classifier performance is only slightly affected as skewness increases, even though in the tails of these distributions there is now insufficient data available for PD to be able to create acceptable patterns to characterize this space. The performance of the PD classifier is very stable compared with that of the MICD classifier in this case. This demonstrates that the PD classifier performance is not strongly tied to the inherent shape of the class distribution, nor to the distribution of the noise present during measurement. In particular, assumptions that either of these distributions be Normal are not required.

6.5.3 Covaried Bimodal Class Distributions

The bimodal class distributions in Tables 6.3 and Figure 6.4 are not linearly separable, but still contain a high degree of internal structure, as the covariance of each mode matches the covariance in the unimodal case.

All the non-linear classifiers found this problem relatively easy, out-performing the MICD classifier from the outset. Notable again was the deviation in the performance of the PD classifiers as Q changes; again Q=10 was the optimal value shown because of the balancing between discretization resolution and statistically sufficient expectation. In particular the performance of Q=20 was noticeably lower as there were not enough training samples to support this level of quantization as high order patterns were not discovered. These class distributions are clearly divided, although in a non-linear way, and the class divisions follow the orthogonal orientation of the feature quantization space, so PD performance was quite similar for Q=5 and Q=10.

Here again, performance of the PD(Occurrence/All) implementation is better than that of the PD(WOE/-Ind) weighted scheme, indicating that the WOE based weightings may again be confusing the classification system into choosing an incorrect classification. As N (the size of the training set) increases, this effect becomes much less noticeable.

The apparent performance anomaly in BP as separation increases in this bimodal distribution test is due to local minima problems. Once the separation increases to the point where the inner modes can be well distinguished, the bimodal data becomes a continuous version of the XOR problem — a problem well known in the machine learning community for its need for a classifier which can function in a non-linear space. While BP can solve this problem, it is known to be “hard” for BP to find a solution, as several minima exist in the error space. When examining the individual performance numbers within the BP jackknife tests, it was found that several of the classifiers had a performance of exactly 0.5, indicating that they have failed to separate the modes of two of the classes. If these failed tests are removed, the BP performance also saturates at 1.0 along with the other classifiers tested.

6.5.4 Spiral Class Distributions

For the spiral data, the effect of N can be clearly seen. Noting the performance of the PD classifiers with Q=20 when N=1000, it can be seen that the PD classifiers do not have a large enough training set to characterize this complex data. Once a training set of sufficient size is available, or if the quantization value is kept reasonably low, the PD classifier performance rises to rival that of the BP classifier, which performs admirably in this case.

Table 6.4, when N=1000, demonstrates the strengths and weaknesses of the PD classifier:

- as the shape of the underlying distributions is curved, the resolution of the discretization bins is desired to be high, thus Q=5 has poor performance;

- the number of training examples is not sufficient to support 20 quantization intervals so performance is very low for Q=20;
- using a Q value high enough to capture as much resolution in the data as possible without defeating the threshold placed on expectation (*i.e.*, Q=10), a performance comparable to BP can be achieved.

At high Q and low N, the PD(Occurrence/All) classifier performs abysmally, as there are not enough training examples to discover any patterns when separation is low. This leads to a large number of unassigned values, and biases the performance statistic to low values.

When examining the results for each of the jackknife sets, it was found that at separation 0.125, no rules were produced at all for the N=100 case, only 2 rules were produced at separation 0.5, and the largest number of rules produced was 8, all of order 2. In contrast, at N=1000 up to 92 rules were produced to characterize this data.

The low number of rules produced indicates that only a small number of the hyper-cells are covered; in the remaining hyper-cells, no classifications will be performed. In cases where classification does occur, it is performed based on few and conflicting rules.

The PD(Occurrence/All) classifier is particularly affected in this case as the few rules established are of low order and exhibit a high degree of conflict. This leads to nullification of the accumulated weights and cause incorrect assignments formed on small amounts of poor information.

6.5.5 Overall Performance Analysis

For a given value of Q, as N is increased, the relative performance of the PD classifiers improves, relative to the BP classifiers. This performance improvement is a function of the ability of the PD algorithm to discover higher-order pattern with the requisite statistical confidence (*i.e.*, $e_{\mathbf{x}_l^m} \geq 5$). The discovery of these higher order patterns allows more accurate classification decisions to be made.

The interplay between N and Q is such that if Q is too low, increasing N will have little effect. Conversely, if Q is to be set to a high value, a large N will be required before any high order patterns are available. For the continuous-valued class distributions studied, it seems that Q=10 provides a reasonable compromise between sufficient quantization resolution and the ability to discern high order patterns without the need for training sets which are unlikely to be available in practice. Correspondingly, data sets of size no larger than N=1000 are sufficient to support characterization through PD patterns and achieve high performance during classification.

When examining the effect of differing data distributions, it was shown that the PD algorithm is largely insensitive to variations in data topology, and is not reliant on assumptions such as Normality. This is expected, as like BP, the underlying concepts of the PD algorithm avoid any specific assumptions about data distribution topology. The only assumption made by the PD algorithm is the null-hypothesis that unrelated events are independent and uniformly distributed.

The minimal effect of the training error on the performance of the PD classifiers is due to the statistical rigour used in defining a pattern. This allows erroneously labelled training examples to be ignored, providing they do not occur a statically valid (and therefore unlikely) number of times. Training errors therefore will not affect the patterns discovered, nor subsequent classifications made.

It is clear that the performance of the PD classifier itself is improved with the use of Occurrence weighting and the firing of all the patterns in many cases. This implies that the WOE weighted, "independent" pattern selection strategy has some weakness which is compensated for by Occurrence weighting and firing of all patterns. When examining the underlying data records which differ between the two algorithms, it was found that the WOE statistic may be over-weighting the patterns found.

Examining the differences in performance between PD(WOE/Ind) and PD(Occurrence/All) in Tables 6.1 and 6.3 it is clear that the performance differences between the algorithms decrease as N increases. This suggests that the under-performance of the WOE algorithm is related to low N. In turn, this implies that WOE weighting is over-weighting patterns made on low numbers of training events, which in turn contributes to erroneous classifications. This may be caused by the variability in the estimate of WOE. The patterns themselves are correct, as shown by the higher performance obtained by using the same patterns with different weights; it is the weighting values themselves which are flawed.

The PD classifiers performed well for all of the class distributions studied, even though in general, they are disadvantaged by the fact that the orthogonality and interval distribution of their discretization space is created without regard to class boundaries. The decision surfaces of the MICD classifiers can therefore out-perform those of PD classifiers by creating an optimal hyper-plane when the data is linearly separable. The BP classifier can similarly create non-orthogonal planes to represent the class distributions, in both linearly separable data and in the general cases. In this regard, improved PD classifier performance could be expected if class-dependent discretization schemes were used.

6.6 Conclusion

The PD and BP classifiers performed comparably on the selection of class distributions studied; using simple linearly separable class distributions there is little difference and when examining non-linearly separable class distributions the PD classifier performance is at worst comparable.

Both the PD and BP classifier require the setting of control parameters in order to function optimally:

- for the PD classifier, the main constraints are to have sufficient quantization resolution to adequately represent the important aspects of the class distributions while maintaining the expectation bound of (3.9); the general desire is to increase the number of quantization intervals until the expected number of occurrences in a hyper-cell is below statistical reason. Statistical uncertainty can be easily avoided with only a cursory examination of the dimensionality of the class distributions and knowledge of the number of training examples available.
- for the BP classifier, the number of hidden nodes, learning rate and momentum must be chosen. Suitable values for each of these can be dependent on class distribution, making selection difficult without extensive knowledge of the data or significant experimentation.

The original authors of the PD algorithm (Wang, 1997; Wong and Wang, 1997, 2003; Wang and Wong, 2003) have evaluated its performance on a variety of ordinal and discrete-valued data problems. The results presented here suggest that the PD classifier, using MME quantization, can be successfully used for analysis of continuous- or mixed-value data as well.

The current results demonstrate that the performance of the PD algorithm used as a classifier and applied to continuous-valued data is comparable to the well-accepted MICD and BP classifiers across a number of class distributions and shows that the PD classifier performance is robust.

The amount of training data available places constraints on the extent to which input data can be quantized and can limit the performance of PD classifiers. When compared with BP classifiers, it is evident that the cost of this quantization is not over-large. The ability to confidently configure PD classifiers, their strong absolute and relative performance when applied to continuous- or mixed-valued data and the transparency and strong statistical basis for the patterns discovered should allow PD classifiers to be successfully applied to classification problems where the rationale and confidence of the underlying decision is important. The combination of the data mining abilities of the PD algorithm and the transparency of PD based classifications provide the framework in which the context required for effective decision support and analysis is provided.

6.7 Summary

While the PD algorithm performance may be exceeded by natively continuous classifiers such as BP and MICD, the difference is not very great.

The PD algorithm is stable across distributions; in particular the log-Normal distribution performance of PD is quite good while the Bayesian MICD algorithm has significant problems as skewness increases. There is also stability of the PD algorithm across training error. As these tests have been proven successful on the base PD algorithm, we will not need to repeat them for the derived FIS.

The performance of PD is also stable with respect to *N*, within the bounds set by the occurrence estimation calculation of (3.4).

Of the results in this chapter, one of the most significant is that within the PD system, the implementation of the new Occurrence based weighting has higher performance (in terms of the number of correct decisions made) than the WOE based standard PD algorithm. A further important result is that while the performance of the PD system is lower than that of BP, the performance is significantly high across a variety of data type distributions.

Chapter 7

Synthetic Data Analysis of Fuzzy Inference System

There's no sense in being precise when you don't even know what you're talking about.

— John von Neumann

In this chapter the results of experimental evaluations of the performance of the fuzzified PD classification algorithm are provided. These results are organized by the type of class distribution used for the experiments. After the results are presented, a full analysis is given in Section 7.4, Discussion. These evaluations will provide a measure of the function of the PD/FIS performance while functioning as a classifier. The measurement of this performance will allow a discussion of extent to which the label values suggested by the system are correct; once the system has been measured in this way, the system's knowledge of its own confidence can be discussed. This discussion of confidence is placed in Chapter 9.

7.1 Covaried Data Results

Figure 7.1 shows results based on the covaried data tests trained with N=1000 records per jackknife training set. All experiments are summarized in Table 7.1 using the weighted performance statistic of (5.12). The data in Table 7.1 are mean values, with corresponding standard deviations. As in the last chapter, the graph in Figure 7.1 is clipped at 0.5 to allow the lines to be more clearly visible, and the x-axis is plotted in $\log_2$ for clear separation of the data points.

Table 7.1: Covaried Class Distribution Summary

	Covaried Class Distribution			
Classifier	**N=100**	**N=500**	**N=1000**	**N=10000**
FIS(Mamdani/All,crisp;Q=10)	0.49±0.07	0.60±0.03	0.64±0.02	0.68±0.01
FIS(Mamdani/All;Q=10)	0.54±0.08	0.64±0.03	0.66±0.02	0.70±0.01
FIS(Occurrence/All,crisp;Q=10)	0.47±0.07	0.63±0.03	0.66±0.02	0.69±0.01
FIS(Occurrence/All;Q=10)	0.53±0.08	0.65±0.03	0.67±0.02	0.70±0.01
FIS(Occurrence/Ind;Q=10)	0.53±0.08	0.61±0.03	0.62±0.03	0.70±0.00
FIS(WOE/All;Q=10)	0.54±0.07	0.64±0.03	0.67±0.02	0.70±0.01
FIS(WOE/Ind;Q=10)	0.54±0.08	0.63±0.03	0.65±0.02	0.70±0.01
PD(WOE/Ind);Q=10	0.52±0.07	0.64±0.03	0.63±0.02	0.69±0.01
PD(Occurrence/All);Q=10	0.51±0.07	0.64±0.03	0.66±0.02	0.69±0.01
BP;H=5	0.56±0.04	0.63±0.02	0.61±0.03	0.59±0.01
BP;H=10	0.60±0.04	0.69±0.02	0.71±0.02	0.70±0.01
BP;H=20	0.56±0.07	0.69±0.01	0.72±0.02	0.72±0.01
MICD	0.74±0.07	0.74±0.03	0.74±0.02	0.74±0.01

The various fuzzy implementations are displayed as "FIS(*weighting-scheme*/*rule-firing-method*)," where "Ind" rule firing indicates the PD based independent plan, and "All" indicates the standard fuzzy rule firing.

In both Figure 7.1 and Table 7.1, it is clear that the FIS using occurrence based weights and using all rules has the highest performance of any PD based system. This performance is exceeded by both the BP system and the (optimal) MICD classifier.

Figure 7.1 shows that the performance of all the classifiers is well-behaved with separation: the expected monotonic performance increase with separation is visible, and it is clear that the choice of weighting algorithm has only a subtle effect, as the lines for all candidate PD based classifiers form a tight spectrum across the graph. Indeed, excepting MICD and BP it is difficult to determine the identity of any other line on the graph.

Referring therefore to the numeric data in Table 7.1, we see that the BP classifiers show a higher performance than the PD based systems across all separations, however BP results in all cases are lower than that of the MICD classifier. When N=10,000, the BP classifier with H=10 hidden nodes has a correct classification rate of 0.70 ± 0.01; this is equalled by the PD based FIS. When N is low, the PD based FIS performance is almost equal to that of the BP classifier.

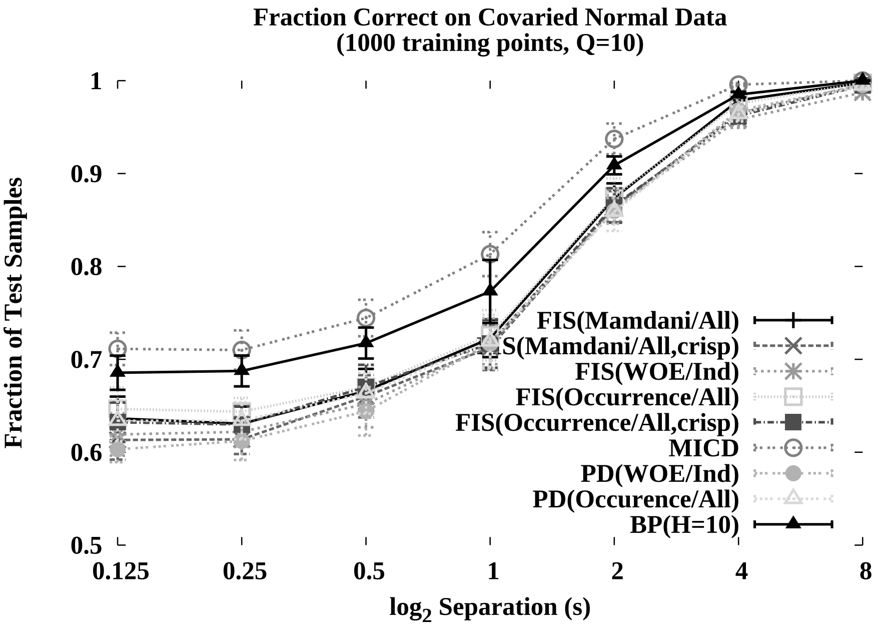


Figure 7.1: Covaried Class Distribution Results

7.2 Bimodal Data Results

Bimodal class distribution results for N=1000 are plotted in Figure 7.2, and a complete summary is provided in Table 7.2. The MICD classifier is no longer optimal, as mentioned in the last chapter, as the data is not linearly separable. The performance for this classifier is presented for completeness and comparison purposes.

The BP classifiers again have superior performance relative to the PD-based classifiers, but as seen in the results, the performance of the FIS(Occurrence/All) classifier approaches the BP classifier results while still under-performing at all separations.

7.3 Spiral Data Results

Spiral results for N=1000 are shown in Figure 7.3, and complete results are presented in Table 7.3.

As a single hyper plane drawn across the data will simply cleave each set in half, the MICD performance here is 0.5. Again, the BP classifier performance is the highest of all tested classifiers. The

Table 7.2: Bimodal Class Distribution Summary

	Bimodal Class Distribution			
Classifier	N**=100**	N**=500**	N**=1000**	N**=10000**
FIS(Mamdani/All,crisp;Q=10)	0.71±0.04	0.79±0.03	0.81±0.02	0.83±0.01
FIS(Mamdani/All;Q=10)	0.74±0.04	0.82±0.02	0.83±0.02	0.54±0.19
FIS(Occurrence/All,crisp;Q=10)	0.71±0.05	0.80±0.03	0.82±0.02	0.84±0.00
FIS(Occurrence/All;Q=10)	0.74±0.05	0.82±0.03	0.84±0.02	0.85±0.01
FIS(Occurrence/Ind;Q=10)	0.74±0.05	0.79±0.02	0.79±0.02	0.52±0.18
FIS(WOE/All;Q=10)	0.72±0.05	0.80±0.02	0.82±0.02	0.84±0.01
FIS(WOE/Ind;Q=10)	0.72±0.05	0.79±0.02	0.80±0.02	0.84±0.01
PD(WOE/Ind);Q=10	0.73±0.05	0.79±0.02	0.79±0.01	0.83±0.00
PD(Occurrence/All);Q=10	0.72±0.05	0.81±0.02	0.82±0.01	0.84±0.01
BP;H=5	0.70±0.03	0.73±0.04	0.74±0.01	0.74±0.02
BP;H=10	0.73±0.03	0.82±0.02	0.82±0.01	0.82±0.01
BP;H=20	0.70±0.05	0.83±0.03	0.85±0.01	0.85±0.01
MICD	0.73±0.04	0.73±0.02	0.73±0.01	0.73±0.01

Table 7.3: 4-feature Spiral Class Distribution Summary

	Spiral Class Distribution			
Classifier	N**=100**	N**=500**	N**=1000**	N**=10000**
FIS(Mamdani/All,crisp;Q=10)	0.29±0.07	0.58±0.04	0.69±0.03	0.72±0.01
FIS(Mamdani/All;Q=10)	0.36±0.07	0.64±0.04	0.71±0.03	0.74±0.01
FIS(Occurrence/All,crisp;Q=10)	0.29±0.07	0.58±0.04	0.70±0.03	0.72±0.01
FIS(Occurrence/All;Q=10)	0.35±0.08	0.64±0.04	0.71±0.03	0.76±0.01
FIS(Occurrence/Ind;Q=10)	0.35±0.08	0.64±0.04	0.69±0.03	0.70±0.01
FIS(WOE/All;Q=10)	0.35±0.08	0.64±0.04	0.71±0.03	0.73±0.01
FIS(WOE/Ind;Q=10)	0.35±0.08	0.64±0.05	0.70±0.03	0.72±0.01
PD(WOE/Ind);Q=10	0.14±0.04	0.58±0.05	0.70±0.03	0.74±0.01
PD(Occurrence/All);Q=10	0.14±0.04	0.57±0.04	0.70±0.03	0.73±0.01
BP;H=5	0.60±0.06	0.71±0.03	0.73±0.02	0.71±0.01
BP;H=10	0.62±0.08	0.71±0.05	0.75±0.03	0.79±0.02
BP;H=20	0.63±0.09	0.76±0.01	0.77±0.03	0.81±0.01
MICD	0.50±0.07	0.49±0.03	0.49±0.03	0.50±0.01

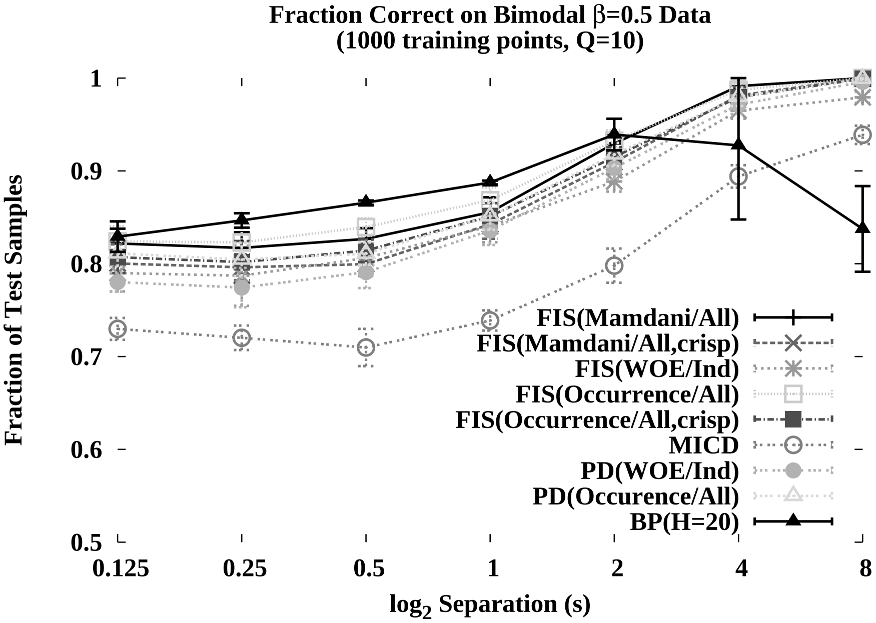


Figure 7.2: Bimodal Class Distribution Results

unmodified PD classifier out performs some of the FIS classifiers when N=10, 000, however the FIS(Occurrence/All) classifier remains the top overall performing PD based classifier.

Table 7.4 indicates the number of records left unclassified by the three PD based classifiers studied. Similar data is not presented for the other class distributions because for these distributions no records were left unclassified. It is clear from Table 7.4 that the FIS with fuzzy support (*i.e.*, the trapezoidal ramps) was able to classify records which were left unclassified by the crisp methods.

7.4 Discussion

The results shown here suggest that the rules produced by PD induce correct, highly confident classifications when implemented by the FIS algorithm. In addition, the use of these rules under a number of input and output weighting schemes provides performance which meets or exceeds the performance of a crisp PD classifier. We see here the amelioration of the effects of quantization through the use of fuzzy input membership functions.

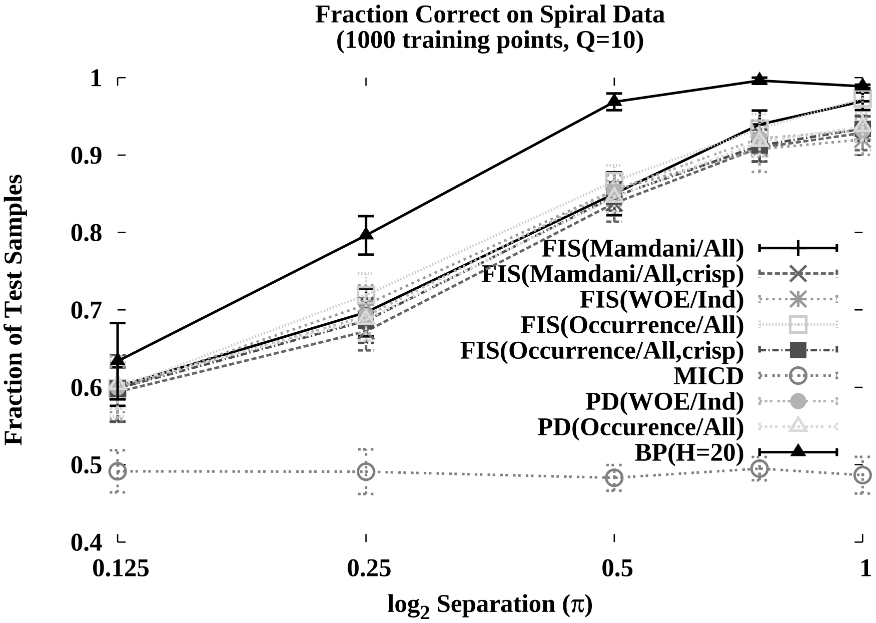


Figure 7.3: 4-feature Spiral Class Distribution Results

PD based rule generation techniques are similar to techniques that use statistical clustering for input space segmentation and contingency tables for generating rule weights (Kukolj, 2002; Chen *et al.*, 2001; Chen, 2002). However, the PD techniques use adjusted residual analysis and, as implemented here, MME quantization. These techniques both differentiate this work and provides a transparency of a type not present in the other techniques. MME quantization is class independent and while class dependent quantization can be applied within the PD framework (Ching, Wong and Chan, 1995) to possibly improve performance, MME quantization may allow a more direct interpretation of the bins created as knowledge of specific class characteristics is not required to interpret the quantization intervals used to define rules and hence contribute to a more transparent classifier.

The adjusted residual analysis provides statistically valid tests to select patterns that are then known to express valid relationships between specific feature values and specific class memberships and are thus selected as rules for classification. Using contingency tables to solely provide rule weightings while using all events as rules for classification may cause confusion due to the occurrence of conflicting statistically insignificant events. Furthermore, adjusted residual analysis allows both positive and negative classifica-

Table 7.4: Spiral Class Distribution Unclassified Records

Classifier	**0.125**	**0.250**	**0.500**	**0.750**	**1.000**
FZ (ramps)	*none*	*none*	*none*	*none*	*none*
FZ (crisp)	0.013±0.0049	0.003±0.0031	0.001±0.0029	*none*	*none*
PD	0.013±0.0049	0.004±0.0032	0.001±0.0029	*none*	*none*

tion assertions to be made that can take advantage of information both supporting and refuting specific classifications.

7.4.1 Performance Across Class Distributions

The performance of the PD-based fuzzy classifiers studied was robust across three very dissimilar synthetic data distributions, indicating that the rules produced by the PD algorithm are robust, and that the adaptation of these rules to provide a fuzzy logic framework maintains the power and discriminative properties of the PD rules, while at the same time overcoming some of the cost incurred by quantization of the input data.

7.4.2 Crisp versus Fuzzy

Comparing the results of the FIS(Occurrence/All) experiments with those using FIS(Occurrence/All,Crisp) the improvement in the FIS produced by establishing the fuzziness of the input membership functions is obvious. Specifically, modelling quantization vagueness contributed to the performance increases represented in Tables 7.1, 7.2 and 7.3. The fuzzification of the bin boundaries allows the FIS with fuzzy bins to make correct decisions through the use of additional information.

The reduction in the number of errors occurs through the firing and use of more (and possibly better) rules near inter-class feature value boundaries. These rules were created for use in the adjacent quantization bins. The validity of their assertions extends into neighbouring bins with a high degree of observed accuracy, improving the overall performance.

7.4.3 Performance Across Weighting Schemes

It is also apparent from Tables 7.1, 7.2 and 7.3 that the use of occurrence based weighting is superior to the WOE mechanism when used within an FIS; this is unsurprising as evaluation of the “occurrence” based rule weighting provides superior performance within PD itself, as seen in the last chapter. In all the result tables it is clear that the difference in performance among the various PD algorithms is small. This bears witness to the admirable strength of the PD-produced rules, supporting the use of several weighting schemes with generally strong performance.

Comparing the results of the FIS(OCCURRENCE/ALL) and FIS(WOE/IND) weighting schemes, it appears that performance when using FIS(WOE/IND) weighting is subject to a penalty due to an over-weighting of the fuzzy rules. This is consistent with the evaluation of the PD system itself where PD(OCCURRENCE/ALL) has performance measures which out-perform those of PD(WOE/IND).

As seen in Tables 7.1, 7.2 and 7.3, the performance improvement of FIS(OCCURRENCE/ALL) over FIS(WOE/IND) is related to the improvement seen in the last chapter of PD(OCCURRENCE/ALL) over PD(WOE/-IND) in Tables 6.1, 6.3, and 6.4.

7.4.4 Performance Based on Training Set Size

As *N* is increased, the performance of all of the FIS and PD classifiers increase. At low *N*, the PD based systems show a stable performance and are still able to extract salient rules from complex data. Examining Table 7.3, it is apparent from the performance of the PD versus FIS classifiers that the implementation of fuzzy membership has allowed the few rules created with extremely low *N* to be used more often, increasing performance at N=100 from 0.14 to 0.35. This performance improvement is possible because of the strength of the discovered rules: even at such low *N*, the rules discovered are trustworthy.

7.4.5 Cost of Quantization

The use of a discrete algorithm in a continuous domain often incurs a cost of reduced performance due to the quantization performed. One of the features of fuzzification of quantized data is to reduce the cost of quantization and improve the performance of discrete algorithms towards that which is exhibited by well considered natively continuous algorithms. The data in Table 7.4 suggests that fuzzification of the input space models some of the vagueness associated with quantization of continuous-valued input data, allowing records which are left unclassified by the crisp classifiers to be classified by the FIS using rules which are obviously still valid.

In the spiral class distribution problem, adjacent bins frequently support differing labels simply due to the topology of the class distributions. Table 7.4 is therefore particularly interesting as it shows that the ramps extend the "region of effect" of a given rule into parts of the data space which are difficult to characterize based on MME quantization, but which still can be covered successfully using the ramped decreasing membership of rules in adjacent bins. The use of rules from adjacent bins provides good performance, even though the spiral class distribution would suggest that this is difficult.

As seen in all the above results, the FIS(Occurrence/All) based classifier exceeds the performance of all other PD based systems. Table 7.1 shows a marked improvement by the FIS classifiers over the straightforward PD classifier. Much of this improvement is due to the presence of the trapezoidal membership function ramps (*i.e.*, extended, or fuzzy support) as indicated by the lower performance of the crisp input membership function version of the FIS.

7.5 Conclusions

The PD algorithm has various weaknesses, some of which can be mitigated by applying the fuzzy set techniques described in this work The improvements resulting from the fuzzification of PD are well demonstrated using the spiral class distribution shown in Figure 7.3 and Table 7.3:

- the curving shape of the underlying distribution poses a problem with respect to the rectangularly divided quantization space;
- this in turn causes a great deal of conflict in some of the quantized hyper-cells, generating no rules for these regions of discord;
- the extended fuzzy support of the input membership bins allows the extension of rules from adjacent hyper-cells into these conflicting regions, allowing classification assertions to be made based on neighbouring cells, albeit with decreasing performance as the distance away from the cell bound increases.

The improved performance obtained through the use of the fuzzified input membership functions occurs through the firing and use of more (and possibly better) rules near inter-class feature value boundaries. These rules were created for use in the adjacent quantization bins, however the validity of their assertions extends into neighbouring bins with a high degree of observed accuracy, improving the overall performance. Together, these factors allow the use of fuzzified PD based classifiers even in strongly conflicting regions of the problem space. Their performance is acceptable in spite of the regions of discord, and they can be used with relatively low N.

While the BP classifier out performs the FIS, the BP classifier itself suffers from several drawbacks, notably the uncertain tuning of configuration parameters such as the number of hidden nodes, learning rate, *etc*. Perhaps the most significant problem is the "black box" functionality of the neural net as a whole, as described in the background discussion provided in Chapter 2. In contrast the FIS can be easily configured based on the PD expectation of (3.4) as seen in the last chapter, and provides the transparency we will require for a DSS.

7.6 Summary

The contributions explored in this chapter are:

- performance of the FIS in relation to PD;
- performance benefits of fuzzy membership functions; and
- performance benefits of the new occurrence weighting scheme.

The performance of the new FIS is better than PD performance. The use of fuzzy membership functions recovers some of the cost of quantization and improves the overall system performance. In particular, the FIS(Occurrence/All) classifier has the highest performance of any PD based classifier.

The FIS and continuous-adapted PD systems have been evaluated on a variety of synthetic data distributions. The FIS performance still does not match the performance of BP, however the classifiers are comparable.

Even though the FIS performance is not the best of the classifiers measured, the performance is reasonably high. A decision support system needs all of: a classifier to produce a suggested label; a good estimate of the confidence of the suggestion *and* exceptional transparency. For the construction of a DSS therefore, a classifier which performs well and exhibits a high degree of transparency is preferred to one which may have a higher classification performance but lower transparency. This implies that the PD based FIS system meets our needs admirably for DSS design purposes, as the performance is comparable to the BP system and the transparency of the inference is much higher.

What remains is to test the system upon real-world examples, and to describe the system in a decision support context.

Part III

Real-World Data and Decision Support

Chapter 8

Analysis of UCI Data

Many data sets are available in the Machine-Learning Repository (Newman *et al.*, 1998) at the University of California, Irvine (UCI) which provides a collection of standard data bases for use in evaluating the performance of various classification systems, and is now recognized as the source of “standard” data for classification performance analysis. Two of the UCI data sets have been chosen for use here as they have continuous-valued attributes: the “thyroid disease” and “heart disease” databases.

8.1 Analysis of Thyroid Disease Data

The human thyroid disease database available at has been chosen as an example both because this data falls within the biomedical domain as well as because the data is multi-class and includes a fair number of training records. Most of the data sets in the UCI repository are provided with only a very small number of records, this is frequently the case when each record is costly to obtain. The 7200 records in the thyroid set therefore represents a reasonably large amount of data, considering it has been collected from a real source. This data set is referred to as `thyroid-disease` within the UCI repository, and is termed “the ANN[*] data set,’ as supplied by Randolf Werner of Daimler-Benz. The access URL for this database is `ftp://ftp.ics.uci.edu/pub/machine-learning-databases/thyroid-disease`.

8.1.1 UCI Thyroid Data Preparation

The relevant attributes of this database are:

[*] *i.e.*, ‘artificial neural network”.

Table 8.1: UCI Thyroid Data Record Counts

Class:	*0*	*1*	*2*
Data Set 1	33	74	1,333
" 2	33	74	1,333
" 3	33	74	1,333
" 4	33	74	1,333
" 5	34	72	1,334
Total	166	368	6,666

- the data set is fully continuous;
- there are a reasonable number of records in the data set, having $3,428$ designated for "training" and $3,772$ for "testing" as presented in the repository for a total of 7200; and
- the data set represents real disease states recorded from human patients.

The one drawback of this data set is its rather poor documentation, as the features are not named. It is impossible, therefore, to get a good sense of how this data set relates to actual disease data.

For this work, a number of jackknife sets were produced using the following procedure:

- the "testing" data was concatenated onto the end of the "training" data;
- records for each separate class $\{0, 1, 2\}$ were isolated into separate files while preserving the order to allow later comparison by other readers;
- each class was divided into 5 sets, and finally
- the sets of each class were combined to form 5 jackknife data sets, grouping the first of each of the 5 sets together to form the first jackknife set, then using the second of each, *etc*.

This divides the $7,200$ available records among five files with the record counts by class as shown in Table 8.1.

8.1.2 Results

Using only the continuous-valued attributes from the data described in 8.1.1 (and thus using records with 6 input values plus a label for training), the following familiar classifiers were evaluated:

- a BP network;
- the MICD classifier;
- the base PD algorithm;

Table 8.2: 6 Feature Thyroid Data Correct Classification

Classifier	*Fraction Correct*
FIS(OCCURRENCE/ALL)	0.962±0.005
FIS(WOE/IND)	0.929±0.016
PD;Q=5	0.864±0.035
MICD	0.586±0.067
BP;H=4	0.960±0.012

- the FIS(OCCURRENCE/ALL) and FIS(WOE/IND) classifiers of Chapter 4.

For the FIS classifiers, Q=5 was used due to the very low numbers of records, especially in class "0."

The results for these experiments is shown in Table 8.2. This table displays the average count of the number of records classified correctly and of those with error, as well as the variance in the count for both correct and error. None of the algorithms left any of the records unclassified.

Interestingly, the FIS(OCCURRENCE/ALL) hybrid system outperforms the BP algorithm, though only by a very slight margin. The BP classifier in turn out-performs both PD(WOE/IND) and the unmodified PD classifier. The MICD performance is very interesting, as it indicates that the data is certainly not a unimodal Normal distribution.

8.1.3 Discussion

The relatively high performance of the FIS(OCCURRENCE/ALL) algorithm in the face of small numbers of training records is a strong feature, especially when compared with the performance of the BP system.

The difference in performance between the various FIS algorithms is similar to that when tested on synthetic data. The separation in classification performance of BP and FIS(OCCURRENCE/ALL) has vanished, and the relative ranking has reversed, though the difference in performance is negligible. This seemingly surprising under-performance of the BP system is most likely due to the generally small number of records available, and specifically the very small number of incidences of class "0." The asymmetric distribution of the labels will cause the training of BP to favour the most-observed label to some degree.

The analysis of thyroid data has therefore confirmed the relative ranking of the PD/FIS and PD classifiers, and has demonstrated the well-known weakness of BP when used with very small data sets. This indicates that the PD/FIS, specifically the FIS(OCCURRENCE/ALL) form, may be somewhat less susceptible to problems of low *N* than BP; an algorithm known to need a great deal of training data. This encouraging result indicates that the analysis of synthetic data may have played into strengths of the BP algorithm, due to the regular structure of the data. The PD/FIS system obviously extracts a comparable amount of information from the thyroid data as does the BP classifier; further, the performance across the jackknife sets is quite stable.

This result supports the previous conclusions that the FIS(OCCURRENCE/ALL) PD/FIS system performs admirably well when functioning as a classifier. The labels suggested by the system are of as high quality on this real-world problem as are those suggested by the BP system.

To further explore the performance of the PD/FIS system when dealing with the amount and type of data found in real-world problems, another data set was examined.

8.2 Analysis of Heart Disease Data

As a further real world example we will consider the data set of the Hungarian Heart Disease Database[†] from the online repository at UCI. These databases are available through the URL `ftp://ftp.ics.uci.edu/pub/machine-learning-databases/heart-disease`.

8.2.1 Choice of Heart Disease Database

There are four "heart disease" data sets at the UCI repository: "Cleveland," "V.A.," "Hungarian," and "Switzerland." The collection protocol between the databases varies slightly, preventing the use of the collected records as one data set.

The database with the largest number of records, "Cleveland", contains some fields which are in error, as described in the documentation accompanying the database. For this reason, it was not used. The "Hungarian" database has almost the same number of records (294 instead of 303), and has no history of errors or corruption; the "Hungarian" data set is therefore used for the analysis presented here.

8.2.2 Heart Disease Database Features

The heart disease data set has 13 numeric input features which are commonly used, which are outlined in Table 8.3. These 13 features provide a sufficiently high order to allow the PD system to discover

[†] This data was collected through the efforts of Andras Janosi, M.D. of the Hungarian Institute of Cardiology in Budapest. Dr. Janosi donated this data to the UCI Machine Learning Online Repository (Newman *et al.*, 1998) through David W. Aha in July 1988.

Table 8.3: UCI “Hungarian Heart Disease” Database Feature Names

Feature	*Type*[a]	*Description*[b]
AGE	C	age in years
SEX	N	sex [1 = male, 0 = female]
CP	N	chest pain type [1 = typical angina; 2 = atypical angina; 3 = non-anginal pain; 4 = asymptomatic]
TRESTBPS	C	resting blood pressure (in mm Hg on admission to the hospital)
CHOL	C	serum cholesterol in mg/dl
FBS	N	fasting blood sugar > 120 mg/dl? [1 = true; 0 = false]
RESTECG	N	resting electrocardiographic results [0 = normal; 1 = having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV); 2 = showing probable or definite left ventricular hypertrophy by Estes’ criteria]
THALACH	C	maximum heart rate achieved (bps)
EXANG	N	exercise induced angina [1 = yes; 0 = no]
OLDPEAK	C	ST depression induced by exercise relative to rest
SLOPE	N	slope of the peak exercise ST segment [1 = upsloping; 2 = flat; 3 = downsloping]
CA	I	number of major vessels (0-3) coloured by fluoroscopy
THAL[c]	N	[3 = normal; 6 = fixed defect; 7 = reversible defect]
LABEL	N	label for disease state, values are “healthy” and “diseased”; the original data characterized 3 levels of disease state, however the accompanying documentation states that all known classification tasks have combined all disease states into a single class.

[a] C=continuous, N=nominal, I=integer

[b] These features and descriptive text are reproduced from the documentation accompanying the data set.

[c] This rather cryptic field is not explained in the documentation for the database in the UCI repository.

informative rules. This data will be further used in the decision support discussion in Chapter 10 where inspection of the relationships found will allow the reader to observe interesting consequences of the rule evaluation in the context of this real-life data.

In Table 8.3, each column is described in terms of its “type.” In this mixed mode data set, some features can be treated as continuous values using MME, and some must be treated as distinct. All features labelled “continuous” are pre-processed using MME. All “nominal” or “integer” features are left unprocessed.

Table 8.4: Rule Order for Hungarian Heart Disease Data

Order	*# of Rules*
2nd	31.888±0.485
3rd	45.306±1.024
4th	38.412±1.891
5th	17.092±0.510

8.2.3 Overall Statistics from PD Analysis

In order to generate rules for this database, a “leave-one-out” protocol was established to jackknife over all 294 records using a single record as a test case and the remaining 293 records for training. The full jackknife protocol was used in order to make the most of the limited number of data records. All results in this chapter are therefore averaged over all 294 tests.

The PD algorithm was run using Q=7 on these 294 different jackknife tests. This Q value was chosen in order to ensure a reasonably high order of pattern could be discovered, as controlled by equation (3.4). Note that the Q factor only affects columns with continuous data values, as integer and nominal data values are not processed by MME. The number of elements in a non-continuous column is therefore determined solely by the number of unique values occurring for that feature and is entirely independent of Q.

8.2.4 Pattern Discovery Generated Rules

For the heart disease data, there were on average 132.7 rules generated to describe a jackknife trial, with a standard deviation of 2.554. The high number of rules indicates the complexity of the data set as observed by the PD algorithm with MME binning.

The average and standard deviation of the order of these rules is shown in Table 8.4. No rules higher than fifth order were discovered in this data set, due to the small number of training records available.

These low numbers indicate that the contingency table is quite sparsely characterized.

Most of the rules were weighted with non-infinite WOE, as an average of only 2.74 positive and 9.76 negative infinite rules were found in each test. On average, there were 65.27 rules characterizing disease per jackknife set, with a standard deviation of 1.28. The average number of rules characterizing normal patients was 67.42 with a standard deviation of 2.02.

From an information standpoint, this shows us that there is approximately equal complexity in the relations regarding diseased and normal patients. Further, the low number of infinite-weight rules indicates

Table 8.5: Heart Disease Data Correct Classification

Classifier	*Fraction Correct*
FIS(OCCURRENCE/ALL)	0.830±0.376
FIS(WOE/IND)	0.813±0.390
PD;*Q*=7	0.813±0.390
MICD	0.765±0.424
BP;*H*=2	0.639±0.480
BP;*H*=4	0.636±0.481
BP;*H*=7	0.653±0.476

that both classes are mutually conflicted. The ranked decrease in order shown in Table 8.4 indicates that the higher order rules may have been discovered if more input records had been available.

8.2.5 Analysis of Performance

Performance statistics were gathered for the same classifiers used in thyroid data testing:

- the MICD classifier;
- BP;
- unmodified PD;
- FIS(OCCURRENCE/ALL) and
- FIS(WOE/IND).

Results for correct classification performance are shown in Table 8.5.

The FIS system is the next best classifier, outperforming the MICD classifier by a significant margin, and improving on the performance of PD using MME quantization both with an increase in performance and a decrease in variability.

The BP classifier has quite low performance with high variation, indicating that this problem contains a local minima in the BP error surface.

8.2.6 Discussion

Significant improvements are again shown by the FIS(OCCURRENCE/ALL) algorithm against the PD algorithm alone, as in the tests on the thyroid and synthetic data sets.

The ranking of the other classifiers is stable across all the tests performed when the fact that BP frequently becomes trapped in local minima while evaluating this problem is taken into account.

This example shows that the FIS(Occurrence/All) PD/FIS hybrid can attack a problem with limited amounts of training data and perform quite acceptably on a hard problem. The performance of the MICD classifier shows that this problem has a distribution which is not unimodal Normal, and the BP performance indicates that the amount of information in the data is quite small.

Looking again at Table 8.4, one wonders whether even higher order rules may increase the robustness and discriminative power of the rule set, as the performance of 0.830 of the records classified correctly is still much lower than one would desire, especially in a medical characterization problem. The performance of the FIS(Occurrence/All) system would therefore likely be higher if more data were available for training.

Overall, the performance of the FIS(Occurrence/All) system is quite encouraging, as again the BP system performance has been exceeded. This PD/FIS system can therefore function as a classifier with comparable performance to other popular classifiers. This implies, as in the thyroid data, that this complex real-world data set shows strengths in FIS(Occurrence/All), or weaknesses in BP, that were not apparent in the synthetic data.

The results from the synthetic analysis are therefore representative of problems for which FIS(Occurrence/All) is not strong. This in turn indicates that the performance shown in Chapter 7 does not over-estimate the performance capabilities of the PD/FIS system; instead the analysis here shows that the synthetic data class distributions provided a difficult test for the PD system.

8.3 Conclusions from UCI Data Analysis

The hybrid PD/FIS system performs comparably to the BP algorithm on two important real-world data sets. Particularly high measured performance was observed when using the FIS(Occurrence/All) weightings and rule firings. This further indicates that the system is well-behaved as a classifier, supporting the conclusions of the synthetic data analysis of the previous chapter.

The contributions of this work are shown to consistently improve the performance of PD based classification systems and provide a consistent benefit to use of PD as a classifier on these real-world data exmples. The specific contributions of interest are:

1. improvements to performance through FIS(Occurrence/All) weighting; and
2. improvements to performance through the use of fuzzy membership.

This measured classification performance shows that a DSS based on this method will produce a correct suggestion with comparable likelihood to that of other classification systems. The use of the PD/FIS system as a classifier therefore is suggested for use in a DSS as the penalty in classification performance is outweighed by the benefits provided by the transparency of the system.

Chapter 9

Confidence and Reliability

> *Science is built up of facts, as a house is with stones. But a collection of facts is no more a science than a heap of stones is a house.*
>
> — Jules Henri Poincaré

As mentioned in the introduction, a decision support system (DSS) is useless if it does not have a mechanism of reporting an estimation of the confidence that can be placed in the suggestion provided. This requirement of a DSS is the means by which the system may fail gracefully. A confidence measure provides the means by which the user may understand when there is conflict or poor statistical support in a suggestion, and differentiate these occasions from those when the suggestion is made on confident and clear measures. Using such a confidence measure, the user is informed under what conditions an error is likely to be made.

9.1 Confidence and Reliability

A confidence measure is an estimate of the true reliability of the system. Equation 2.6 in Chapter 2 describes the measurement of reliability in terms of the measurable probability of failure:

$$\begin{aligned} C &= \mathcal{E}(\mathcal{R}) \\ \mathcal{R} &= 1 - \Pr(\text{failure}) \equiv 1 - \Pr(\text{Incorrect}). \end{aligned}$$

In the context of the hybrid system described in this work, the probability of failure is equivalent to the probability of providing the user with an incorrect suggestion.

Reliability is measured over a series of results, however, and as such is usually quantified as the average performance of a system. ROC analysis (as described in Section 2.3.1) provides a measure of the reliability of a two-outcome test or classification system.

What is desired for the hybrid system described here is an estimate of the system reliability, however a prefereable estimate is one which maps the confidence of a correct association being made based specifically on the current input values. Of the class of decisions made by the system, not all can be made with equal ease, and a good confidence estimate will be one which correlates well with the probability of successfully classifying records at different degrees of difficulty.

We will therefore use the term reliability to indicate the system wide performance, and use the term confidence to indicate the estimate of performance based on input data values.

This chapter will evaluate a set of possible confidence measures. Evaluation will consist of examining how well each measure predicts the true probability of success or failure of the system, as evaluated over the training/test data.

To choose a confidence measure, that which best predicts the true (measured) probability of a successful classification being produced of the system as calculated using the synthetic data type distributions is determined. This measure is then used to provide feedback to the user to indicate a varying expectation of uncertainty in the suggested outcome. This provides the user a context of estimated certainty for the suggestion put forward in a DSS.

9.2 Implementation of Confidence in Hybrid System

In the hybrid system described in this work, an assertion is assigned for each of the possible outcomes. As described in Chapter 4, the assertion in support or refutation of class k produced by input value $\mathbf{x}$ is termed $\mathcal{A}_k$, and this value $\mathcal{A}_k$, is bounded in the domain $[-1 \ldots 1]$. Using these $\mathcal{A}_k$ values we can correlate the distances in the assertion space with the observed errors found in training. From these data we can evaluate various possible confidence measures in order to choose the candidate that suits our purposes the best. Rather than evaluating confidence measures for all classifiers discussed, only the best-performing classifier, FIS(Occurrence/All), will be evaluated.

9.2.1 Construction of τ and δ Measures

Of the possible outcomes for any classification, one outcome will have the highest assertion value. Remembering that

$$k^\star = \operatorname*{argmax}_{k}^{K} \mathcal{A}_k$$

from equation (4.4), we can define

$$\tau = \mathcal{A}_{k^\star} \tag{9.1}$$

to indicate the value of the highest assertion made. The label value which τ supports will be the label $Y{=}y_{k^\star}$, which is the label used as the final suggested characterization or classification.

In any system where $K > 1$ (*i.e.*, any system containing more than one class), there will exist a class whose assertion value is the highest value if the τ-label is ignored (*i.e.*, the assertion with the second-highest value).

The assertion in support of this label is termed τ_2, which is defined as

$$\tau_2 = \max_{\substack{i=0\\ i\neq k^\star}}^{K} \mathcal{A}_i. \tag{9.2}$$

The conflict between the two classes associated with τ and τ_2 can be measured by comparing the conflicting support of the output assertions using

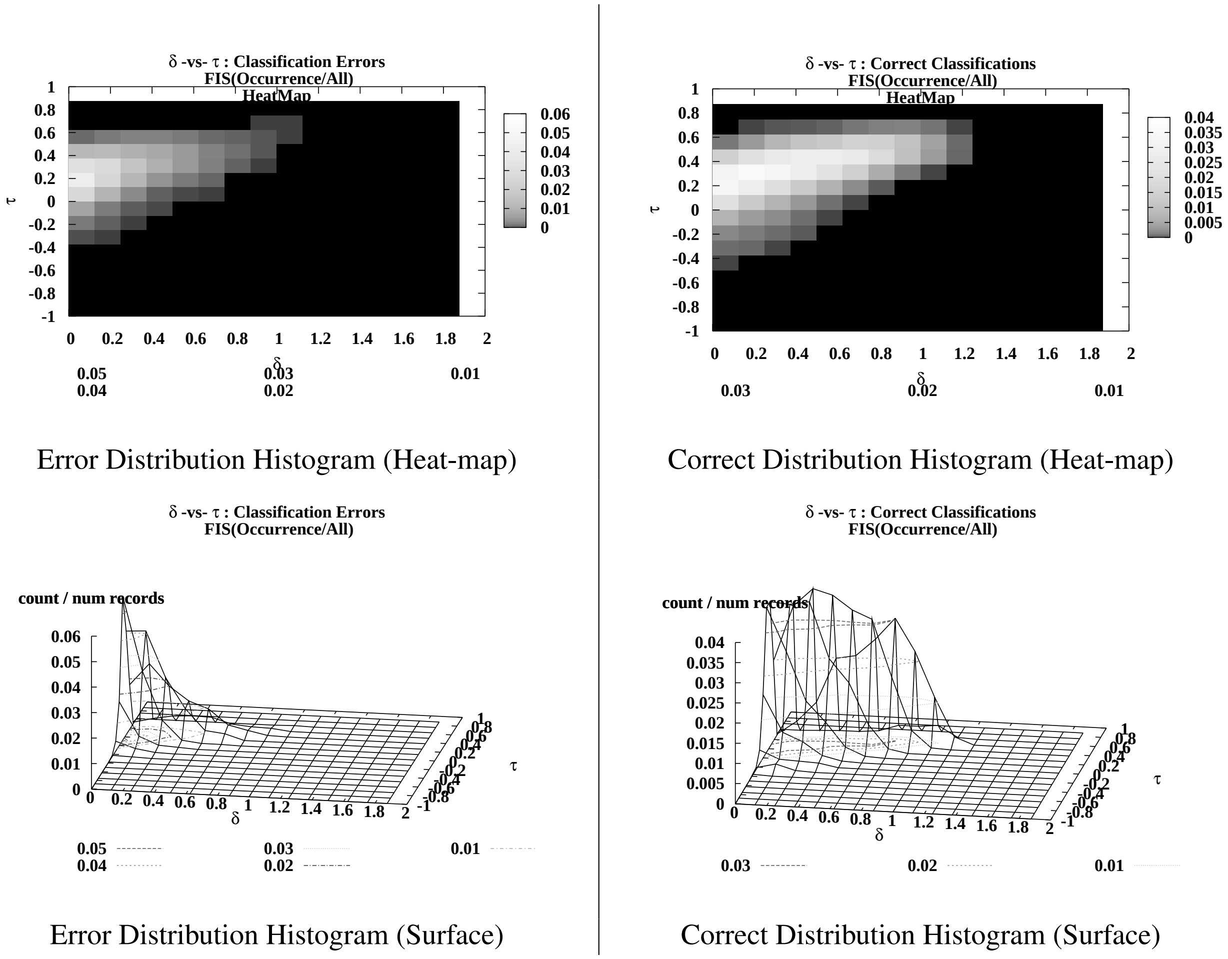
$$\delta = \tau - \tau_2. \tag{9.3}$$

In cases of equal values for τ and τ_2, the value for δ is zero. The value for δ approaches the maximum distance of 2 units in the case when there is total support for the class of the label associated with τ and total refutation for all other classes; that is, when

$$\begin{aligned} \mathcal{A}_{k^\star} &= 1 \text{ and} \\ \mathcal{A}_i &= -1 \forall i \in K, i \neq k. \end{aligned} \tag{9.4}$$

9.2.2 Evaluating Confidence Through δ and τ

Figures 9.1 and 9.2 display histograms of the δ/τ values observed for classifications performed on the covaried data at separations of 0.125 and 4.0 respectively. All classifications in these plots were performed by the FIS(Occurrence/All) classifier. In the figure, the x and y directions indicate δ and τ. Associated



Error Distribution Histogram (Heat-map)

Correct Distribution Histogram (Heat-map)

Error Distribution Histogram (Surface)

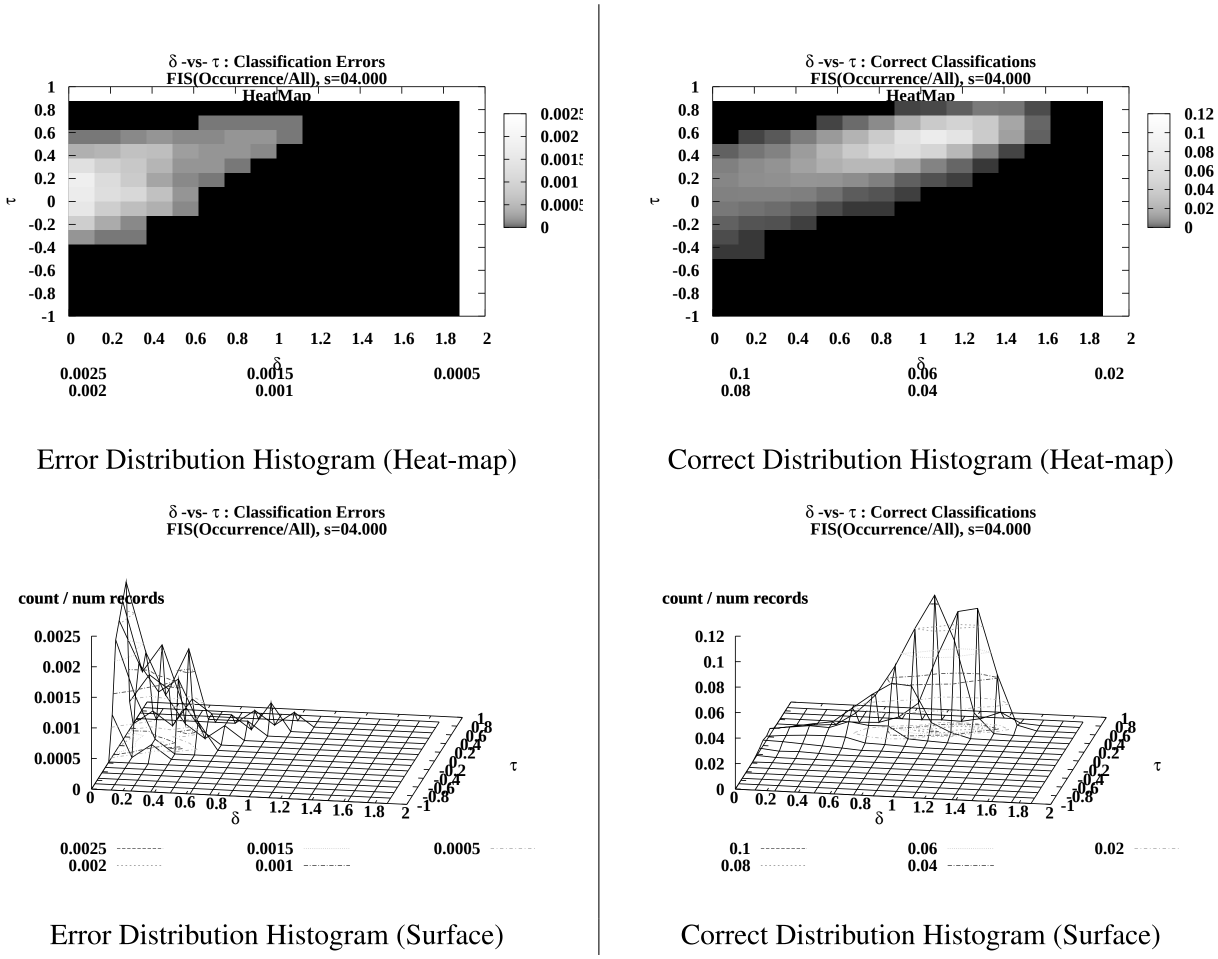
Correct Distribution Histogram (Surface)

Figure 9.1: δ/τ Histograms from Covaried Data, s=0.125

with each (δ, τ) pair is a z value that indicates the fraction of the total number of records observed that fall into the adjacent cell of the (δ, τ) domain.

The top row of each figure indicates a heat-map representation of the data, where lighter colours on the map indicate regions with higher values of z. White areas on the heat-maps indicate cells with the highest count for that map, black areas indicate values of zero. The same data is plotted below each heat-map in the form of a surface histogram, where the surface is extended in z to indicate the cell count, directly related to the colouring of the heat-map.

There is significant overlap seen in Figure 9.1 (separation $0.125s$), between the correct and incorrect



Error Distribution Histogram (Heat-map)

Correct Distribution Histogram (Heat-map)

Error Distribution Histogram (Surface)

Correct Distribution Histogram (Surface)

Figure 9.2: δ/τ Histograms from Covaried Data, s=4.0

distributions, even though the midpoints of the two classes are well separated. This is expected as the number of errors made is relatively high due to the poor discernibility of the problem. The overall expected confidence based on these decisions is low.

Considering Figure 9.1 from the standpoint of reliability, only the best-separated portions of the surfaces indicate regions where a high-reliability decision is expected; in sections with a great deal of overlap, the expected decision reliability is low.

Comparing Figure 9.1 to Figure 9.2 we see that, as expected, when separation increases the separation in the δ/τ space between errors and correct classifications increases. As well, the overall number of errors has dropped considerably. This can be seen by the much smaller maximum extent in z of the error graph in Figure 9.2 versus that in Figure 9.1.

These facts indicate that in portions of the plots where no errors are made, high reliability classifications will be observed.

9.2.3 Candidate Confidence Measures

Based on the observation that the main mass of erroneous classification in Figures 9.1 and 9.2 are in a different portion of the (δ, τ) space than corresponding masses of correct classifications, it would seem reasonable that a measure may be constructed that can discriminate decisions made in error from those made correctly, using τ and δ as inputs. If such a measure is related to the distance between means, or similarly, the relative probability of conflict in this space, that measure will serve as a robust indicator of the trueprobability of a correct classification label being suggested.

A great many different methodologies may be constructed under this hypothesis. Four candidates are considered in this work:

- MICD Based Confidence;
- δ/τ Observed Probability Confidence;
- Normalized $[0 \ldots 1]$ Bounded τ Confidence; and
- PD (WOE Based) Probabilistic Confidence.

MICD Based Confidence

This is a measure based on the distance in δ/τ space as whitened using an MICD classifier (as per equation (5.10) on page 60) and then compared against the sum of all distances. This is calculated by

$$C_{\text{MICD}} = \frac{e^{-\frac{1}{2} d_{\text{correct}}}}{e^{-\frac{1}{2} d_{\text{correct}}} + e^{-\frac{1}{2} d_{\text{incorrect}}}}. \tag{9.5}$$

The values d_{correct} and $d_{\text{incorrect}}$ are based on distances for correct and incorrect decisions, respectively. The value d_{correct} indicates the distance to the mean of the whitened distribution of δ/τ values calculated by examining decisions made correctly using training data. Similarly, $d_{\text{incorrect}}$ is the distance related to those decisions made incorrectly.

The rationale behind this confidence measure is the use of the Bayesian decision surface underlying the inter-PDF measures in the whitened MICD space. The relative distance in this space corresponds to the conditional probability of assignment under an assumption of Normally distributed “correct” and “incorrect” points. This conditional probability directly mirrors the probability of failure.

While this simple probabilistic relationship provides a direct mapping to an *a priori* conditional probability, neither the “correct” nor “error” distributions are Normal, as can be easily seen in Figures 9.1 and 9.2. The data is generally bell-shaped, but the bounds of the (δ, τ) space sharply truncate the distribution. We have seen in previous chapters that the MICD classifier still performs reasonably well with non-Normal data, degrading in a predictable fashion. For this reason, it is a useful candidate confidence measure.

δ/τ **Observed Probability Confidence**

This measure is based on a histogram-generated probability of δ/τ values

$$C_{\delta/\tau\ \text{Probability}} = \frac{\text{Count}(\text{correct}, \delta, \tau)}{\text{Count}(\text{correct}, \delta, \tau) + \text{Count}(\text{incorrect}, \delta, \tau)}, \tag{9.6}$$

where $\text{Count}(\text{correct}, \delta, \tau)$ represents the number of a correct classifications associated with the cell containing (δ, τ) on a histogram of the type shown in Figures 9.1 and 9.2. The term $\text{Count}(\text{incorrect}, \delta, \tau)$ represents the count of incorrect classifications drawn from a similar histogram.

The observation of a (δ, τ) pair during classification will let us report the likelihood of error observed for similar values at training time. This directly returns the probability of error based on a δ/τ pair. As such, this should report a good estimate of probability of successful classification, and as such an indication of system reliability as considered on records associated with these values of δ and τ.

The main drawback to this scheme is the possibility of observing a (δ, τ) value which corresponds to no known bin during the classification of new data values. It will be impossible to determine a true confidence measure for such a value, as there is no prior data upon which to base such a decision. It will, however, be possible to flag such values to the user and indicate that a confidence measure cannot be computed.

A similar constraint involves values rarely observed during training; confidence based on such values will have a much higher degree of inaccuracy than confidence based on histogram data cells which received many points during training. This implies the possibility of a “confidence of confidence” measure; this extra complexity will be at best confusing, and is certainly not desirable.

Normalized $[0\dots1]$ Bounded τ Confidence

This calculation normalizes τ over the sum of all assertions made, after shifting τ so that it is $[0\dots1]$ bounded. This confidence is calculated using

$$C_{\tau[0\dots1]} = \begin{cases} \frac{1+\tau}{\sum\limits_{i=1}^{K}(1+\mathcal{A}_i)} & \text{if } \sum_{i=1}^{K}(1+\mathcal{A}_i) > 0 \\ \\ 0 & \text{otherwise,} \end{cases} \tag{9.7}$$

remembering that τ is simply the highest-valued $\mathcal{A}_i$ as defined in equation (9.1), and that all $\mathcal{A}_i$ (including τ) will be bounded by $[-1\dots1]$ as output from the FIS.

The rationale in this measure is that a $[0\dots1]$ bounded τ-space will behave in some ways similarly to a probability space. Once this is done, the summation will contain a probability-type measure. A side effect of this measure is that it changes the way in which information is represented. In PD and in the discussion of the FIS so far, information has always been represented in terms of deviation from a central 0, where strong degrees of support or refutation provided a strong deviation. Information for each class i is therefore $\propto |\mathcal{A}_i|$, and values around 0 will therefore result from rules which fire with low confidence.

In this scheme, the 0 value of the FIS becomes $\frac{1}{2}$, similar in meaning and behaviour to a probability value; information is measured from this central limit. This means that if the reported confidence falls below $\frac{1}{2}$ then the "best" guess has actually been made without a positive τ value.

One interesting property of this measure is that it is calculated only in terms of values internal to the FIS system. The previous two measures require an external analysis of the δ/τ values from training data.

PD (WOE Based) Probabilistic Confidence

This measure is not directly based on δ/τ, but instead is based on the WOE measures associated with the fired rules.

It is possible to derive a conditional probability measure for association with a particular class $Y{=}y_k$ based on a given input vector $\mathbf{x}_l^m$, as constructed from the accumulated WOE:[*]

$$\Pr(Y = y_k|\mathbf{x}_l^{\oplus}) = \frac{1}{\left[\left(\frac{1}{\beta^{\text{WOE}}}\right)\left(\frac{[1-\Pr(Y=y_k)]}{\Pr(Y=y_k)}\right)\right] + 1} \tag{9.8}$$

[*] Construction of this equation, derived from the definition of WOE in (3.7), is courtesy of Lou Pino (2005). A copy of the derivation is provided in Appendix B, beginning on page 163.

where $\mathbf{x}_l^{\oplus}$ is the composite pattern formed by considering all columns matched by the patterns fired, that is

$$\mathbf{x}_l^{\oplus} = \bigcup \mathbf{x}_l^{\star} \qquad \mathbf{x}_l^{\star} \in \mathcal{M} \tag{9.9}$$

where $\mathbf{x}_l^{\star}$ is the input portion of a pattern, as described in (3.5) when defining the use of WOE in PD in Chapter 3.

This value provides an estimate of the probability of association with the class chosen as the label. As this maps the underlying probability of association within the data space itself, this is a reasonable choice as a means of estimating system failure.

The main source of noise in this estimate is the presence of the discretization bins, as all values within the same bin generate the same classification outcome and therefore, by necessity, have the same reported confidence based on the same defined events.

This conditional confidence cannot be directly used in the hybrid system as we cannot make the assumption of independence between the columns of input data.

In order to establish independence and approximate the confidence of the underlying PD classifier, the "PD (WOE Based) Probabilistic Confidence" scheme uses the WOE based conditional probability of equation (9.8) in conjunction with the WOE value computed by considering only the rules which would fire under the "independent" rule selection scheme, though the "occurrence" scheme is still used to select a label value. We must restrict our consideration to just this subset of the rules in order to maintain independence in the WOE calculation.

The benefit of this scheme is the simpler relationship with the underlying conditional probability. some interference with the correlation relation to "true" confidence is expected. as only a subset of the rules actually fired are used in the confidence calculation.

As in the $[0 \ldots 1]$ bounded τ metric, this scheme also uses only values internal to the FIS calculations.

9.2.4 Confidence Measure Evaluation Methodology

In order to compare these possible confidence values, a means of measuring their relative merit is required. The most direct method of evaluating confidence as a predictor of the true probability of correct classification is simply to correlate the reported confidence values with a "true" confidence calculated by examining the conditional probability of the winning class underlying the synthetic data distributions described in Chapter 5.

To perform this calculation, all of the tests performed in the jackknife tests on the synthetic data were examined and the confidence values recorded. These values were then compared with the conditional probability of occurrence of the winning class. The computation of the conditional probability for the synthetic data distributions is supplied in Appendix C. Each such comparison will result in a pair of data values, consisting of the expected and reported confidence. These values can then be correlated, and ideally will result in a straight line.

As the data to be correlated are quantized values, Spearmannrank correlation will be used, rather than the common Pearson rank correlation, which correlates values from continuous random variables.[†]

SpearmannRank Correlation

The Spearmannrank correlation coefficient (Press, Teukolsky *et al.*, 1992; Lehmann and D'Abrera, 1998) calculates the correlation between the ordinal positions of elements in a list.

The Spearmannranking is defined in Press *et al.* (1992, pp. 640) as

$$r_{Spearmann} = \frac{\sum_{i=1}^{N} \left[\left(R_{xi} - \overline{R_x} \right) \left(R_{yi} - \overline{R_y} \right) \right]}{\sqrt{\sum_{i=1}^{N} \left(R_{xi} - \overline{R_x} \right)^2} \sqrt{\sum_{i=1}^{N} \left(R_{yi} - \overline{R_y} \right)^2}} \tag{9.10}$$

where R_{xi} is the set of rankings in the x dimension and R_{yi} is the related set in the y dimension such that for an element $a_i \in X$, the ranking of this element is R_{xi}, R_{yi}.

The values $\overline{R_x}$ and $\overline{R_y}$ are the mean rankings in x and y respectively, and the ranks for duplicate values are all equal to the mean values of the rankings that would have been applied over the range, thus the list

$$X = \{\ (0, 7.5),\ (0, 10),\ (0, 12.5),\ (0, 15),\ (5, 15)\ \}$$

receives the rankings

$$X = \{\ (2.5, 1),\ (2.5, 2),\ (2.5, 3),\ (2.5, 4.5),\ (5, 4.5)\ \}.$$

The purpose of the Spearmannranking is to give a simple correlation calculation which is unbiased by the domain of the actual values, as long as they can be applied to an ordinal sequence.

In the Spearmannranking, a value of 0 indicates independence, and a positive or negative value indicates dependence along the major axis of $y{=}x$ or $Y{=}-x$, respectively.

[†] A discussion of the performance of several other possible metrics is provided in Appendix D. These measures include: mutual information, symmetric uncertainty and the interdependence redundancy measures. Results for these measures are provided in Appendix D but are not included here as the saturation of the measured values causes a significant distortion in the provided results. See the appendix for details.

The set

$$X = \{\ (0,0),\ (1,1),\ (2,2),\ (3,3)\ \}$$

will result in a Spearmannranking of 1. The set

$$X = \{\ (0,3),\ (1,2),\ (2,1),\ (3,0)\ \}$$

will result in a -1 ranking.

9.2.5 Confidence Measure Evaluation Results

Table 9.1 summarizes the Spearmannrank correlation of the probability of a successful classification and reported confidence for all the metrics examined. This table is provided to allow easy comparison between all reported data. Note that the table only contains values up to separation 2.0. Data at higher separations is not included as there are too few errors observed at these separations to provide a reliable estimate of the curve.

Table 9.1: Conditional Probability -vs- Reported Confidence SpearmannRanked Comparison

	C_{MICD}	$C_{\delta/\tau \text{ Probability}}$	$C_{\tau[0...1]}$	$C_{\text{PD-Probabilistic}}$
Covaried				
0.125	0.952	1.000	0.955	0.915
0.250	0.957	0.996	0.979	0.914
0.500	0.933	0.999	0.977	0.930
1.000	0.957	0.997	0.963	0.932
2.000	0.955	0.990	0.960	0.977
μ	0.951	0.996	0.967	0.933
Bimodal				
0.125	0.894	0.988	0.931	0.913
0.250	0.858	0.988	0.925	0.869
0.500	0.886	0.991	0.930	0.891
1.000	0.930	0.993	0.932	0.925
2.000	0.871	0.974	0.888	0.940
μ	0.888	0.987	0.921	0.907
Spiral				
0.125	0.399	0.841	0.612	0.387
0.250	0.842	0.931	0.889	0.645
0.500	0.914	0.938	0.882	0.960
0.750	0.683	0.727	0.665	0.931
1.000	0.586	0.916	0.583	0.892
μ	0.685	0.871	0.726	0.763

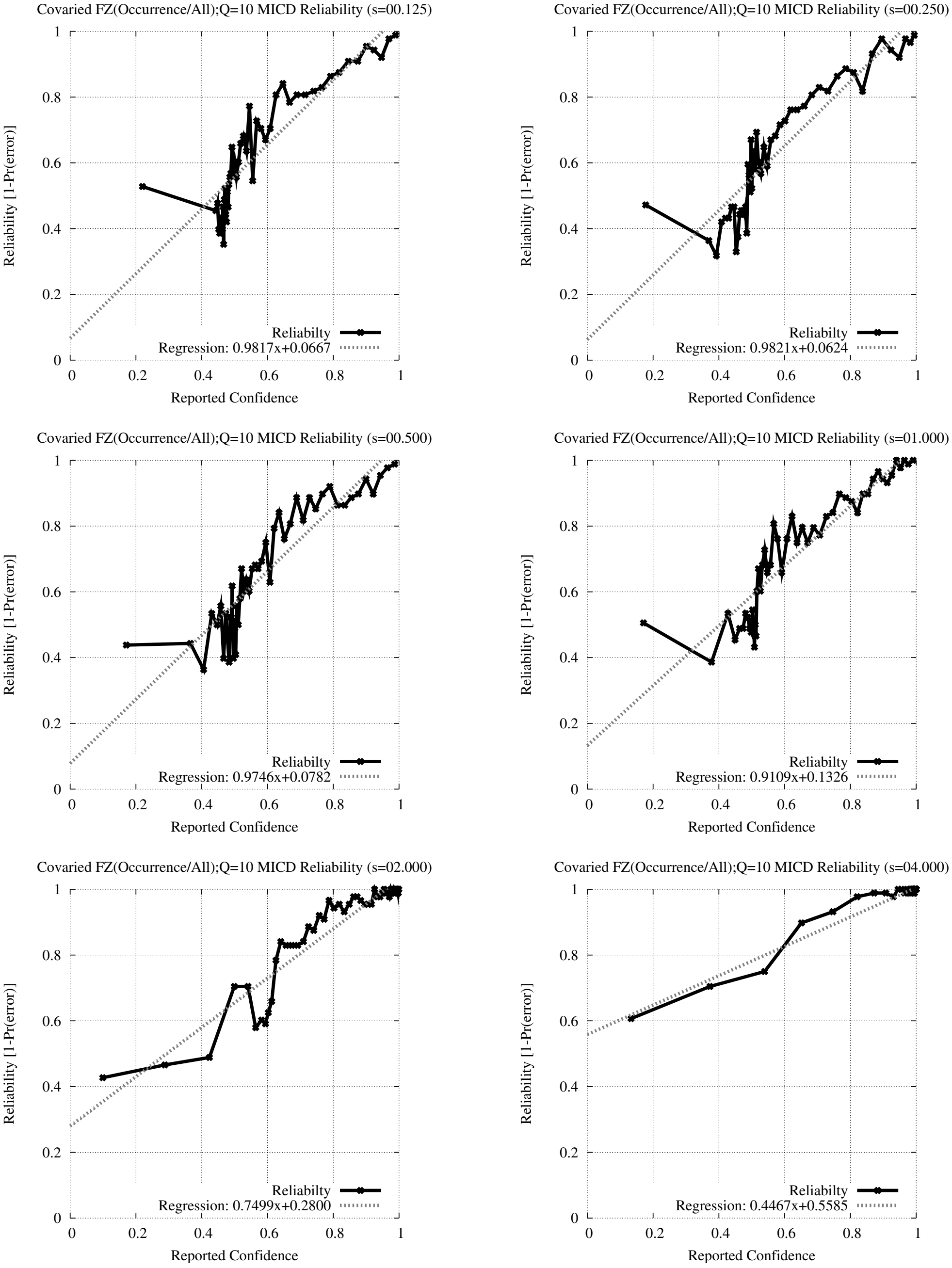
MICD Based Confidence: C_{MICD}

Figures 9.3, 9.4 and 9.5 display the relationship found between C_{MICD} and the probability of a successful classification for the covaried, bimodal and spiral data respectively. Examining these figures, we see a significant amount of noise, however a general trend towards higher reliability at higher reported confidence is visible. The linear regression line placed across the plots clearly shows the trend to be in the correct direction.

In Figure 9.3 there is quite a bit of noise in the central region of the curve, especially at low separations when many errors are being made. As the problem becomes easier at higher separations, the line straightens out and a higher correlation is recorded. This is also seen in Table 9.1, where the values for this confidence measure are shown in the column entitled C_{MICD}.

Notable on the covaried plots at low separation in Figure 9.3 is the downward portion of the initial part of the line. This portion of the plot is based on the MME bin with the largest range, and as such contains points from a large range of confidences. This explains why the line deviates so sharply from the trend in the rest of the graph. As this line indicates only that the reliability may be quite high when the confidence is low, this artifact is not a critical problem.

Bimodal data, which exhibits fewer classification errors to begin with, has much less noise, as seen in the plots in Figure 9.4. The spiral data in Figure 9.5 shows the most noise in the confidence/reliability curve, as more errors are made in the assignments. As the bimodal data set is quite information rich in comparison with the other two synthetic data sets, this response is expected.



Covaried FZ(Occurrence/All);Q=10 MICD Reliability (s=00.125)

Covaried FZ(Occurrence/All);Q=10 MICD Reliability (s=00.250)

Covaried FZ(Occurrence/All);Q=10 MICD Reliability (s=00.500)

Covaried FZ(Occurrence/All);Q=10 MICD Reliability (s=01.000)

Covaried FZ(Occurrence/All);Q=10 MICD Reliability (s=02.000)

Covaried FZ(Occurrence/All);Q=10 MICD Reliability (s=04.000)

Figure 9.3: C_{MICD} Plots on 4-Feature Covaried Data

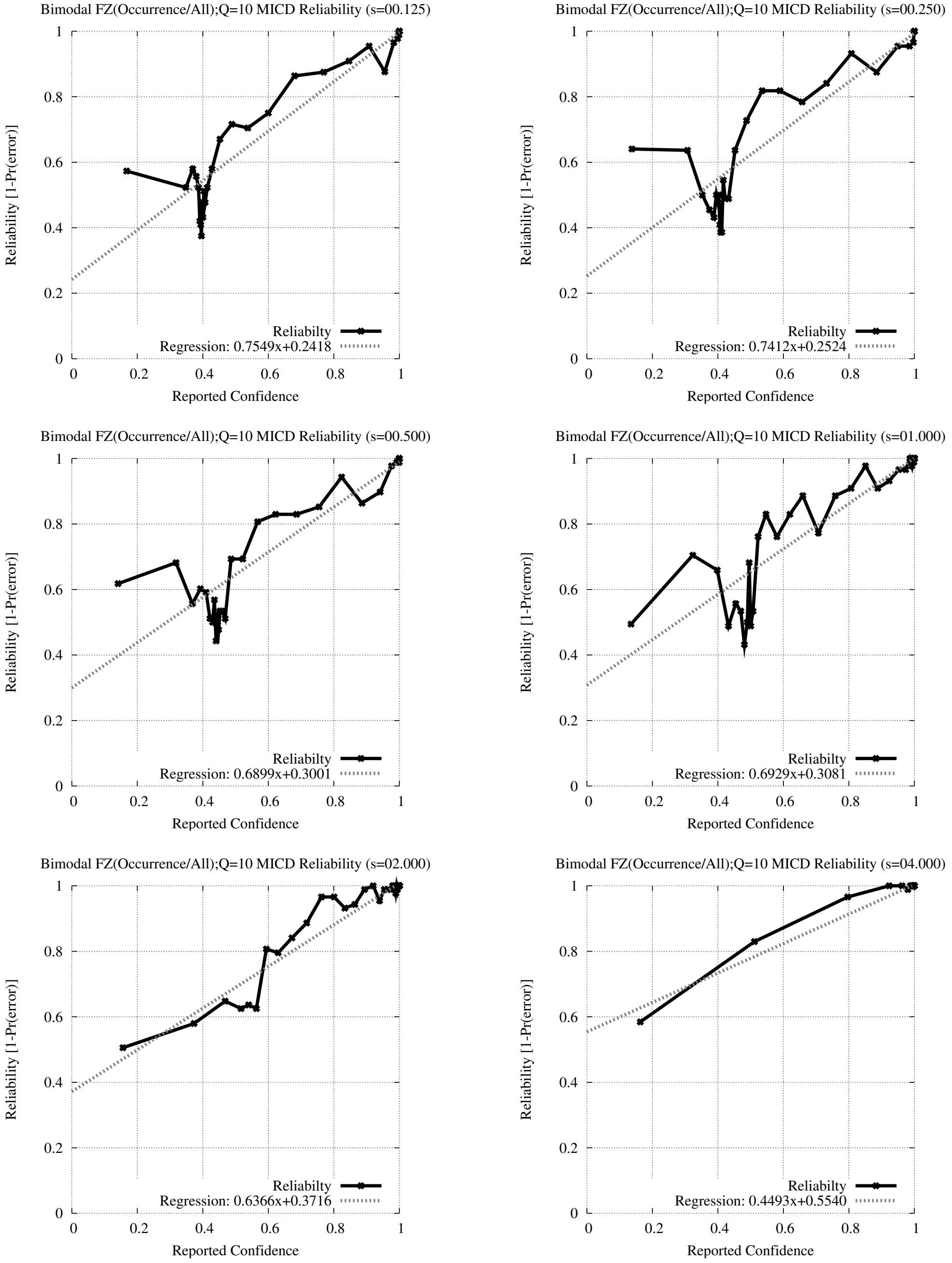


Figure 9.4: C_{MICD} Plots on 4-Feature Bimodal Data

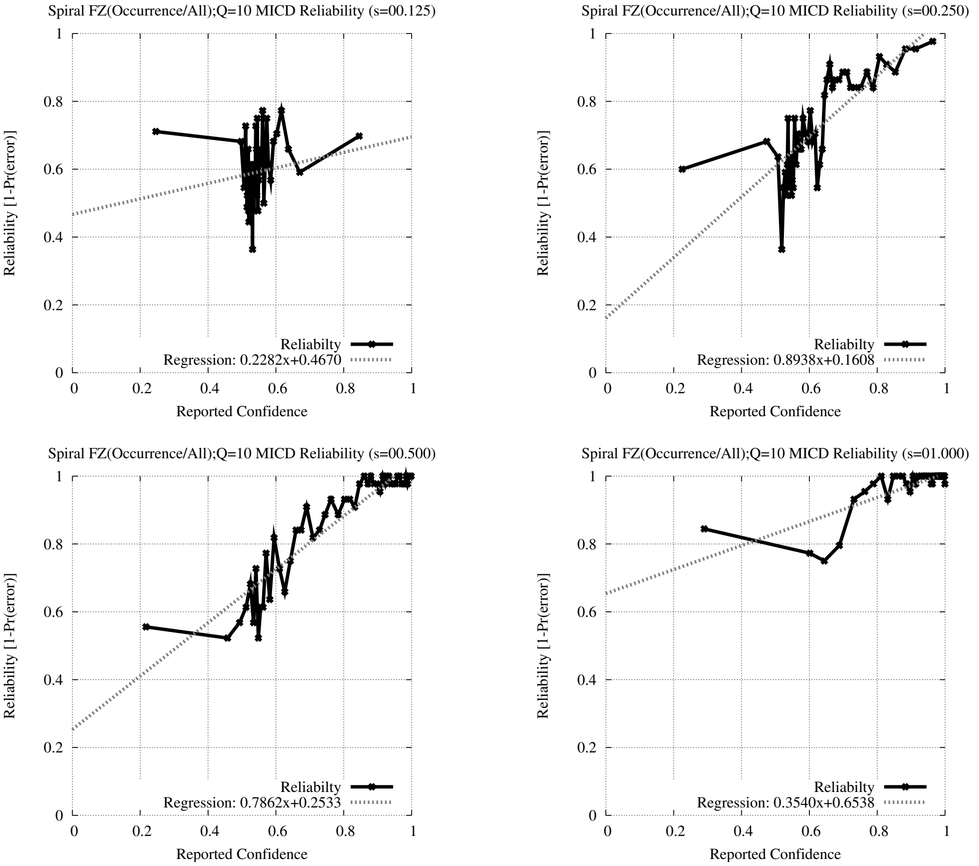


Figure 9.5: C_{MICD} Plots on 4-Feature Spiral Data

δ/τ Observed Probability Confidence: $C_{\delta/\tau\ \textbf{Probability}}$

The plots showing the correlation with reliability for the $C_{\delta/\tau\ \text{Probability}}$ measure are shown in Figures 9.6, 9.7 and 9.8, again for covaried, bimodal and spiral data respectively. All three figures exhibit a strikingly noise-free estimation. Both the covaried data in Figure 9.6 and the bimodal data in Figure 9.7 show little deviation from the plotted linear regression. The slope of the regression itself clearly approximates the x=y desired slope on these two plots. There is some decrease in the slope as the separation increases, as the number of overall errors drops. Examining the spiral data in Figure 9.8, we see significantly more noise than is shown in either of Figures 9.6 or 9.7. As in the MICD based confidence measure, it seems that when the problem is harder, there is more noise in the confidence estimate.

Evaluating the Spearmanncorrelation coefficients in Table 9.1, we see that this confidence measure, marked $C_{\delta/\tau\ \text{Probability}}$ has the highest correlation with true reliability.

While the downward-sloping trend is present at lower measured reliability values, it is less significant than when seen in the C_{MICD} plots.

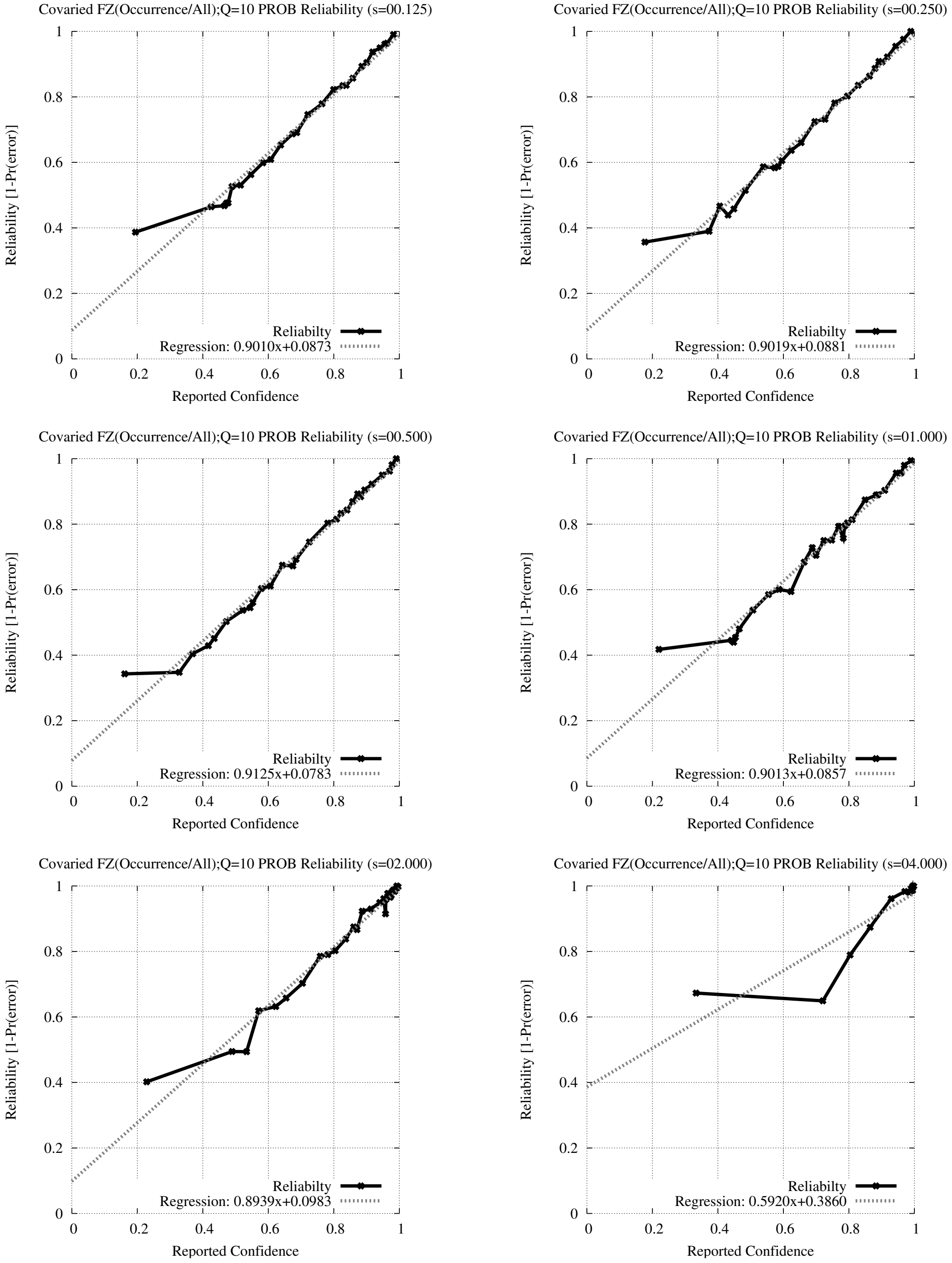


Figure 9.6: $C_{\delta/\tau}$ Probability Plots on 4-Feature Covaried Data

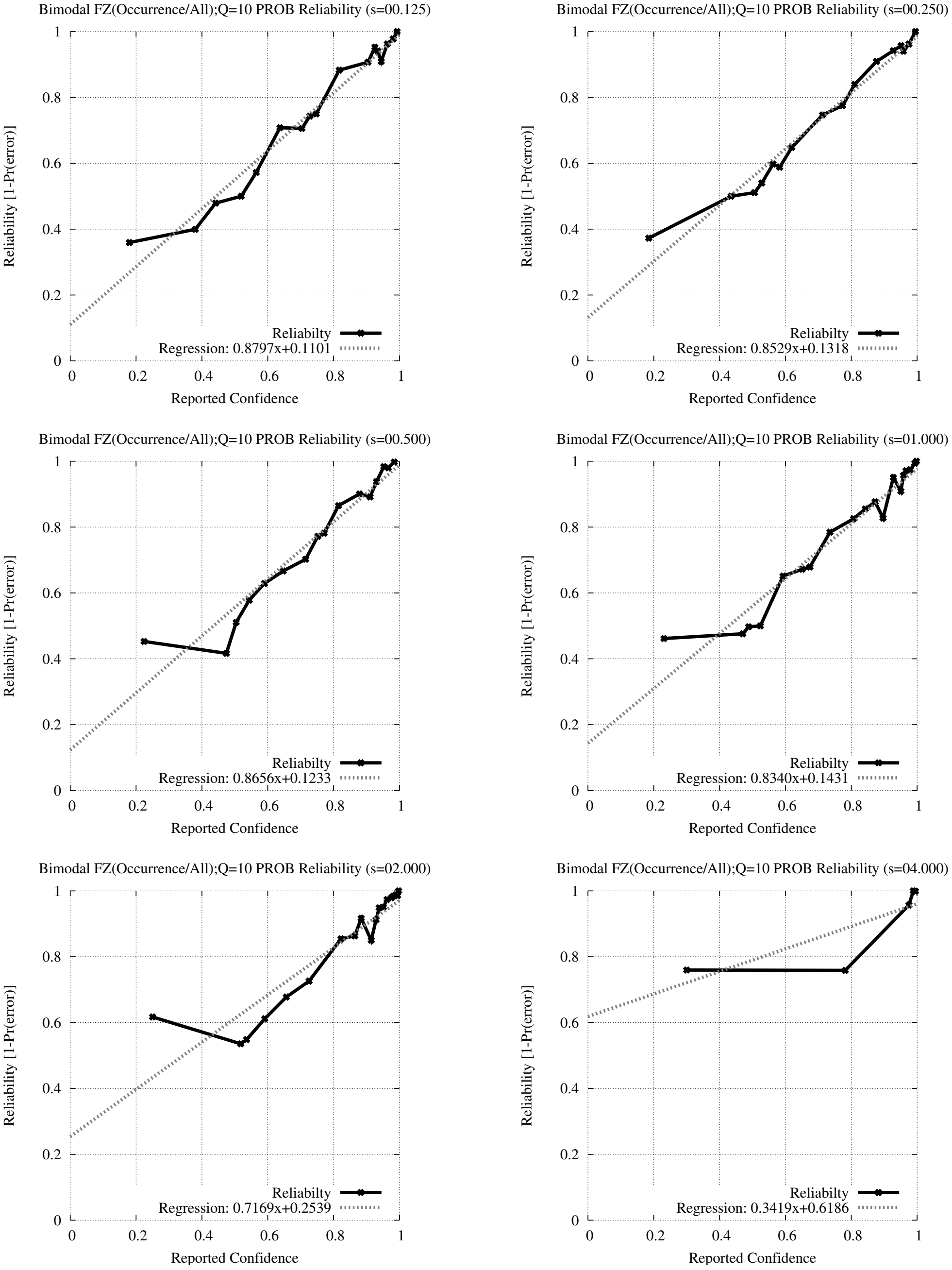
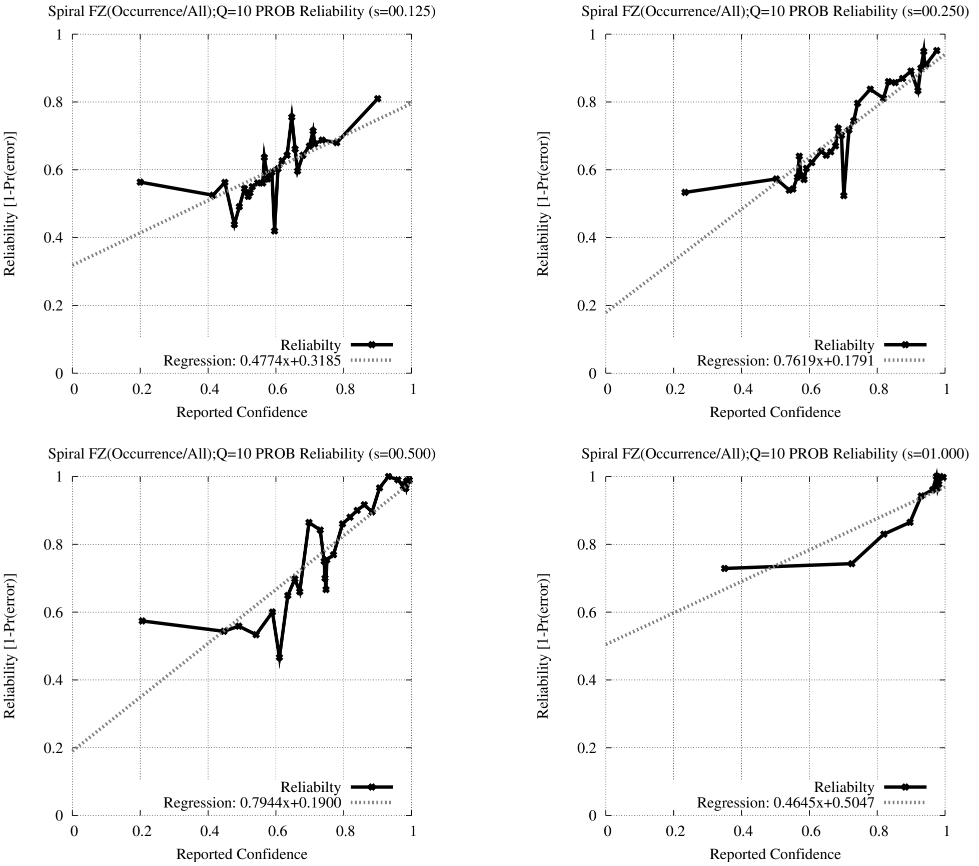


Figure 9.7: $C_{\delta/\tau \text{ Probability}}$ Plots on 4-Feature Bimodal Data



Spiral FZ(Occurrence/All);Q=10 PROB Reliability (s=00.125)

Spiral FZ(Occurrence/All);Q=10 PROB Reliability (s=00.250)

Spiral FZ(Occurrence/All);Q=10 PROB Reliability (s=00.500)

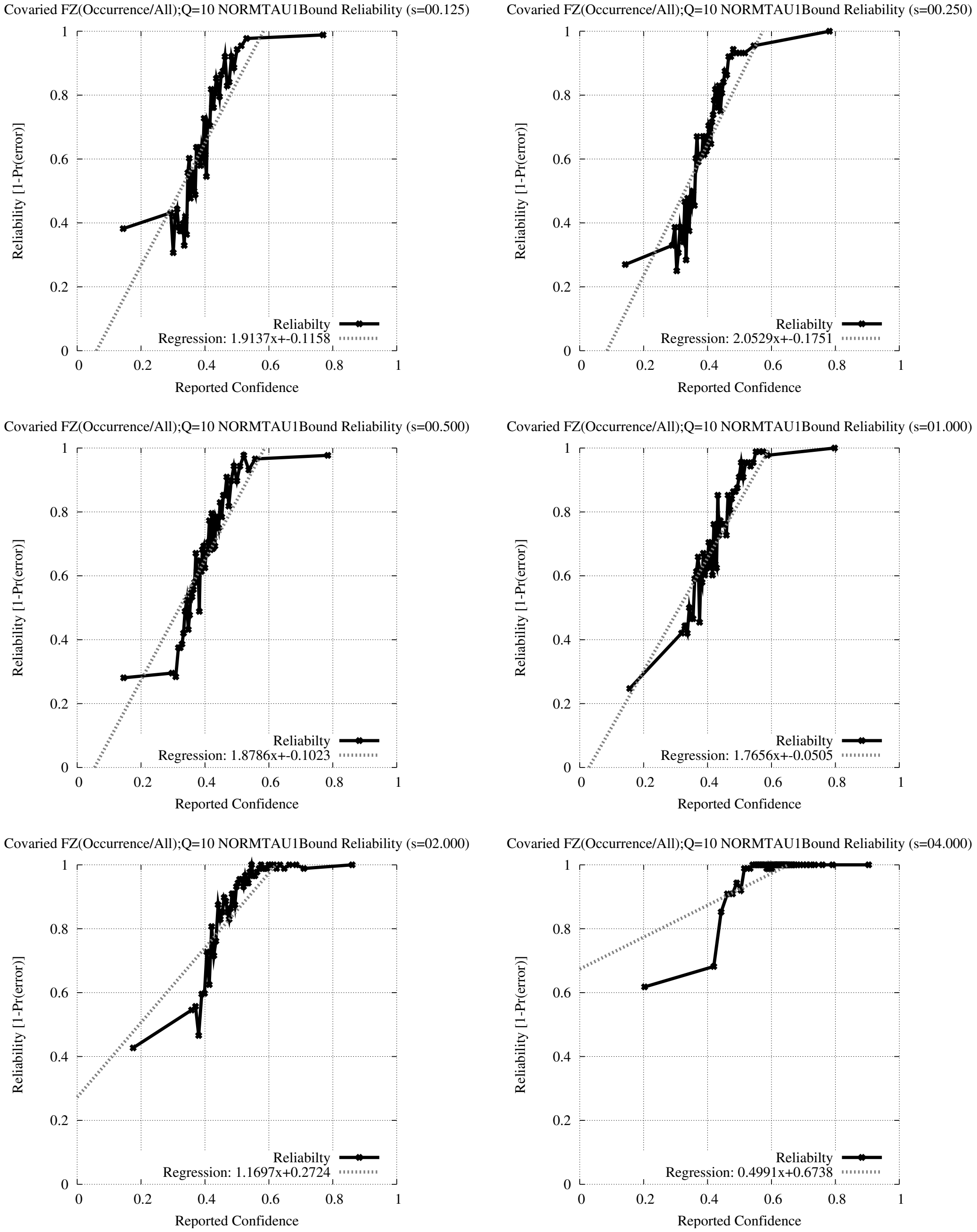
Spiral FZ(Occurrence/All);Q=10 PROB Reliability (s=01.000)

Figure 9.8: $C_{\delta/\tau\ \text{Probability}}$ Plots on 4-Feature Spiral Data

Normalized $[0 \ldots 1]$ **Bounded** τ **Confidence:** $C_{\tau[0\ldots1]}$

Turning to the Normalized $[0 \ldots 1]$ bounded τ confidence measure, we see that the correlations shown in Figures 9.9, 9.10 and 9.11 (covaried, bimodal and spiral) again show a strong linearity, however with more noise in the estimate than is the case in $C_{\delta/\tau \text{ Probability}}$.

The Spearmannranking in Table 9.1 shows us that this measure is comparable to that of C_{MICD}, however is not as strong as $C_{\delta/\tau \text{ Probability}}$.



Covaried FZ(Occurrence/All);Q=10 NORMTAU1Bound Reliability (s=00.125)

Covaried FZ(Occurrence/All);Q=10 NORMTAU1Bound Reliability (s=00.250)

Covaried FZ(Occurrence/All);Q=10 NORMTAU1Bound Reliability (s=00.500)

Covaried FZ(Occurrence/All);Q=10 NORMTAU1Bound Reliability (s=01.000)

Covaried FZ(Occurrence/All);Q=10 NORMTAU1Bound Reliability (s=02.000)

Covaried FZ(Occurrence/All);Q=10 NORMTAU1Bound Reliability (s=04.000)

Reliability [1-Pr(error)]
Reported Confidence
0
0.2
0.4
0.6
0.8
1
Reliabilty
Regression: 1.9137x+-0.1158
Regression: 2.0529x+-0.1751
Regression: 1.8786x+-0.1023
Regression: 1.7656x+-0.0505
Regression: 1.1697x+0.2724
Regression: 0.4991x+0.6738

Figure 9.9: $C_{\tau[0...1]}$ Plots on 4-Feature Covaried Data

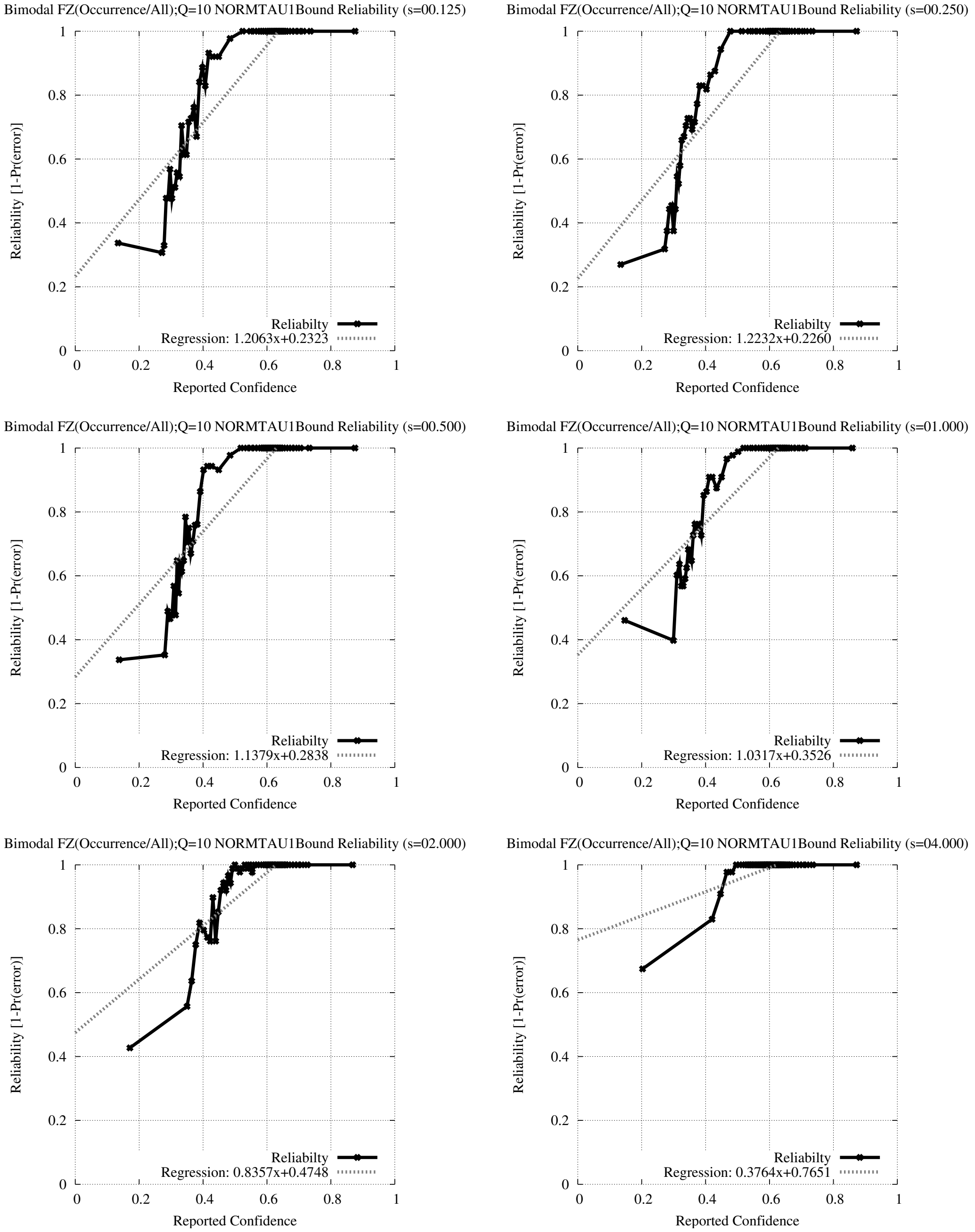


Figure 9.10: $C_{\tau[0...1]}$ Plots on 4-Feature Bimodal Data

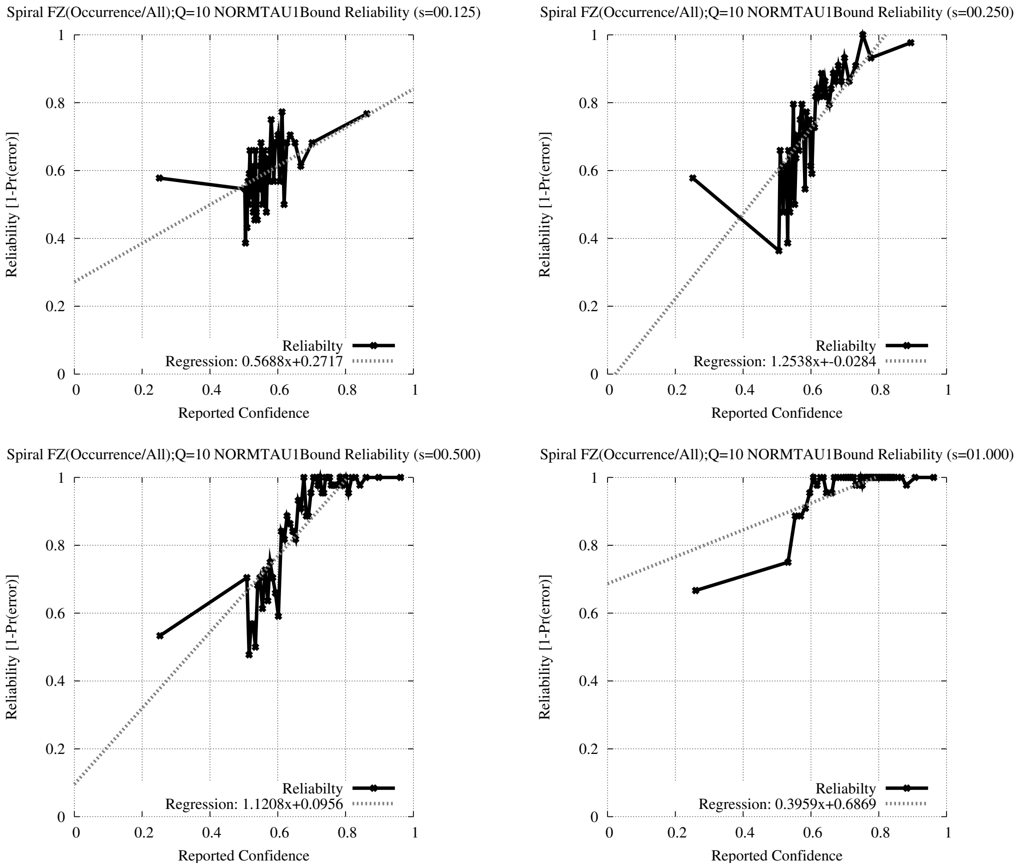


Figure 9.11: $C_{\tau[0...1]}$ Plots on 4-Feature Spiral Data

PD (WOE Based) Probabilistic Confidence: $C_{\textbf{PD-Probabilistic}}$

Examining the performance of the $C_{\text{PD-Probabilistic}}$ column in Table 9.1, we see performance which is significantly poorer than that of the other classifiers.

A look at Figures 9.12, 9.13 and 9.14 show two significant things: there is quite a lot of noise in all three plots and the slope of the lines on each plot are considerably lower than those of the related plots in the other measures.

While neither of these pose disastrous problems (as noted by the high correlation in Table 9.1), the comparison with the other confidence measures shows that this estimation technique is inferior.

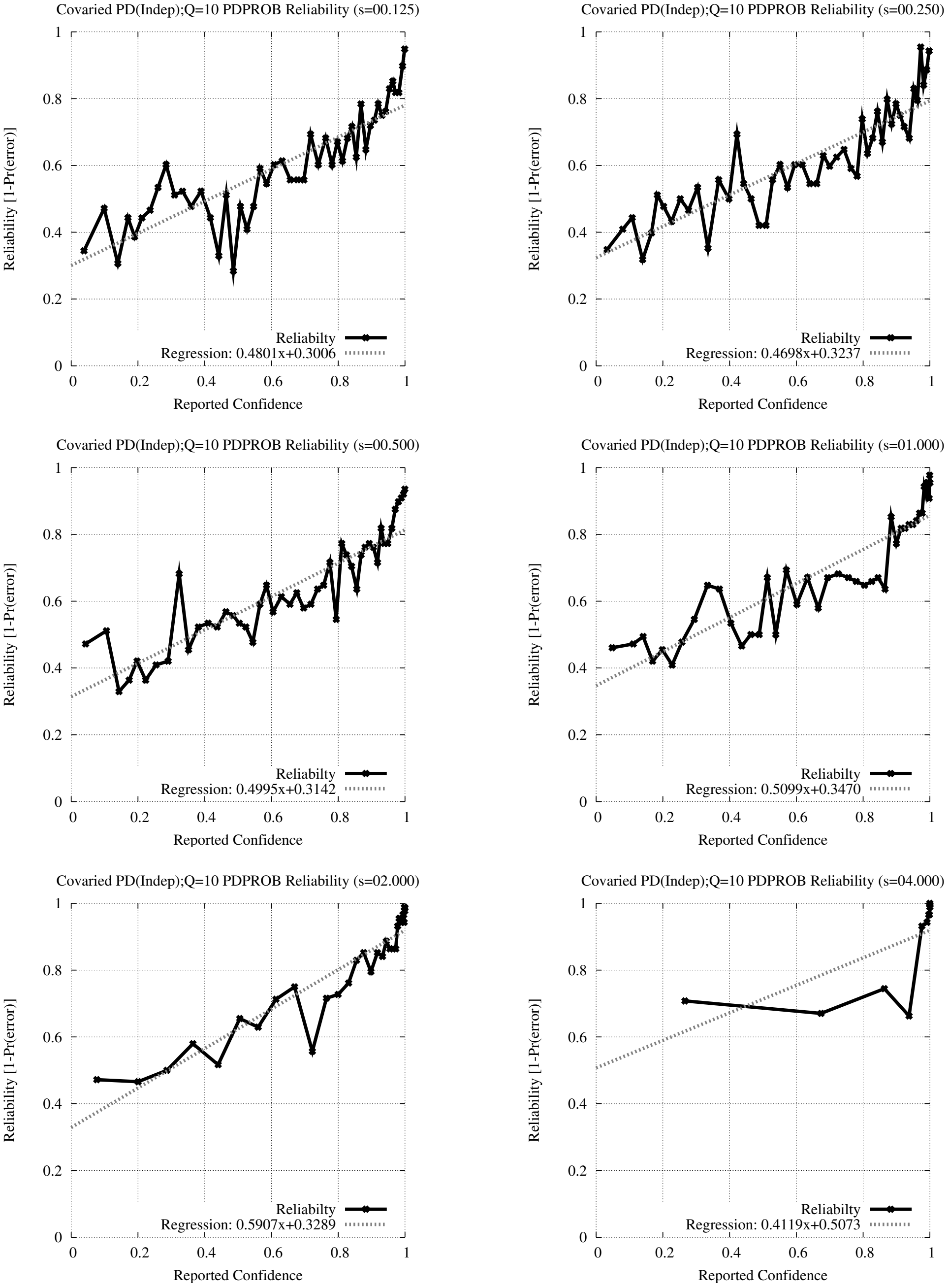


Figure 9.12: $C_{\text{PD-Probabilistic}}$ Plots on 4-Feature Covaried Data

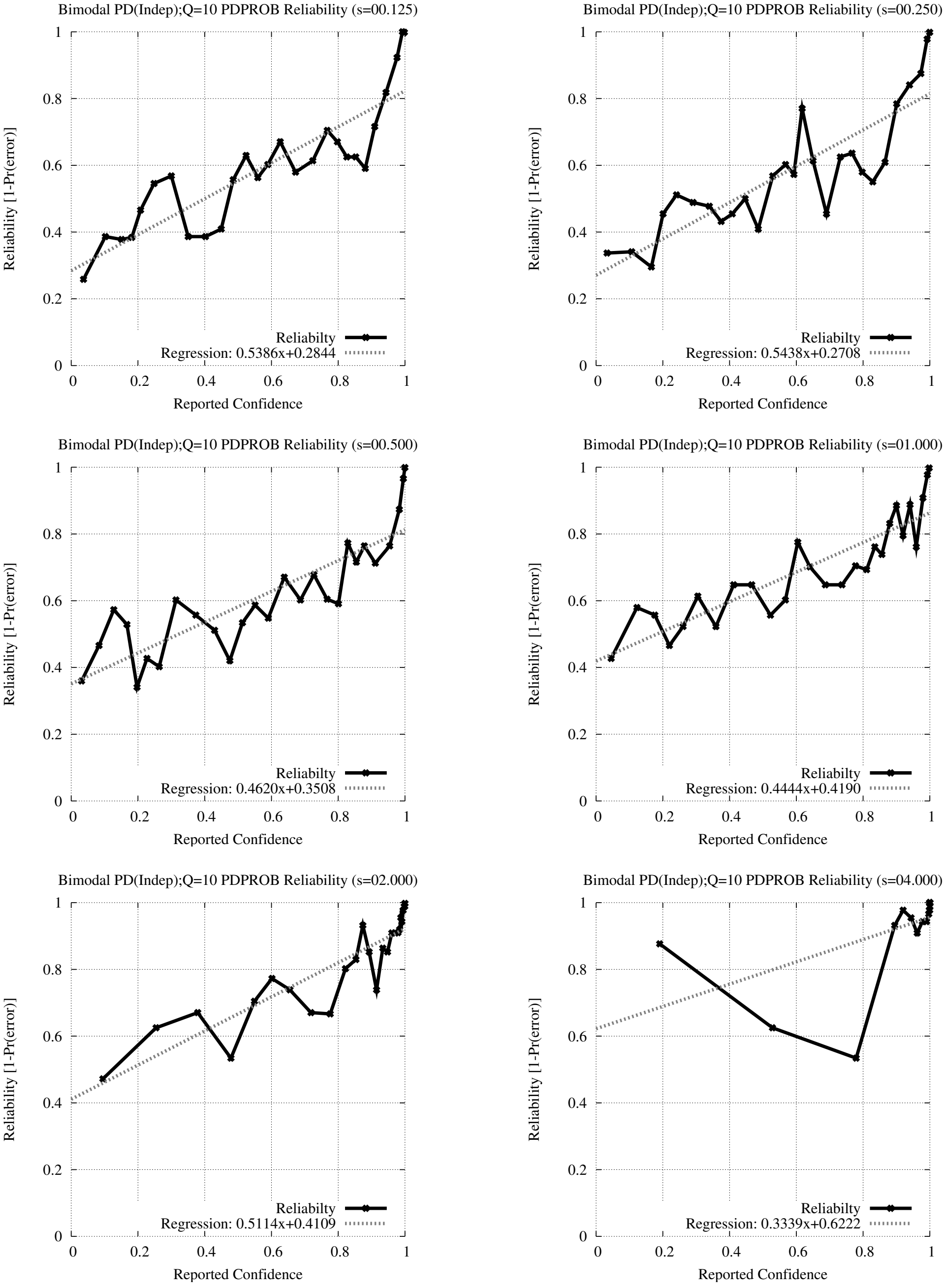
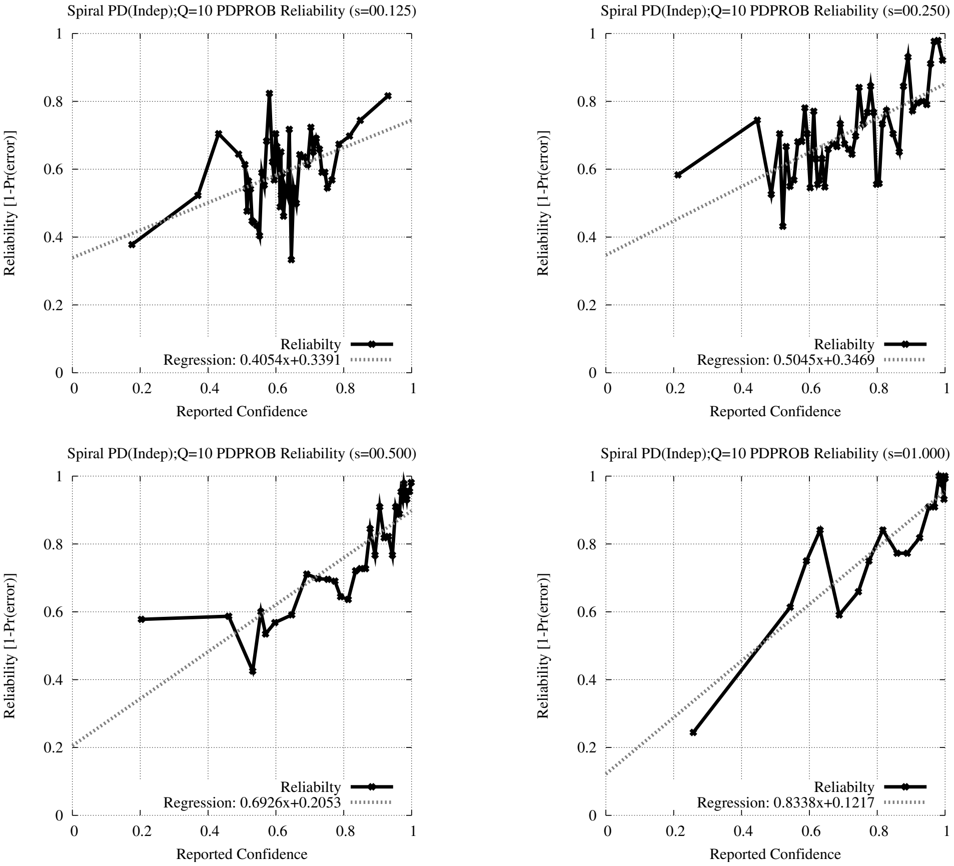


Figure 9.13: $C_{\text{PD-Probabilistic}}$ Plots on 4-Feature Bimodal Data


Spiral PD(Indep);Q=10 PDPROB Reliability (s=00.125)

Spiral PD(Indep);Q=10 PDPROB Reliability (s=00.250)

Spiral PD(Indep);Q=10 PDPROB Reliability (s=00.500)

Spiral PD(Indep);Q=10 PDPROB Reliability (s=01.000)

Figure 9.14: $C_{\text{PD-Probabilistic}}$ Plots on 4-Feature Spiral Data

9.3 Discussion

All of the measures discussed provide an overall linear relationship with respect to reliability. Further, the trend of all measures show increasing reliability correlated with increasing reported confidence.

The MICD based measure shows quite acceptable performance, showing saturation of the reported confidence at the same time that total reliability is reached.

The $C_{\tau[0...1]}$ measure, while not showing large amounts of noise, saturates at complete reliability significantly before the confidence measure predicts this. Using a confidence measure which under-predicts the true reliability is certainly preferable to an over-prediction, however this does decrease the measured correlation significantly.

The measure $C_{\text{PD-Probabilistic}}$ shows significant noise, and is the poorest of all of the measures discussed. The noise in this measure is due largely to the quantization of confidence along with the quantization of decision outcome based on the MME quantization of the PD data space.

A further consideration in $C_{\text{PD-Probabilistic}}$ is that the rules fired for this measure do not exactly correspond to those used for WOE calculation, as only the rules which would be fired using the PD rule-independence scheme are used. This will introduce a further confounding factor into the relationship between assertion confidence and this measure of confidence reporting.

The plots for the $C_{\delta/\tau \text{ Probability}}$ measure show the least deviation from linearity. The regression line on these plots shows that there is a close relation with the desired reliability as the slope is near 1.0. This outcome is not surprising, as of the four measured evaluated, this is the only one which is directly based on an evaluation of error; the high correlation is expected in a calculation directly reporting observed error during training.

The smoothness of the lines in Figures 9.6, 9.7 and 9.8 is further due to the averaging effect of the histogram calculation of expected confidence combined with the averaging effect of the quantized reliability calculation. This “double filtering” in the $C_{\delta/\tau \text{ Probability}}$ confidence reduces the noise exhibited by this measure, and provides a better estimate of the overall system reliability.

The irregularities which appear in the spiral plots in Figure 9.8 are due to the poor separation of errors in the δ/τ space for this problem. This in turn is driven by the difficulty of the problem overall, indicating that there is a relationship between the information available to the system to solve a problem, and the information available to estimate the reliability of the answer within the problem.

9.4 Conclusion

While all of the measures exhibit reasonable performance, the extremely good correlation of the $C_{\delta/\tau \text{ Probability}}$ measure with respect to true system reliability indicates that this will be the best estimator to choose.

In order to use this estimator with a given decision support problem, a histogram of δ/τ values must be constructed. The construction of this histogram need only use an examination of the same training data used to construct the rule-base itself. Once created, this histogram functions as a lookup table from which the confidence values are chosen.

The explainability of this measure is very high, as it allows a decision to be characterized in terms of the reliability on similar decisions made using training data. An explanation of this form allows a decision-making user to understand that the confidence is not constant for the system, but is easily parameterized by a set of direct measures (δ and τ) based on the training data.

$C_{\delta/\tau \text{ Probability}}$ therefore is the measure which will be used in the decision support interface in order to report decision confidence.

The presentation of such a confidence value allows a decision maker using the suggested label to combine the DSS presented suggestion with data available from other sources. While this idea is normally conceived in terms of a human decision maker, an equally plausible scenario would see the suggested classification label and the confidence value as inputs to a computationally based multi-classifier voting system.

9.5 Summary

Several possible candidate measures for decision confidence have been evaluated. The measure selected for this hybrid system is that which provides the best indication of the system performance and which best estimates the reliability of the suggestion. This measure will allow the user to incorporate the likelihood of an erroneous label being suggested into a larger decision framework. The contribution of this confidence measure within this work provides a new tool for measuring PD based performance and estimating system confidence.

This type of confidence measure, coupled with a transparent means of determining a suggested label, are the defining characteristics of a functional DSS.

Chapter 10

Decision Support and Exploration

Statistics are no substitute for judgement.

— Henry Clay

Decision support is a rigorous field in which the mechanism by which data is presented to the user will define the utility of the system. The entire purpose of a decision support system is to provide explainable insights into the decision process, based on numerical evaluation. If the results of this evaluation are opaque or obscure, the resulting system will be confusing and, at worst, may make the resulting decisions less reliable.

As stated by Edward Tufte (1997, pp. 27):

> *When we reason about quantitative evidence, certain methods for displaying and analyzing data are better than others. Superior methods are more likely to produce truthful, credible, and precise findings. The difference between an excellent analysis and a faulty one can sometimes have momentous consequences.*

10.1 Design

It is therefore important to evaluate the work-flow by which a user of the hybrid system described here will obtain and evaluate a decision suggested by this system. The resulting inference exploration is top-down, starting with the presentation of a class label for a given input vector and working back towards

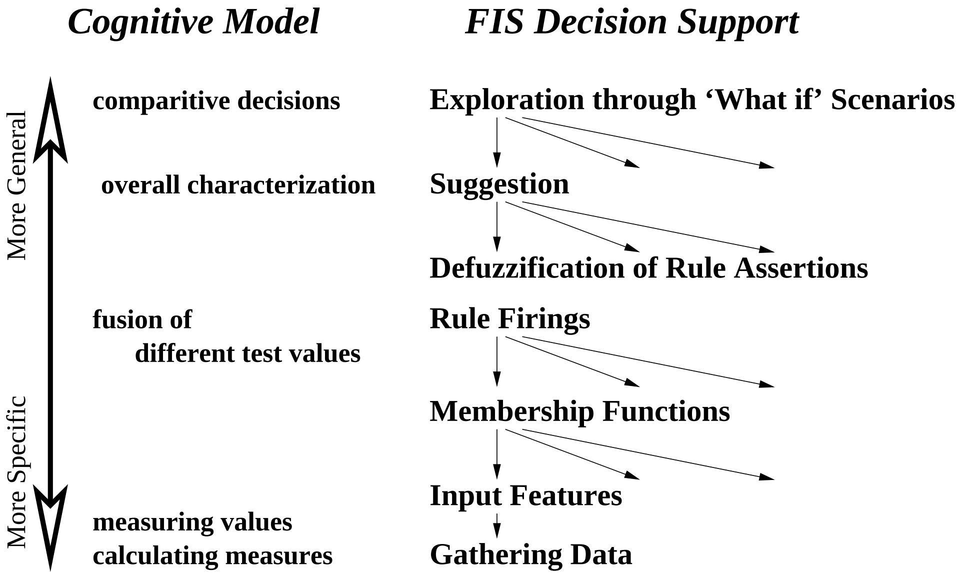


Figure 10.1: Hierarchical Design Encourages Drill-Down Exploration

the underlying statistical support structures forming the basis for the decision. Effectively, the process by which the classifier functions must be examined by running the data through in reverse.

The decision support system (DSS) supports the cognitive model pictured in Figure 10.1 by modelling the problem using a similarly layered logical abstraction as the conceptual abstractions in the cognitive model.

10.2 Evaluation

In order to evaluate the DSS proposed here, a detailed discussion of the user work-flow will be discussed for a few illustrative examples.

The following inference results will be examined:

- a straightforward positive classification (one for which PD has infinite WOE). This will provide an overall flavour of the system and its confidence support.
- a classification at the decision boundary of a simple domain. This will allow rule and data explanations containing more complexity.
- a positive classification based on real-world data, containing a small degree of conflict. This will

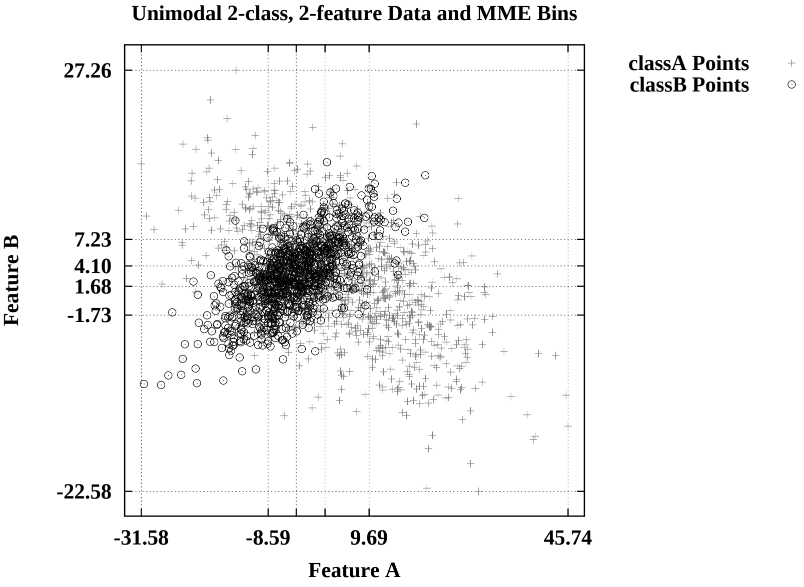


Figure 10.2: MME Division of 2-class, 2-feature Covaried Data

introduce the measurement of uncertainty as it is presented to the decision-maker, relating the confidence measure described in Chapter 9 to the exploration of the FIS rules.

- a positive real-world data classification with significant conflict. This will provide a further exploration of the uncertainty measurement, with further stress on the ability of the decision maker to assure themselves of the degree of support for any possible decision.

10.2.1 Unimodal, 2-Feature Covaried Data

Two feature covaried data will be used to provide a simple example to begin the discussion. Simple 2-class, 2-feature data was created by generating 1000 records for each of two classes (classA and classB). This data was then coloured using covariance matrices of $\mathbf{Cov}_A = \left[\begin{smallmatrix} 160 & -52.58 \\ -52.58 & 48 \end{smallmatrix}\right]$ and $\mathbf{Cov}_B = \left[\begin{smallmatrix} 48 & 16.63 \\ 16.63 & 16 \end{smallmatrix}\right]$. The means of the two distributions were separated along the x axis by 9.36. The PD/FIS algorithm was then run on this data with Q=5 quantization intervals, generating descriptive rules as well as the quantization grid shown in Figure 10.2. This grid shows the training data points as well as the quantization bin boundaries into which they have been divided. Note that the most extreme points at top, bottom, left and right fall exactly onto (and actually define) the outer grid boundaries. This two-dimensional data set can be easily

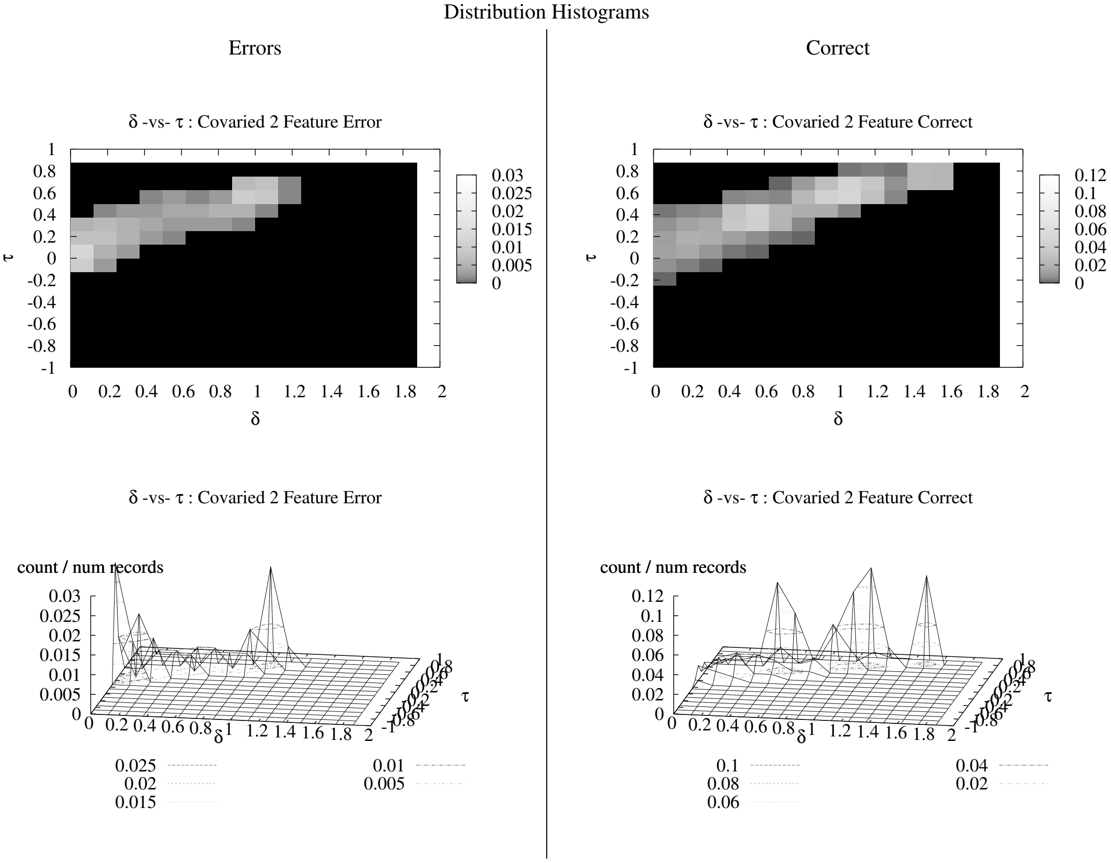


Figure 10.3: δ/τ Histograms from Covaried 2-feature Data

visualized, allowing the relationship between the measured data values and the decision space to be well understood.

Figure 10.3 shows the δ/τ histogram surfaces for this data set, upon which we will calculate $C_{\delta/\tau \text{ Probability}}$ confidence as the measure chosen in Chapter 9. In this figure we see that the extent of the errors is similar to that of the correct classifications, however there are significantly more correct observations in the intersecting area.

F1	F2	*Suggestion*	*Confidence*	*Conflict*	*# of Rules Fired*
40.0,	−20.0	classA	1.000	0.0	6 of 54

Figure 10.4: Covaried Classification With High Confidence Summary

10.2.2 Unimodal, 2-Feature Covaried Classification With High Confidence

For the first analysis, consider the record (40, −20), which in the simple data topology of Figure 10.2, will appear in the lower-right corner. Records at this location are unambiguously associated with classA.

If we assign this point a label using the FIS in a DSS context, we are presented with the summary output display shown in Figure 10.4, which shows the input data from features F1 and F2, along with: the suggested labelling; the decision confidence calculated using $C_{\delta/\tau \text{ Probability}}$; and the decision conflict. The conflict is calculated simply using

$$\text{Conflict} = \begin{cases} \tau - \tau_2 & \text{if } \tau_2 > 0 \\ 0 & \text{otherwise.} \end{cases} \tag{10.1}$$

As is obvious in Figure 10.4, there is no conflict for this simple example, and the projected reliability is unity. While rather an extreme projection of confidence, due to the position of the point in a portion of the space with no conflicting classifications having a τ value of 0.858 and a δ of 1.573. Locating these values in Figure 10.4 places the (δ, τ) location in the black bar above all of the observed errors, but within the observed correct classifications. This location therefore has the support of previous correct classifications, and has never been associated with an error, so the reported confidence is in fact reasonable based on $C_{\delta/\tau \text{ Probability}}$.

The output shown in Figure 10.4 is kept terse for two major reasons:

- the purpose of a decision support system is to reduce the cognitive load by summarizing decision information. The decision exploration task is driven by user request in order to allow greater clarity in the summarization;
- this technique will allow summaries from multiple decisions to be collected — this will allow “what if?” scenarios to be explored, where the decision maker can examine the effect of changing an input value without removing the result of the true values. Such a scenario will be discussed at the end of

	Assertion	*Implied When . . .*
		(F1 is Very High and F2 is Very Low)
classA	1.000	F1 is Very High and F2 is Very Low
	0.928	F1 is Very High
	0.645	F2 is Very Low
	0.858	**Support for classA (Chosen Label Value)**
	1.000	*Confidence*
	0.000	*No Assertion Conflict*

		(F1 is Very High and F2 is Very Low)
classB	−0.290	F2 is Very Low
	−0.855	F1 is Very High
	−1.000	F1 is Very High and F2 is Very Low
	−0.715	Refutation for classB

Figure 10.5: Covaried Classification With High Confidence Rule Set

the Chapter.

Decision Exploration: Drill-Down Into Rules

Assuming the user would like further information regarding the means by which the system arrived at this suggestion, rules associated with this suggestion may be displayed. The appropriate rules are presented, ranked in decreasing order of assertion value, separated by the class of support, as shown in Figure 10.5, which shows the same data as was indicated in Figure 10.4.

As is seen in Figure 10.5, only six rules have been used to characterize this data value, so a complete inspection of all the rules associated with this assignment will be both clear and informative.

The display shows the results in support (or refutation) of the suggested labelling on the top, with the label value highlighted. Choices for other classes are then displayed in decreasing order of overall assertion value. In this example there are only two possible labels, “classA” and “classB.”

Considering the rules associated with classA, one has fired with the highest possible assertion value (1.0), and the other two have fired with strongly positive assertions. Note that the higher order rules displayed here have higher assertion values. This is consistent with a rule set that has only support for one of the possible labellings, and this is reflected to the user by the phrase “No Assertion Conflict.”

When examining the competing class, the display of Figure 10.5 indicates that there is indeed no

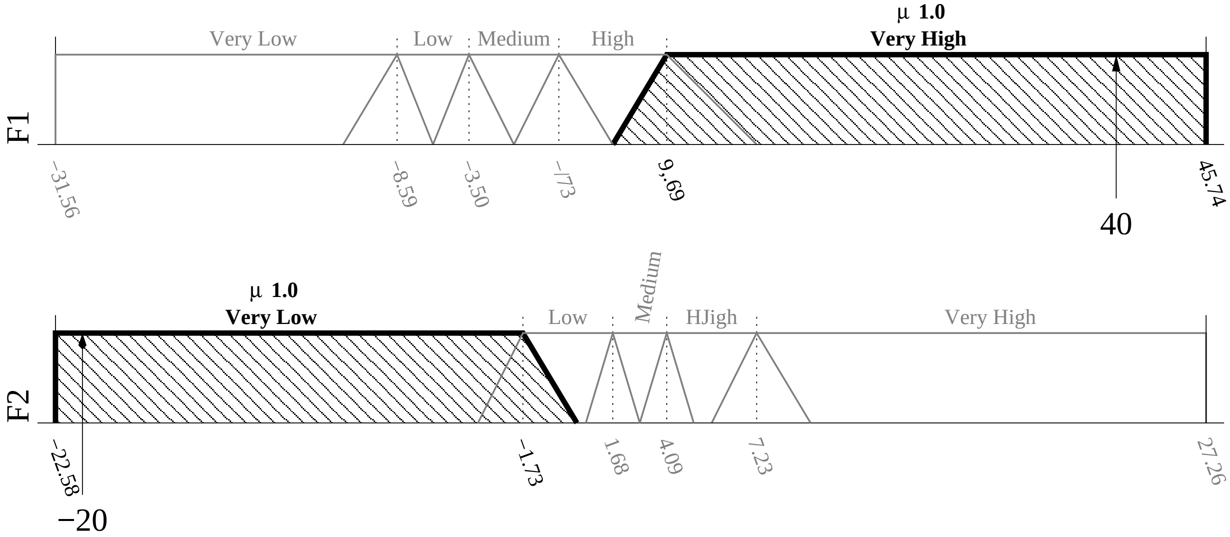


Figure 10.6: Membership Values For 2-Feature Simple Classification Example

support at all for CLASSB as all the assertion values are negative (indicating a complete refutation of the label CLASSB). This is indicated in the summary line with the word "refutation" instead of "assertion."

To review the success of this display, it is illustrative to list the goals which have been met:

Relay Complexity of Decision : In this simple decision, all the rules have been presented.

Describe Degree of Conflict : Conflict is explicitly labelled in this simple plot. The conflict value is intentionally offset in the column of numerical data in order to avoid visual clutter with the assertion value.

Decision Exploration: Drill-Down Into Membership Functions

If the user wishes to "drill down" to discover the degree of membership in the various input classes, the display of Figure10.6 is shown. This display is part of the same example featured in both Figure 10.4 and Figure 10.5, and so shows the same input data values.

This display shows the number line between the minimum and maximum observed training values which forms the universe of discourse (UOD) for this fuzzy relation. The extent of the universe is covered by the fuzzy membership functions describing the various fuzzy input sets.

The names of the fuzzy sets are shown above the membership functions, and the points which form the bin boundaries in the underlying PD training configuration are labelled. These are the same points that correspond to the ends of each trapezoidal plateau, from which the ramp extends. The end points of the ramps are not labelled, to avoid excessive "chart-junk" as described by Tufte (1983, pp. 100-121).

Also following the principles outlined by Tufte (1997, pp. 73-78), the "smallest effective difference" is employed to highlight the information in the display while maintaining the background of supporting information in a non-distracting form. This is achieved by highlighting the membership function to which the input point is assigned using the same line style but with a darker weighting. The set name is also coloured with a darker pen. The membership within the set is added above the set name, associating the set with its information content and dissociating this set from those which are irrelevant to further discussion. Finally, the input point itself is added, along with a vertical line to describe the point at which the input enters the number line and a tag to describe its value.

Given this description, we can see that the display in Figure 10.6 clearly shows us that the two input values for Features "F1" and "F2" were 40 and −20 respectively and places these visually within the universe of observed events. These input values have been assigned to the fuzzy input sets of "VERY HIGH" in Feature F1 and "VERY LOW" in Feature F2. As the points fall on the number line far away from any regions of fuzziness, there is membership in only a single fuzzy set for each UOD.

The goal in this display is to associate the rule set with underlying fuzzy set membership functions. For this reason, the greatest visual draw in the display are the membership function boundaries, which indicate to the user the portion of the UOD associated with the fired rules. Secondary to these are the actual input values, as when using MME and fuzzy membership aggregation schemes, the distinct values from the input are preserved only in terms of the fuzzy set membership relations. This relationship is managed by the relative amount of ink devoted to the membership function versus that for the representation of the input values.

The remainder of the display is shown in muted tones in order to reduce clutter while maintaining a framework for the important information.

10.2.3 Unimodal, 2-Feature Covaried Classification With Low Confidence

As a contrasting test, let us classify the record $(-2, -20)$. This location will fall on the decision surface between the classes and will therefore demonstrate what happens when the certainty of the system is minimal in this very simple data domain.

For this point, the display shown in Figure 10.7 is presented to the user as the initial summary screen. This point falls very near the quantization boundary between the two classes. The PD algorithm would have assigned this point to CLASSB but here it is instead marked UNCERTAIN as no class had a positive τ value.

F1	F2	*Suggestion*	*Confidence*	*Conflict*	*# of Rules Fired*
−2.0,	−20.0	UNCERTAIN(CLASSB)	0.500	∞	7 of 54

Figure 10.7: Covaried Classification With Low Confidence Summary

As a further indication that there is no positive support for any class, the “Conflict” column in Figure 10.7 is marked as ∞. This shows the conservatism of the PD/FIS DSS relative to simple PD.

The overall confidence of this classification is marked as 0.5, as for this δ/τ location we have observed an equal number of correct and incorrect decisions.

All of the above factors combine to make it clear to the user that this suggestion is based on only the thinnest degree of discernment. A very important corollary of such an assertion is that *a confident decision cannot be made* based on this data, and that therefore in the context of a larger decision, this lack of confidence needs to be taken into account.

This will suggest to the decision maker that testing using another source of data may be prudent, as the DSS suggestion indicates that the particular data values recorded are not able to produce a confident decision. If an additional test or report is available to provide more insight, the decision may thereby be improved.

Decision Exploration: Drill-Down Into Rules

Figure 10.8 shows the rules upon which this decision is based. Note that the support for the both classes in Figure 10.8 have rule output values which are both positive and negative. This indicates that the rules triggered within the FIS capture the knowledge that this location is a boundary location, and further, that logical arguments exist which would assign it to either class. It is by means of the simple aggregation through the defuzzification operation that these sets of contradictory votes are turned into single scalar assertions ($\mathcal{A}_k$) summarized at the bottom of each column.

The overall $\mathcal{A}_k$ for both classes are negative, indicating that τ will be negative. This indicates that no classification will be performed, thus the report of the summary display in Figure 10.7.

To further amplify the point that no assertion is made due to complete lack of support, the conflict display is marked as ∞ and the accompanying summary text is set to “No Positive Assertions”.

	Assertion	*Implied When …*
		(F1 is Medium and F2 is Very Low)
classB	0.640	F1 is Medium
	0.723	F1 is Low
	−0.290	F2 is Very Low
	−0.675	F1 is Medium and F2 is Very Low
	−0.009	**Refutation for classB (Chosen Label Value)**
	0.500	*Confidence*
	∞	*No Positive Assertions*

		(F1 is Medium and F2 is Very Low)
classA	0.645	F2 is Very Low
	−0.445	F1 is Low
	−0.280	F1 is Medium
	−0.575	F1 is Medium and F2 is Very Low
	−0.750	F1 is Low and F2 is Very Low
	−0.183	Refutation for classA

Figure 10.8: Covaried Classification With Low Confidence Rule Set

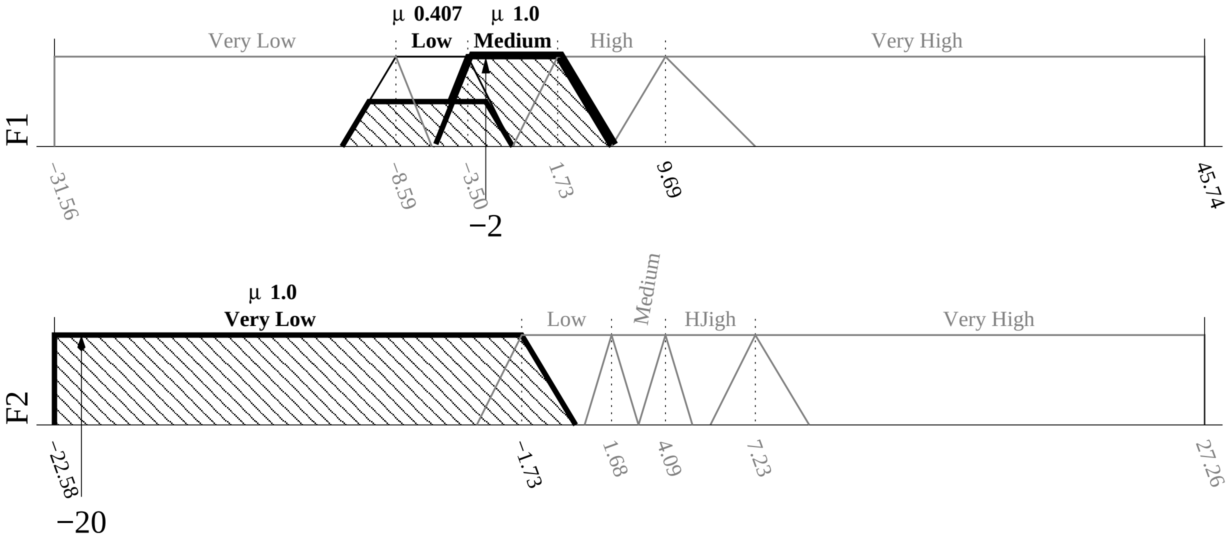


Figure 10.9: Membership Values For 2-Feature Simple Classification Example

Decision Exploration: Drill-Down Into Membership Functions

Examining the membership functions underlying the rule set of Figure 10.8 will bring up the display in Figure 10.9. Note that feature F1 is no longer assigned uniquely to a single fuzzy membership function. The point −2 now falls into the overlapping membership functions for the fuzzy sets MEDIUM and LOW, with memberships of 1.0 and 0.407 in these sets respectively. This in turn causes more rules to be activated and shown in Figure 10.8, as both the rules associated with the set MEDIUM and LOW may be used.

Comparing Figure 10.6 from the last membership function example to Figure 10.9 we see that the figure at once looks strikingly different, even though the positions of the lines defining the memberships have not changed. Confining the highlighting to changes in intensity allows various different parts of the figure to be brought to focus easily, while maintaining a common paradigm for all decisions made by a given rule base. Remembering that the membership functions are defined as a product of training on a given data set, this means that when applying this knowledge system within a specific application area an operator will always see the same membership functions for the same feature, regardless of the input values. It is only the highlighting of the memberships which will change from decision to decision.

This interactive drill down mechanism thereby allows the user to explore as much, or as little, of the decision space as is desired, while keeping the summary information terse and thus not confusing.

10.2.4 Heart Disease Data

The Hungarian heart disease data introduced in Chapter 8 will be used as a “real-world” example, in order to demonstrate how the system may be used by a medical professional. This data set has been chosen to support the discussion as it provides a complex, real-world data example in which the relationships between the features are not easily discovered by casual inspection by the reader, yet is from a domain in which the conclusions reached by the system can be examined in terms of the reader’s understanding about the factors underlying heart disease data. As this is a topic with significant discussion in both the popular press and the medical community, it is expected that the reader will have a passing familiarity with the symptoms and import, if not directly the measurement, of heart disease.

For these reasons, though it is assumed that only a trained user will be completely familiar with the fields which have been described in Table 8.3 (from page 92 in Chapter 8), persons with a casual interest in heart disease will still find some of these features to be familiar from other literature, and this will frame the following rule evaluations into a useful context.

For characterization purposes, the suggestions based on the heart disease database have one of two values: “Diseased” or “Normal.”

The δ/τ distribution surface histograms are shown in Figure 10.10, which shows that the separation between correct and incorrect is present, but not substantial. The fraction of characterizations made correctly by this system is 0.83 of the total. This in conjunction with the wide distribution of the error values indicates that there will seldom, if ever, be a value which will be asserted at unity (1.0) confidence as was seen in the simplistic 2-feature data just examined. Similarly, the confidence values will attain reasonable values as inspection of the error and correct confidence surfaces of Figure 10.10 shows relatively few errors occur over much of the region where δ/τ values have been observed.

10.2.5 Heart Disease Classification With High Confidence

In order to provide an example from a real database with as much clarity as possible, the record was located which had the highest degree of confidence in its suggestion of any record in the Hungarian heart disease data set. The values for this record are shown in Table 10.1.

When this record is considered by the FIS decision support system, we are presented with the results shown in Figure 10.11. This display is quite similar to the 2-feature display shown in Figure 10.4, with the addition of several more input feature values. The confidence value is again very high as the histogram cell associated with (δ, τ) pair $(0.679, 0.626)$ received 34 correct classifications and one error during training. This datum will be available to the user by highlighting the “Confidence” value of 0.971 in Figure 10.11.

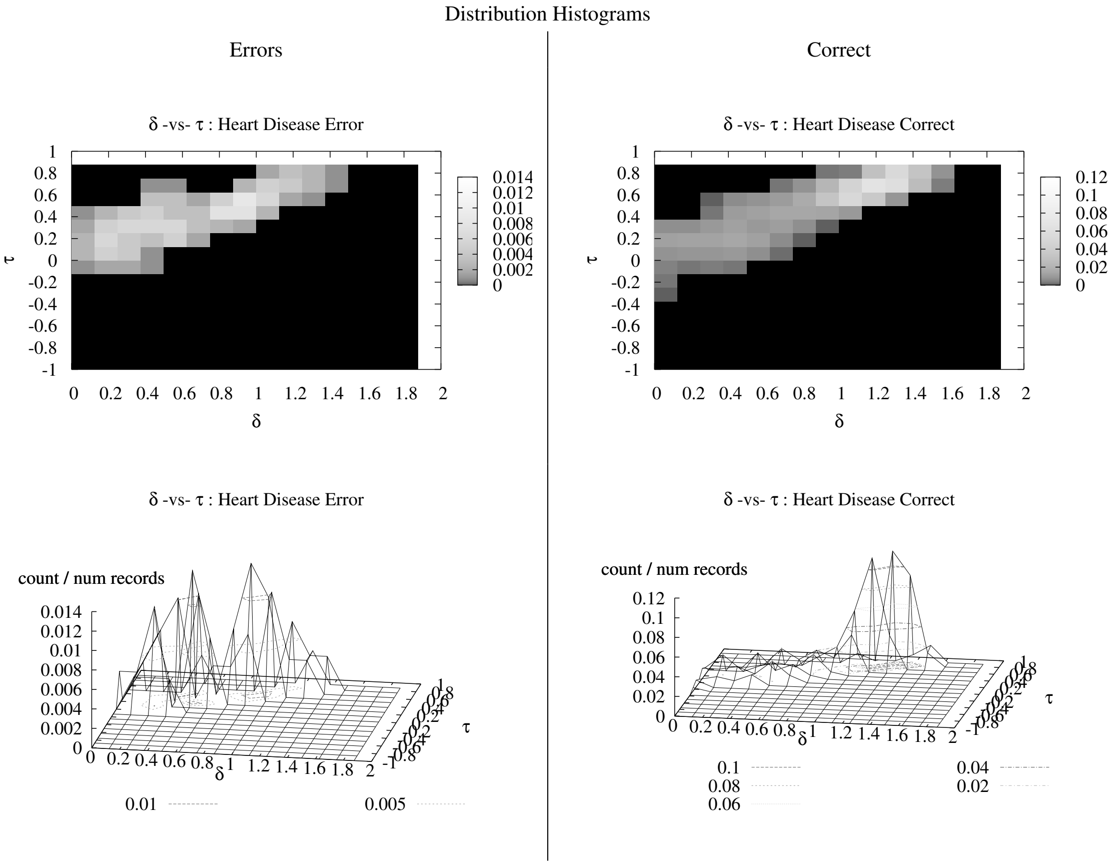


Figure 10.10: δ/τ Histograms from Heart Disease Data

Age	Sex	CP	TRestBps	Chol	FBS	RestECG	ThalAch	ExAng	OldPeak	Slope	CA	Thal	*Suggestion*	*Confidence*	*Conflict*	*# of Rules Fired*
48.0,	1,	4,	160.0,	193.0,	0,	0,	102.0,	1,	3.0,	2,	0,	3	Diseased	0.971	0.0	39 of 132

Figure 10.11: Heart Disease Classification High Confidence Summary Display

Table 10.1: Heart Disease Classification High Confidence Data Record

Feature	*Value*	
AGE	48.0	
SEX	1	(male)
CP	4	(asymptomatic pain)
TRESTBPS	160.0	mm Hg
CHOL	193.0	mg/dl
FBS	0	(fasting blood sugar ≤ 120 mg/dl)
RESTECG	0	(normal)
THALACH	102.0	bps
EXANG	1	(exercise induced)
OLDPEAK	3.0	
SLOPE	2	(downsloping)
CA	0	major vessels coloured by fluoroscopy
THAL	3	(normal)

Decision Exploration: Drill-Down Into Rules

Assuming again that the user wishes to drill down to the underlying rules, the display shown in Figure 10.12 is displayed. As there are 39 rules associated with this classification, it is neither feasible nor desirable to display them all. Instead, the top 10 rules are shown for each class, ranked in order of decreasing assertion value. The user is provided access to the complete list via another drill-down control.

The bracketed rules across the top of each section of Figure 10.12 show the compound rule formed from the highest WOE independent rule firings (this is the same compound rule described in relation to equation (9.8) in the $C_{\text{PD-Probabilistic}}$ discussion on page 104). While this form of rule construction is not used to generate a weighting, it is expected that this information will aid the user in understanding the characterization. The brackets in this statement indicate the FIS rules from which the compound has been composed. Each bracketed sub-clause therefore refers to one of the rules in the list below. The entire statement is shown in a muted tone in order to indicate that while it contains information useful to the user, it should not distract from the rule list below.

Note that in this example there is no conflict recorded, as the defuzzified assertion values are $\mathcal{A}_{\text{Diseased}} = 0.642$ and $\mathcal{A}_{\text{Normal}} = -0.518$ respectively, even though the maximum rule assertion is positive in both classes. The positive values for the Normal class shown in Figure 10.12 are combined with the other Normal assertions into the defuzzified centroid value shown at the bottom of each column.

The suggestion of the label Diseased for this record is therefore a conflict-free, high confidence classification as the highest assertion ($\tau = 0.642$) is both positive and distant from the second highest assertion ($\tau_2 = -0.518$ so therefore $\delta = 1.160$), leading to a very high confidence of $C_{\delta/\tau\ \text{Probability}} = 0.971$. As this is the highest-confidence record in the database, and given the common knowledge that males are at higher risk for heart disease, along with an understanding that the correlation with disease increases with age, it is unsurprising that this record would be a Male of at least Medium age.

	Assertion	*Implied When . . .*
		(Age is Medium High and Sex is Male and Slope is Medium High and CP is Asymptomatic and FBS is Medium) and (RestECG is Medium Low)
Diseased	0.950	Age is Medium High and Sex is Male and Slope is Medium High
	0.905	Age is Medium High and Sex is Male and ExAng is High
	0.852	Age is Medium High and Sex is Male and CP is Asymptomatic and FBS is Medium
	0.846	Age is Medium High and Slope is Medium High
	0.833	Age is Medium High and Sex is Male and CP is Asymptomatic
	0.800	Age is Medium High and Sex is Male and CP is Asymptomatic and RestECG is Medium Low
	0.791	Slope is Medium High
	0.787	ExAng is High
	0.786	Age is Medium High and ExAng is High
	0.730	Age is Medium High and CP is High
		10 more rules fired with confidence [0.675 . . . 0.292]
	0.642	**Support for Diseased (Chosen Label Value)**
	0.971	*Confidence*
	0.000	**No Assertion Conflict**

		(Age is Medium High and Chol is Low) and (Sex is Male and CP is Asymptomatic) and (OldPeak is High) and (ExAng is High)
classA	0.813	Age is Medium High and Chol is Low
	−0.128	Sex is Male
	−0.277	OldPeak is High
	−0.345	Age is Medium and ExAng is High
	−0.349	Age is Medium and Slope is Medium High
	−0.368	Age is Medium High and Sex is Male and FBS is Medium
	−0.398	Age is Medium High and Sex is Male
	−0.434	ThalAch is Very Low
	−0.482	Age is Medium High and CP is Asymptomatic
	−0.490	CP is Asymptomatic
		9 more rules fired with confidence [−0.507 · · · − 0.903]
	−0.518	Refutation for Normal

Figure 10.12: Heart Disease Classification High Confidence Rule Set

Decision Exploration: Drill-Down Into Membership Functions

If the user wishes to proceed to examine the inputs triggering the rule firings, the input membership functions will be displayed by user selection as discussed earlier. The display shown in Figure 10.13 indicates the mapping from the input value into the mixture of membership functions named in the rules for five input features of the heart disease database. Only the features AGE, SEX, CP, TRESTBPS and CHOL are shown, in order to allow the figure to be placed on a single page. The remaining features would appear below the ones shown here on a computer based system. If necessary, a scroll bar will allow the user to see all the data elements.

Note that in contrast to Figure 10.6, there are input values which trigger membership in more than one fuzzy set in Figure 10.13, specifically AGE and CHOL.

The age of the patient indicated in this record (*i.e.*, 48 years of age) falls within the central region of one of the membership functions while only catching the ramp at the end of the neighbouring function. The reading for serum cholesterol (CHOL) exhibits a similar relationship to its membership functions. As indicated, this means that the patient has "Low" cholesterol in the terminology of the FIS system, and may be considered "VERY LOW." In contrast, the measurement for TRESTBPS (resting blood pressure in mm Hg upon admission to the Emergency department) falls uniquely into the "VERY HIGH" category.

The measures for SEX and for CP are shown as crisp sets; this is because these fields are stored as nominal integer labels in the input data set. Representation of SEX with non-fuzzy membership makes sense as (except for the tiny fraction of the population exhibiting hermaphroditism) this is a physiologically crisp division. The CP value represents a nominal label applied by an examining physician; while there may be some vagueness associated with the label choice, information representing the degree of vagueness associated with this choice is not available to us, so a crisp nominal label is the clearest and most informative choice here. Remembering that the purpose of a DSS is to aid in decision making, it is clear that representation of vagueness in the data must be driven by that vagueness which we can accurately quantify and thus represent truly. Adding further vagueness measures will simply add more complexity to the system, making it more confusing and therefore less useful.

The use of a crisp set within the FIS simply prevents the construction of "ramps" and ensures that all possible associated memberships are equal to 1.

Using the architecture of the FIS, the linguistic labels {MALE, FEMALE} and {TYPICAL ANGINA, ATYPICAL ANGINA, NON-ANGINAL PAIN, ASYMPTOMATIC PAIN} have been loaded as the names of the sets in the FIS. This allows the input database to remain in its standard integer form, but supply the user with the appropriate label.

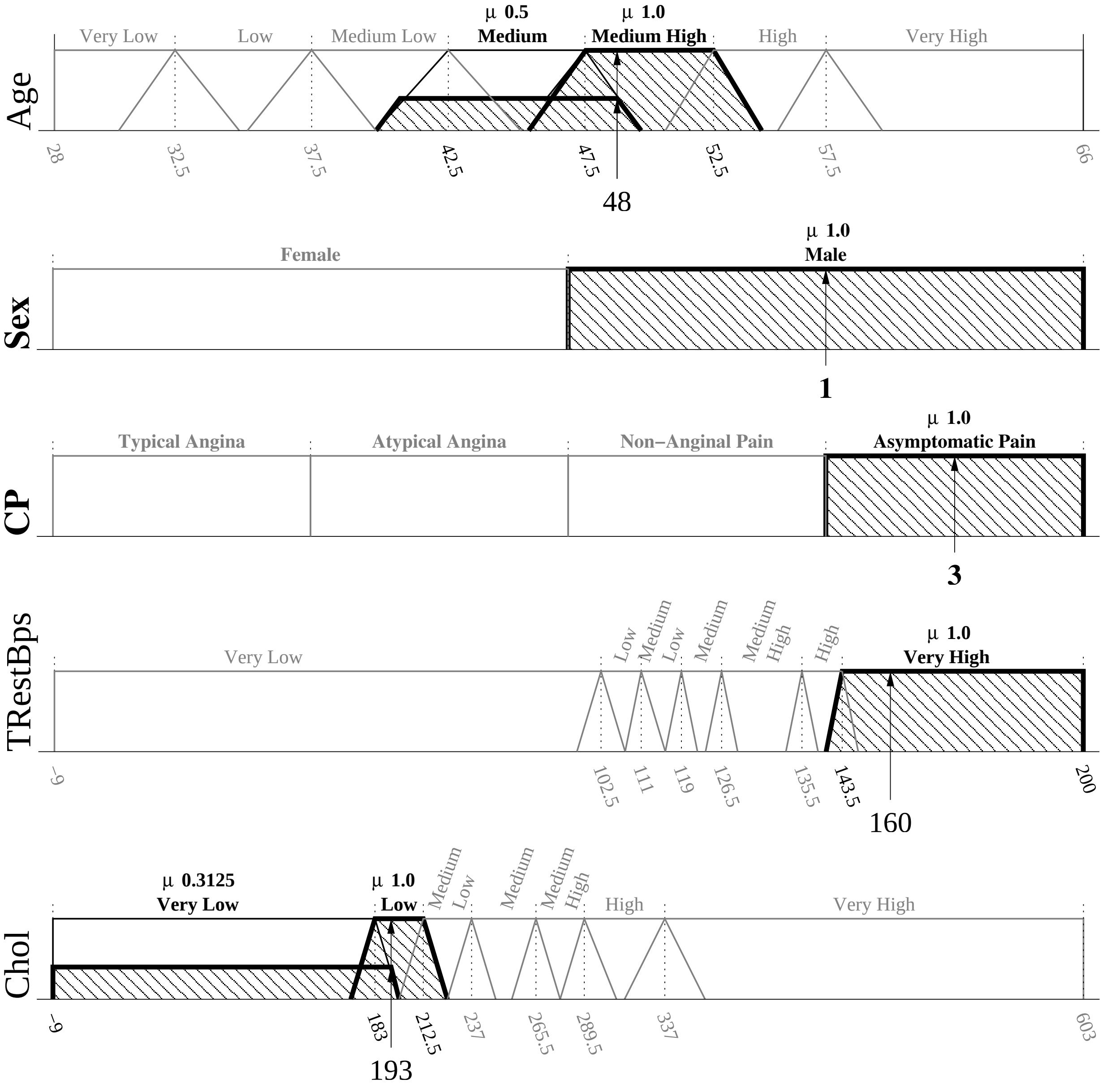


Figure 10.13: Heart Disease Classification High Confidence Membership Functions

Age	Sex	CP	TRestBps	Chol	FBS	RestECG	ThalAch	ExAng	OldPeak	Slope	CA	Thal	*Suggestion*	*Confidence*	*Conflict*	*# of Rules Fired*
42.0	1	3	160.0	147.0	0	0	146.0	0	2.25	2	1	3	Diseased	0.667	0.053	29 of 132

Figure 10.14: Heart Disease Classification Conflict Summary Display

Decision Exploration: Drill-Down Into Statistical Support

A further drill-down is available to the user based on individual rules, allowing inspection of the underlying relationships. For example, if the user wishes to inspect the rule "Sex is Male," the statistics on which this rule was generated are presented. In this case, this second order event (a Male patient in the training set who is Diseased) was observed 94 times out of a total of 212 Male patients, indicating a probability of 0.443 for an incidence of heart disease in Male patients. For Female patients in the training data set, the probability is 12/81 or 0.148.

This form of inspection is available for every rule which has been stored to define the FIS; that is, the complete set of rules (patterns) deemed to be of statistical significance by the PD pattern extraction algorithm.

The mechanism for initiating this inspection will be selection of (*i.e.*, clicking on) the label "Assertion" in a display such as Figure 10.12. This action will cause the assertion values to be replaced with the underlying occurrence numbers. Note also, that as the assertion values are based on FIS(Occurrence/-All) evaluation, these values are simply probabilities calculated through occurrence observations.

10.2.6 Heart Disease Classification With Conflict

The true utility of a decision support system lies in its ability to aid a user in understanding data which contains conflict or uncertainty. Let us therefore assume that the data record shown in Table 10.2 is processed. This record was selected by examining the database for records with conflict that still contained a high τ value.

The summary display of Figure 10.14 resulting from the processing of this record immediately flags the fact that there is some conflict in the decision involving this patient record, and that the confidence is reduced from the previous case as more errors were found relative to correct classifications in this δ/τ histogram cell. The increased visual contrast of the conflict measure allows a user to immediately

Table 10.2: Heart Disease Classification Conflict Data Record

Feature	*Value*	
Age	52.0	
Sex	0	(female)
CP	4	(asymptomatic)
TRestBPS	130.0	mm Hg
Chol	180.0	mg/dl
FBS	0	(fasting blood sugar ≤ 120 mg/dl)
RestECG	0	(normal)
ThalAch	140.0	bps
ExAng	1	(exercise induced)
OldPeak	1.5	
Slope	2	(downsloping)
CA	0	major vessels coloured by fluoroscopy
Thal	3	(normal)

distinguish between records of this type and those with no conflict.

Decision Exploration: Drill-Down Into Rules

In response to a drill-down selection, Figure 10.15 displays all the rules which have been fired. In this interesting record, it is instructive to examine all of the record values, rather than just the top 10 of each set, therefore the display in Figure 10.15 has been fully expanded.

Note the range of assertions in the defuzzification space for Normal which results in a low assertion of 0.053 even though some rules support this classification with values as high as 0.875. Defuzzified support for the Disease class is much higher as there are very few votes refuting this label, in contrast to the Normal class which has both strong support and refutation. The conflict apparent in the characterization as a Diseased patient's record thus has obvious roots in the rule base.

Whether a user of the decision support system would wish to use a record with such conflict is strongly tied to the application area. A user may proceed to make a decision incorporating both the characterization shown here and the new knowledge that there are strong conflicts relating to this record, based on an understanding of the relationship between conflict and reliability; conversely, as in the simple conflict example discussed previously, the user may decide that further analysis is required.

	Assertion	*Implied When . . .*
		(Age is Medium High and Slope is Medium High) and (OldPeak is Medium High) and (Slope is Medium High) and (ExAng is High) and (Sex is Female)
Diseased	0.846	Age is Medium High and Slope is Medium High
	0.813	OldPeak is Medium High
	0.791	Slope is Medium High
	0.787	ExAng is High
	0.786	Age is Medium High and ExAng is High
	0.730	Age is Medium High and CP is Asymptomatic
	0.675	CP is Asymptomatic
	0.391	Age is High and ExAng is High
	0.375	Age is High and Slope is Medium High
	0.340	Age is High and CP is Asymptomatic
	−0.590	Sex is Female
	0.626	**Support for Diseased (Chosen Label Value)**
	0.667	*Confidence*
	0.053	**Conflict**

		(Age is Medium High and Sex is Female and Thal is Low) and (CP is Asymptomatic) and (ExAng is High) and (Slope is Medium High) and (OldPeak is Medium High)
classA	0.875	Age is Medium High and Sex is Female and Thal is Low
	0.852	Sex is Female
	0.840	Age is Medium High and Sex is Female
	0.833	Age is Medium High and Sex is Female and FBS is Medium
	0.833	Age is Medium High and Sex is Female and CA is Low
	0.810	Age is Medium High and Sex is Female and RestECG is Medium Low
	0.580	Age is Medium High and Chol is Low
	0.375	Age is High and ThalAch is Medium
	−0.274	Age is High and Slope is Medium High
	−0.277	Age is High and CP is Asymptomatic
	−0.308	Age is High and ExAng is High
	−0.482	Age is Medium High and CP is Asymptomatic
	−0.490	CP is Asymptomatic
	−0.570	Age is Medium High and ExAng is High
	−0.666	ExAng is High
	−0.673	Slope is Medium High
	−0.706	OldPeak is Medium High
	−0.720	Age is Medium High and Slope is Medium High
	0.053	Support for Normal

Figure 10.15: Heart Disease Classification Conflict Rule Set

Explored Rules and Heart Disease

It is instructive to examine the rules themselves, as such relationships are displayed such as an assertion of NORMAL with 0.852 assertion support simply because the patient is female. Another unsurprising relationship triggered by this record is that of low cholesterol and NORMAL classification even while age is MEDIUM HIGH; this is in agreement with innumerable articles in the popular press.

Clear phrasing of these relationships will aid in understanding and acceptance of a system of this type by professionals in the area of application.

10.2.7 Interactive Exploration — Brushing and Selection

A further means of exploring the interaction between rules and input values may be made available through brushing.

When holding the mouse pointer over any rule in the list (for example, the list shown in Figure 10.12), the associated input membership functions for that rule will highlight (in this case, the appropriate membership functions shown in Figure 10.13).

If this rule is selected (clicked), the highlight will remain after the mouse pointer is moved away, until the rule is selected a second time, or another rule is selected.

In this manner the interplay between multiple rules and their input values can be shown.

Similarly, brushing an input membership function will highlight all of the rules in the associated list, allowing an exploration of the relationship from input values through rules.

10.2.8 "What If?" Decision Exploration

A stream of output values are produced as a decision is explored, such as that shown in Figure 10.16 in which the results of several "what if?" suggestions exploring the results of changing various input feature values are shown.

This activity is expected to be useful in the context of exploring the consequences of input value changes in the data space. Consider the case where a patient has gone to a clinic and is exploring their medical status with a physician with respect to heart disease. The PD/FIS system may have classified this patient as being at risk for disease.

In order to explore this with the patient, the physician may ask the system "what if — the patient reduced their cholesterol count?" by adjusting that input parameter. The PD/FIS system would then respond with the suggested characterization involving the new data.

F1	F2	*Suggestion*	*Confidence*	*Conflict*	*# of Rules Fired*
40.0,	−20.0	CLASSA	0.867	0.0	6 of 54
30.0,	20.0	CLASSA	0.867	0.0	6
10.0,	−20.0	CLASSA	0.841	0.0	6
10.0,	−10.0	CLASSA	0.658	0.028	7

Figure 10.16: Covaried Classification With High Confidence Summary - Exploration

Upon examining the rule base related to the new suggestion, the physician may note that the highest factor still supporting heart disease risk may be a low number of hours per week of exercise. The physician may then ask "what if — exercise hours per week were adjusted?", and again a new suggested characterization would be presented.

In this way, the decision maker can use the system to evaluate various possible courses of action, based on an interactive query–and–response work-flow involving the rule base and presented suggestions.

10.3 Discussion

The DSS presented here exhibits the main features required of a decision support system: exceptional transparency and reliable confidence.

10.3.1 Transparency

The transparency of the system is maintained through interactive inspection of the relationships between input data, rules fired and output classification values.

The user is directed through the chain of inference from the highest level (the suggestion and summary values) back through the rules and input data values to the underlying statistics framing the input data in terms of the training data set. At each level, the system relays the complexity of the decision made by summarizing the confidence, conflict and number of rules fired.

This functionality is critical in a DSS, as only through the presentation of complete transparency will a user be able to understand the reasoning by which a decision has been presented. In the context of a larger decision being made from several sources (of which the DSS is but one), total transparency is required in order that aspects of the decision involving a deep appreciation of the interplay between measured data values can be brought into the context of the final decision.

Without the ability to inspect, for instance, the effect that gender has on heart disease, it is impossible for a user of the system to be able to suggest a course of action involving any other parameter, such as cholesterol level, as this secondary attribute needs to be taken into account in a gender-specific way.

Similarly, if a patient who has a family history of heart disease is being examined, it is important to be able to correlate both gender and other factors into a larger history. This form of exploration must accessible and supported in order that a larger understanding of the import of the data record of a particular patient may be fully brought to light.

10.3.2 Confidence

The confidence in the system is calculated using a metric proven to have a good correlation with true system failure. This metric provides the user a real sense of how likely the system is to be suggesting an incorrect characterization as measured using the internal assertion weightings of the algorithm.

As seen in Chapter 9, this provides a very good estimate of the probability of failure. The evaluation of the system in Chapters 7 and 8 shows that the system performance is strong for both the tested synthetic and real data.

10.4 Conclusions

The DSS presented in this chapter captures the most important features required for confident decision analysis and exploration.

This chapter has provided an overview of the exploratory mechanism present in the FIS. Several mechanisms have been presented by which an investigative user may determine rationale supporting a presented characterization. A graphical display for the presentation of input value membership in the fuzzy sets driving the system has been described.

The overall statistical nature of the FIS system allows the presentation of the statistical metric underlying each rule used; simple weighted combinations of rule assertion values result in an assertion for the class overall. This simple relationship leaves itself open to inspection by means of iterative drill-down analysis.

The hierarchical structure of the data, as shown in Figure 10.1 naturally supports an interactive exploratory work-flow. The stages of inspection supported by the drill-down within the FIS correspond to the relative abstraction of the cognitive model the decision maker will employ. Branching shown in the hierarchy shows the amalgamation of information from different sources at each level. Considering the hierarchy in a bottom up formation, each branch can be considered a simplification and summarization of the data space, allowing high level decisions to be made based on the underlying data, but without needing to directly evaluate each fact.

The weights and the rules may both be inspected, and the previously mentioned confidence value can be used to weight the overall suggestion in terms of a larger decision making process.

The next obvious step in the design process is to evaluate this decision making architecture with real decision makers. This future work will be done under separate cover adapting the system to a specific application area, such as clinical electromyographic decision support.

Chapter 11

Conclusions

"The best thing for being sad," replied Merlyn, beginning to puff and blow, "is to learn something. That's the only thing that never fails. You may grow old and trembling in your anatomies, you may lie awake at night listening to the disorder of your veins, you may miss your only love, you may see the world about you devastated by evil lunatics, or know your honour trampled in the sewers of baser minds. There is only one thing for it then — to learn. Learn why the world wags and what wags it. That is the only thing which the mind can never exhaust, never alienate, never be tortured by, never fear or distrust, and never dream of regretting. Learning is the only thing for you."

— T.H. White, "The Once and Future King"

Evaluation of the PD/FIS system has shown several interesting strengths and weaknesses.

The system meets the goals required for a decision support system. Specifically, it provides suggestions to a decision making user with sufficiently high performance to be useful, coupling each decision with a characterization of the degree of confidence the suggestion can bear while providing a transparent, explorable, explainable framework describing how the decision was formed.

11.1 Transparency

The PD based FIS provides a type of transparency not common among decision support systems. It provides not only the transparency of a rule based system, but also allows inspection of the means by which the MME quanta were created as well as the assurance (backed up by occurrence based probabilities if desired) that the rules found have statistical significance.

The rules themselves are formed based on the ability to distinguish input data values from each other in order to provide a labelling. In contrast to systems such as BP, this allows a user to see several important aspects of the underlying rationale: exactly what parts of the current input data vector carry the greatest information; the path of inference supporting the suggested characterization; and the logic by which that path was formed.

This transparency in PD, along with the confidence measure, allows a drill-down based analysis to be performed supporting a user-driven decision exploration. Through such an interactive exploration, the complexity of the decision making is made manageable. A sufficiently simple summary can be presented to the user at the highest level while still allowing the underlying data to be available as context. This reduces the cognitive load decision makers will experience when using the system. The drill-down methodology allows the decision maker to inspect any part of the decision process about which they hold curiosity. This allows the system to be regarded as trustworthy, as a “white box” presentation is maintained.

11.2 Performance

Through evaluation of the PD/FIS system on a variety of continuous and mixed type data, it has been shown that the FIS performance is generally measurably lower than that of “natively continuous” classifiers such as BP. For decision support purposes, the transparency of the PD/FIS makes it the superior candidate.

When training data set is small, such as those seen in the thyroid and heart-disease data sets, the PD/FIS, and especially the FIS(OCCURRENCE/ALL) implementation, may have a measured performance equal to that of BP, as the scarcity of the data provides insufficient resources for BP training. For these reasons, although the performance of the PD/FIS system classification performance may be lower than that of other classifiers, the outstanding qualities as a means of supporting explainable decisions makes it preferable in this context.

Reviewing the PD/FIS systems tested, the performance of the FIS(OCCURRENCE/ALL) system is particularly interesting as it is significantly higher than that of the PD system when run on continuous and mixed-mode data. The addition of the fuzzy input membership functions along with the occurrence based rule weighting improves the PD system and allows classification of more input values even in complex data class distributions such as the spiral data set examined here.

This shows that the addition of fuzzy attributes and a re-thinking of the rule weightings provides a useful improvement over the raw PD system from which the FIS(Occurrence/All) algorithm is derived.

The performance of the FIS(Occurrence/All) classifier is therefore sufficiently high that, coupled with the confidence measure presented, a reliable DSS can be constructed. This is superior to any design using a system with slightly higher classification performance but lower transparency.

11.3 Confidence

A confidence measure has been provided which is a good indicator of true reliability. This has been evaluated over a series of data class distributions and found to be stable, but related to the amount of information available in the problem.

This confidence measure allows the user to incorporate a suggestion from the DSS into a larger decision context, providing a means of differentiating between occasions when the suggestion is trustworthy, and those when the probability of an erroneous suggestion has become high. This design allows the system performance to degrade gracefully as input values carry the inference into regions of the decision space which are poorly understood or have a large degree of conflict. Such grace is required in a DSS, as without a means of indicating the variability in confidence as a function of the input data and internal system state, the user cannot gain trust in the decisions made.

11.4 Decision Exploration

Decision exploration describes the means by which a user will learn about a decision space and thereby come to understand and trust a decision support system. The FIS(Occurrence/All) based DSS described here provides a "drill-down" based exploration metaphor which accurately represents the hierarchy of the cognitive model, allowing a decision maker to explore the problem space in the context of a suggestion being made on a set of input values.

This form of exploration allows the user both to relate a proposed suggestion to other possible labellings and to determine how a suggestion is supported or refuted. By using both positive and negative logic rules, the conflicting support for multiple classes is clearly shown; through the use of fuzzy inference methodology, this conflict is summarized though simple suggestions with accompanying confidence values.

This provides a user with a trustworthy, transparent system which can provide characterization based suggestions across a variety of data domains.

11.5 Quantization Costs

The FIS adaptation of the PD system reduces the quantization cost in the evaluated problems by recasting the crisp quantization of the MME bins into fuzzy input membership functions (*i.e.*, the addition of ramps). A further investigation into the performance of a class-dependent quantization scheme may prove fruitful, as the quantization bin bounds are not in any way driven by the optimal decision surface in the system described here.

The BP based classifiers provide an interesting upper bound describing how high the cost of quantization is; a further evaluation relative to these classifiers will allow more discussion on the recovery of this cost.

11.6 Future Work

Several major directions are possible as further goals of the research here.

A major remaining aim includes the adaptation to a particular application area, specifically characterization of the disease relationship of motor unit potentials within an electromyographic domain. Beyond this direction, the following research topics immediately are accessible as extensions of this work:

Class Dependent Quantization: Much of the quantization discussion during this work has shown that the decision surfaces of classifiers such as MICD are not reflected in the bin boundaries of MME. The use of a class-dependent quantization scheme would allow the exploration of the benefits of driving the bin boundaries from the observed inter-class bounds, independently by feature.

Discrete and Mixed Mode Performance Analysis: The PD(OCCURRENCE/ALL) and FIS(OCCURRENCE/ALL) algorithms have shown a marked improvement in classification performance over the PD algorithm in continuous-valued data. A further topic of research will cover the performance analysis of these algorithms on discrete data to determine the relative utility of WOE and occurrence weighting with these data landscapes.

Multi-Part Classification: Of particular interest in electromyographic characterization is the production of a composite suggestion from the analysis of a set of related input vectors. Such a decision could be built upon the system described here by running the PD/FIS system for each input vector and combining the results together, essentially by producing an overall suggestion by a collection of individual ones, in a way similar to that by which rule firings are combined to achieve an overall assertion in the system described here.

Calculation of the confidence for such a scheme would be required; at this point, this remains an open problem, albeit one which could be addressed using techniques adapted from those shown here.

Analysis of Noise Due to Missing Data: While the PD system has been designed to function in cases of missing data values, an authoritative analysis of the effects of missing data values within the PD/FIS system remains to be done. In particular, the effects of the increase in missing data fields on the quality of the analysis is work which demands attention.

Calculation of a Fuzzy Confidence Value: While the performance of the confidence value suggested here correlates highly with measured reliability, an interesting topic of analysis would be the evaluation of a fuzzy prediction of confidence, based on δ/τ or other internal values, and a comparison of these results with those of the $C_{\delta/\tau \text{ Probability}}$ based confidence measure.

"Multi-Bin" Fuzzy Input Set Definitions: The discussion of the fuzzy input set adaptations in this work have concentrated on using each MME bin as the basis for a fuzzy input set. It would be equally possible to join multiple MME bins together and use a rule calculated on the joined bin, allowing membership values to be computed based on a relative measure of association which may differ for each composite MME bin. This would produce fuzzy membership functions from a contiguous join of MME bins, and produce a lower number of rules. These rules may describe the major features of a database more succinctly than the many rules based on smaller divisions.

This approach would be very useful for forming characterizations of a data base in the form of a rule summary. This in turn could form the basis of a data mining system used to explore and identify major patterns within a data set.

Multi-Vote Classification: The combination of a suggested label and a confidence value raises the possibility of an automated means of collecting multiple votes into one composite decision, as mentioned in Chapter 9. Such a system could incorporate the suggestions from a number of weighted classifiers into an overall vote, increasing the likelihood of a correct labelling. The confidence measure just described could then be used as a weighting, allowing the construction of a collaborative voting system using reliability estimation, rather than one simply based on majority voting.

Multi-voting classifiers have been described in Wanas and Kamel (2002); Levitin (2002, 2003) and Montani *et al.* (2003), and have an ongoing popularity as a means of producing an overall system with a higher classification performance than that of any of the composite systems. While many such systems combine "votes" from each sub-classifier simply through majority voting, the ability to weight each vote by its estimated reliability provides an interesting means of letting a classifier "abstain" when confidence is low. Such a system is described in Cordella *et al.* (1999).

Clinical Electromyographic Characterization: The author's main interest as an immediate application of this work is the construction of a clinical DSS based on the techniques presented here. A combination of the data analysis and suggestion just discussed as well as domain knowledge of the electromyographic clinical examination will allow the work described here to be presented in the form of a diagnostic suggestion system for muscular disease. Such a system would allow the analysis of diagnostic suggestion clarity and accuracy as a spectrum of disease involvement is presented. The characteristic changes in input values due to the progression of a disease must be taken into account by a clinical user, and therefore a vital part of the adaptation of this system to clinical work is a discussion and analysis of how disease progression is apparent in the DSS suggestions based on this particular data domain.

Part IV

Appendices

Appendix A

Mathematical Notation

This appendix lists the meaning of the variables used in the equations in the paper including their point of introduction. Greek letters are arranged at the beginning of the table, followed by Roman letters and acronyms beginning with Roman letters.

A.1 Greek Letter Variables

δ the difference between the two best output votes (τ and τ_2) from the FIS. Introduced in equation (9.3) in Section 9.2.1.

δ_k is used in equation (5.10) as part of the discussion of the MICD classifier in Section 5.7.2.

θ index to the spiral of equation (5.8) in Section 5.4 on page 58.

κ the "strength" of the covaried data. Used in equation (5.2) to produce $\mathbf{Cov}_{ij}$. In this work, $\kappa = 0.6$.

μ when used as a variable, a mean. When used as a function, μ indicates the membership function of some fuzzy set.

ν_j the number of discretely observed values (*i.e.*, bin assignment IDs when considering continuous data) observed in column j.

ξ shape parameter used in equation (5.7) to control the degree of spread of the Log-Normal class distribution. Described in Section 5.2 on page 56.

π the circularity constant, $\pi = 3.14159265\ldots$

ρ scale or acceleration of the generating spiral for the spiral class distribution described in equation (5.8) in Section 5.4 on page 58.

σ standard deviation.

τ the best of the output votes $\mathcal{A}_k$ asserted by the FIS. Introduced in Section 9.2.1 and used throughout the discussion of reliability in Chapter 9.

A.2 Roman Letter Variables

$a_{\mathbf{x}_l^m k}$ the assertion produced by firing the rule $\mathbf{x}_l^m$ associated with class k.

$\mathcal{A}_k$ the defuzzified assertion produced in support or refutation of class k. Defined originally in section 4.3 in Chapter 4, on page 48.

$\mathbf{c}_i$ the class mean calculated for a unimodal distribution using equation (5.5) and then used to generate a separation value, s_i. Described on page 56.

$\mathbf{Cov}_{ij}$ a covariance matrix constructed to generate unimodal covaried data. Described on page 54.

$C_{\mathbf{MICD}}$ MICD based confidence, as described by equation (9.5).

$C_{\tau[0...1]}$ bounded normalized confidence based on examination of τ, defined in equation (9.5).

$C_{\delta/\tau\ \mathbf{Probability}}$ probability based confidence, introduced in equation (9.5).

$\mathcal{E}(\Theta)$ expectation operator; provides the "expected value of Θ".

$\mathcal{E}_{\mathbf{x}_l^m}$ the estimate of the number of input records expected at a given order $\mathbf{x}_l^m$. This value is defined in equation (3.9) on page 41.

e the natural constant, $e = 2.71828\ldots$

$e_{\mathbf{x}_l^m}$ the expected number of occurrences of event $\mathbf{x}_l^m$. The construction of this value is shown in equation (3.4). See also $o_{\mathbf{x}_l^m}$.

i a looping variable used for several purposes with local meaning only

j usually indicates the column index, *i.e.*, $j \in [1 \ldots M]$.

K the number of true classes (labels) to which the data may be assigned. Note that there is always an extra "UNCERTAIN" class also, not included in K. See also y_k and Y.

k an index into the number of labels, K.

$\mathcal{L}$ the list of rules used within the workings of the independent or "IND" fuzzy rule firing scheme outlined in Section 4.4.2.

$\mathcal{M}$ the list of highest order match rules used within the workings of the independent or "IND" fuzzy rule firing scheme outlined in Section 4.4.2.

M the number of input columns in the training and testing data set, excluding the label column.

N the number of rows in the training data set.

$\mathcal{N}(0, 1)$ indicates a (Gaussian) Normal distribution with zero mean, and unit standard deviation.

$o_{\mathbf{x}_l^m}$ the number of observed occurrences of event $\mathbf{x}_l^m$. See also $e_{\mathbf{x}_l^m}$.

$\mathbf{Pr}(\mathbf{x})$ the probability of event $\mathbf{x}$.

P the weighted performance measure described in equation (5.12) in Section 5.8.

Q the number of MME quantization intervals into which continuous data has been discretized.

q_j the number of MME quantization intervals into which continuous data in column j has been discretized. For the tests in this work, $q_j = Q \ \forall \ j \in [1 \dots M]$.

r_0 initial distance from the origin for the spiral equation (5.8) of Section 5.4 on page 58.

$r_{\mathbf{x}_l^m}$ the adjusted residual of the current input event. the calculated residual of the current input event. Defined in equation (3.2) on page 35. If $|r_{\mathbf{x}_l^m}|$ exceeds 1.96, the multi order event $\mathbf{x}_l^m$ defines a statistically significant pattern. See also $z_{\mathbf{x}_l^m}$.

s_i a separation value from $\mathbf{S}$, the list of separations. This is calculated in equation (5.6) on page 56.

$\mathcal{T}_k$ transform generated to colour unimodal $\mathcal{N}(0, 1)$ data with a supplied covariance. Defined on page 55 in equation (5.4).

$\mathcal{T}_{\mathbf{err}}$ training error introduced in Section 5.5.

$\mathcal{V}_{\mathbf{x}_l^m K}$ a single value produced by firing rule $\mathbf{x}_l^m$ in support or refutation of class K. This will be evaluated along with all other such values for a given class to produce an assertion $\mathcal{A}_k$.

V a vector of variance values used to generate unimodal covaried data.

$v_{\mathbf{x}_l^m}$ the variance of $z_{\mathbf{x}_l^m}$, defined in equation (3.3) on page 35, and used in the discussion of Bimodal data mode separation on page 57.

WOE the weight of evidence of an event, defined in equation (3.7) on page 37.

$\mathbf{x}_l^m$ an input event at some order m, $m \in [1 \dots M + 1]$, consisting of one or more input columns. In order to be considered as a pattern, it must also contain a label value. The index l indicates the ordinality in the set of of the order-m events.

$\mathbf{x}_l^\star$ the portion of $\mathbf{x}_l^m$ consisting only of the vector of input column data after the label column has been removed, that is,

$$\mathbf{x}_l^\star \cap Y = \mathbf{x}_l^m.$$

$\mathbf{x}_l^\oplus$ a composite event built from the $\mathbf{x}_l^\star$ portions of all rules firing which match a given input event $\mathbf{x}$. This value is used in the construction of conditional probability from weight of evidence, as described in equation (9.8) in Chapter 9.

$\boldsymbol{W_k}$**,** $\mathbf{w}_k$ **and** w_k are all used in equation (5.10) as part of the discussion of the MICD classifier in Section 5.7.2.

Y the output label column.

y_k the kth value of the label, $k \in [1 \dots K]$.

$Y{=}y_k$ a notation describing the considered assignment of label k ($k \in [1 \dots K]$) to the label column.

$z_{\mathbf{x}_l^m}$ the calculated residual of the current input event. Defined in equation (3.1) on page 35.

Appendix B

Derivation of PD Confidence

The following proof describing (9.8) was derived by Pino (2005), and is reproduced here for reference in the discussion of confidence measures within PD and the FIS.

A "composite input vector" is formed by the connection of the input portions of two patterns referring to distinct columns. These patterns have *conditional independence* as conditioned by the class label; that is:

$$\Pr(\mathbf{x}_a, \mathbf{x}_b | L) = \Pr(\mathbf{x}_a | L) \times \Pr(\mathbf{x}_b | L) \tag{B.1}$$

The result is used to produce conditional probabilities of class membership based on the composite WOE produced during the firing of PD rules, as described in equation (9.8) in Chapter 9.

Proof. Derivation of Confidence from WOE The proof begins by considering (3.7), which defines "weight of evidence" (WOE), and proceeds to a form representing the conditional probability of a given label ($Y{=}y_k$), given a composite input vector, $\mathbf{x}_l^{\oplus}$ constructed as the union of all rules matching a complete input pattern $\mathbf{x}_l^m$

$$\begin{aligned} \mathbf{x}_l^{\oplus} &= \bigcup \mathbf{x}_l^{\star} \\ \mathbf{x}_l^{\star} &\in \mathcal{M} \end{aligned} \tag{B.2}$$

where $\mathcal{M}$ is the list of all patterns selected to match the current input pattern using the algorithm described in Section 3.4.1.

The derivation begins

$$\text{WOE} = \log_\beta \frac{\Pr(\mathbf{x}_l^\oplus, Y{=}y_k)\Pr(Y{\neq}y_k)}{\Pr(Y{=}y_k)\Pr(\mathbf{x}_l^\oplus, Y{\neq}y_k)} \tag{B.3}$$

$$= \log_\beta \frac{\Pr(\mathbf{x}_l^\oplus, Y{=}y_k)\,(1 - \Pr(Y = y_k))}{\Pr(Y{=}y_k)\left[\Pr(\mathbf{x}_l^\oplus) - \Pr(\mathbf{x}_l^\oplus, Y{=}y_k)\right]} \tag{B.4}$$

Equation B.4 is the form used in Wang (1997, pp. 94).

Now letting $\Phi = \frac{(1-\Pr(Y=y_k))}{\Pr(Y=y_k)}$

$$= \log_\beta \frac{\Pr(\mathbf{x}_l^\oplus, Y{=}y_k)}{\left[\Pr(\mathbf{x}_l^\oplus) - \Pr(\mathbf{x}_l^\oplus, Y{\neq}y_k)\right]} \cdot \Phi \tag{B.5}$$

$$= \log_\beta \frac{1}{\frac{\Pr(\mathbf{x}_l^\oplus)}{\Pr(\mathbf{x}_l^\oplus, Y=y_k)} - 1} \cdot \Phi \tag{B.6}$$

but $\Pr(Y = y_k|\mathbf{x}_l^\oplus) = \frac{\Pr(\mathbf{x}_l^\oplus, Y=y_k)}{\Pr(\mathbf{x}_l^\oplus)}$

$$= \log_\beta \frac{1}{\frac{1}{\Pr(Y=y_k|\mathbf{x}_l^\oplus)} - 1} \cdot \Phi \tag{B.7}$$

$$= \log_\beta \frac{\Phi}{\frac{1}{\Pr(Y=y_k|\mathbf{x}_l^\oplus)} - 1} \tag{B.8}$$

Rearranging for $\Pr(Y = y_k|\mathbf{x}_l^\oplus)$, we get

$$\text{WOE} = \log_\beta \frac{\Phi}{\frac{1}{\Pr(Y=y_k|\mathbf{x}_l^\oplus)} - 1} \tag{B.9}$$

$$\beta^{\text{WOE}} = \frac{\Phi}{\frac{1}{\Pr(Y=y_k|\mathbf{x}_l^\oplus)} - 1} \cdot \Phi \tag{B.10}$$

Letting $\alpha = \Pr(Y = y_k|\mathbf{x}_l^{\oplus})$ and $\gamma = \beta^{\mathrm{WOE}}$, we continue

$$\gamma = \frac{\Phi}{\frac{1}{\alpha} - 1} \tag{B.11}$$

$$\frac{1}{\alpha} - 1 = \frac{\Phi}{\gamma} \tag{B.12}$$

$$\frac{1}{\alpha} = \frac{\Phi}{\gamma} + 1 = \frac{\Phi + \gamma}{\gamma} \tag{B.13}$$

$$\alpha = \frac{\gamma}{\Phi + \gamma} = \frac{1}{\frac{\Phi}{\gamma} + 1} \tag{B.14}$$

so as $\Phi = \frac{(1-\mathcal{P}(Y=y_k))}{\mathcal{P}(Y=y_k)}$, we $\therefore$ conclude

$$\Pr(Y = y_k|\mathbf{x}_l^{\oplus}) = \frac{1}{\frac{\Phi}{\beta^{\mathrm{WOE}}} + 1} \tag{B.15}$$

or

$$\Pr(Y = y_k|\mathbf{x}_l^{\oplus}) = \frac{1}{\left[\left(\frac{1}{\beta^{\mathrm{WOE}}}\right)\left(\frac{(1-\Pr(Y=y_k))}{\Pr(Y=y_k)}\right)\right] + 1} \quad \square \tag{B.16}$$

Note that in this work all logarithms are base 2 (*i.e.*, β=2).

Appendix C

Conditional Probabilities Derived From Synthetic Data

In order to discuss possible confidence measures, it is logical to refer back to the underlying probability distributions by which the data examined here were generated, as a conditional probability based on the underlying data scheme should exhibit some relationship with the final confidence.

C.1 Confidence and Conditional Probability

The conditional probability of assignment of a given point to each distribution gives us the "true" confidence of assignment, based on the maximum amount of information available by measuring the point location.

C.2 Conditional Probability As Calculated Using z-Scores

Considering the synthetic data discussed in Chapter 5, it is possible to translate data point locations in the n-dimensional spaces examined for performance measurement back into z-scores relative to the various generating means. If this is done, we can then generate conditional probability values from the known PDFs (probability density functions) that generated the data originally.

As shown in Figure C.1 the conditional probability of true association with each class at any given

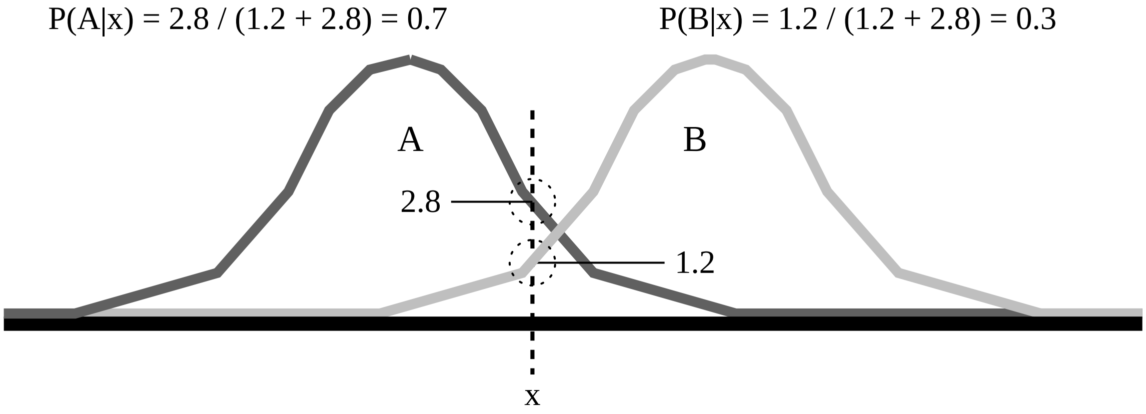

P(A|x) = 2.8 / (1.2 + 2.8) = 0.7
P(B|x) = 1.2 / (1.2 + 2.8) = 0.3
A
B
2.8
1.2
x

Figure C.1: Logically Constructing Conditional Probability from PDFs

z-score along the line defining two 1-dimensional distributions is simply

$$P(y_k|x) = \frac{P(y_k)}{\sum_{i=i}^{K} P(y_i)} \tag{C.1}$$

for any class k of the set of K possible classes. This is easily generalizable to n-dimensional cases by separately considering the Euclidean distance to the K means.

Performing this calculation allows us to produce conditional probability values for any function for which we know the *a priori* PDF definition. Surfaces showing the conditional probability values for two-dimensional versions of the covaried, bimodal and spiral data are shown in Figures C.2, C.3 and C.4, respectively.

By examining the covaried data in Figures C.2—C.4, it can be seen that as one moves away from the mode of class A, the lowest conditional probabilities are found as one approaches class B, as expected.

Points which are distant from both modes, such as the point at location $(-3, -3)$ at the left-hand side of the covaried data in Figure C.2 have the highest conditional probability. This also logically follows, as points distant from both A and B, while unlikely to occur at all (as shown by the low probabilities in their PDF values), have a strong association with the nearest class mode as a condition of their (rare) occurrence.

That is, given that a point $(-3, -3)$ actually occurs, the probability of its association with class A is very much higher than the probability of association with class B.

Similarly, conditional probability surfaces can be constructed for bimodal and for spiral data, as shown

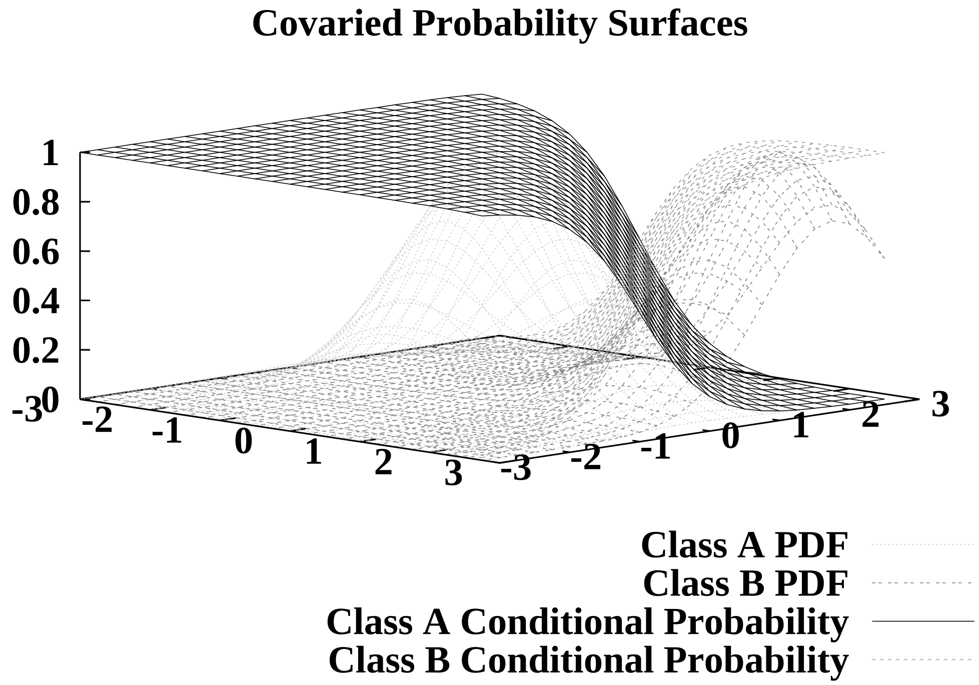


Figure C.2: Covaried Conditional Probability Surfaces

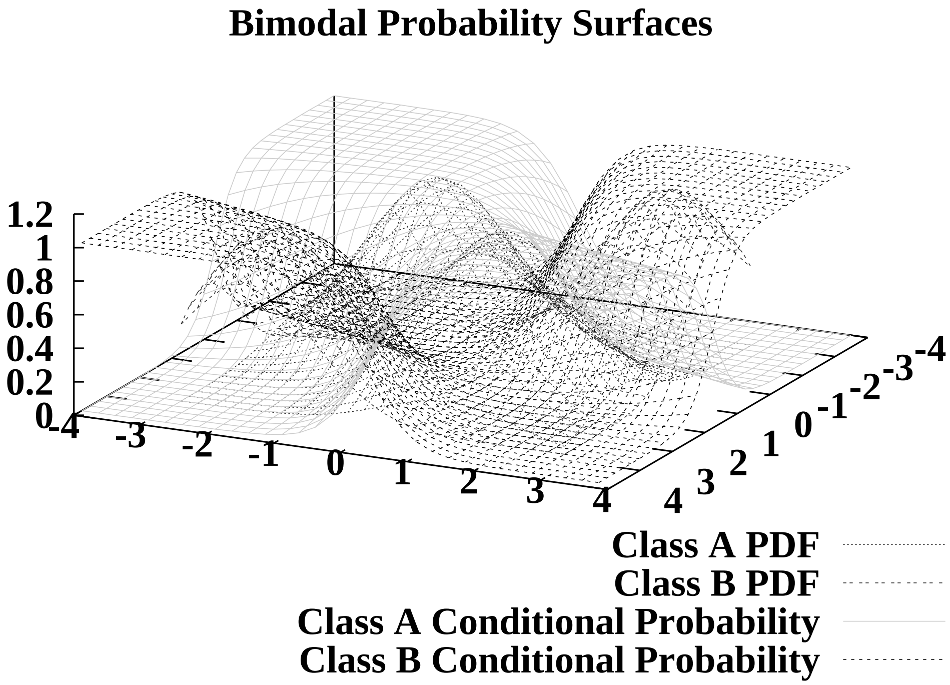


Figure C.3: Bimodal Conditional Probability Surfaces

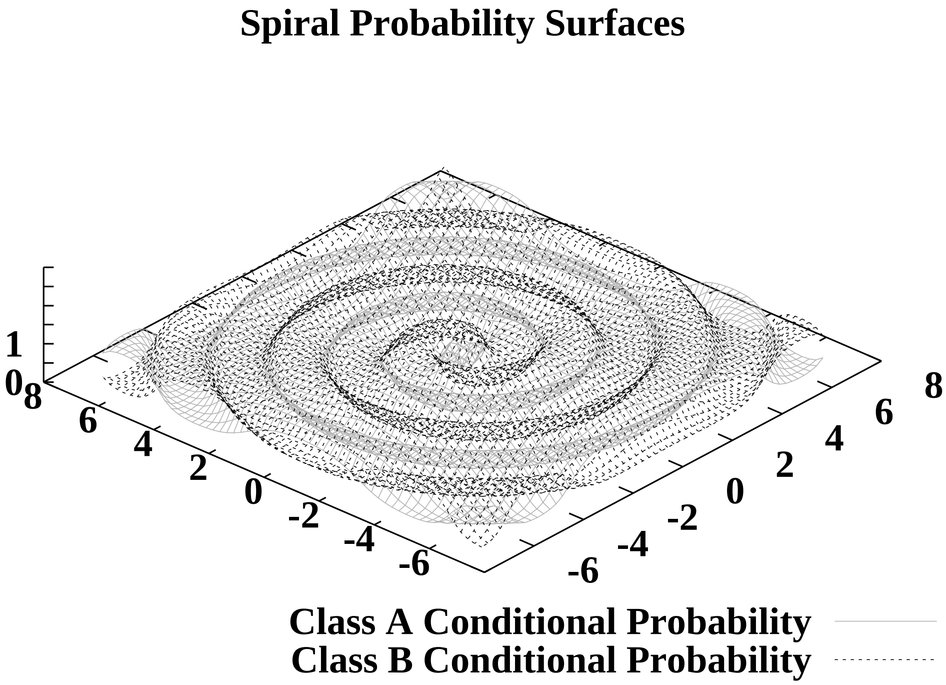


Figure C.4: Spiral Conditional Probability Surfaces

in Figures C.3 and C.4 respectively, simply by summing the probability of association with all modes and adapting (C.1) to include this extra sum, or

$$P(y_k|x) = \frac{P(y_k)}{\sum_{j=1}^{N_{\text{modes}}} \sum_{i=i}^{K} P(y_i)}. \tag{C.2}$$

C.3 Calculating z-Scores from Data Points

For the synthetic data analyzed in this work, we can calculate the z-score because the model by which the data was generated is available. For the covaried data, we simply need to reverse the effects of the applied covariance added by equation (5.4) from Chapter 5 and whiten the data according to the known mean vector and covariance matrix. Each conditional probability is then calculated separately relative to each class, much as is done in the MICD classification algorithm described earlier.

In the case of bimodal data, the process is much the same. The distance relative to each mode is, however, converted separately into a z-score before applying (C.2) and the per-class conditional probability is produced by summing the probabilities among all the modes for each class.

For spiral data, some care must be taken in order to ensure that both points in a positive and in a negative rotation around the origin are taken into account; other than this small concern the PDF can be generated by simply remembering that the $\mathcal{N}(0, 1)$ distribution centred at each point of the arm of the spiral extends around the origin at a fixed radius.

Using the described method, “true” confidence values in the form of conditional probability assignments to each class can be calculated for all the synthetic data considered in this work.

Appendix D

Further Tables Regarding Reliability Statistics

This appendix contains summary information calculated for the relationship between measured and expected confidence in support of the reliability discussion in Chapter 9.

The measures included here are mutual information, symmetric uncertainty and the interdependence redundancy measures, as well as the summaries for Spearmannranking.

The discussion here indicate some of the issues with using a non-rank based measure. Due to the problems discussed in this appendix, only the Spearmannrank distribution is used in the text.

For information regarding the effects of quantization on these measures, please see the accompanying discussion in Appendix E.

D.1 Symmetric Uncertainty

Symmetric uncertainty is a $[0 \ldots 1]$ bounded information based measure which describes the degree of correlation between two values.

The name “symmetric uncertainty” is slightly confusing because of the trend of the result reported, as the symmetric uncertainty value of independent data is 0, while the value reported for data with a perfect correlation is 1.

While this would suggest that a better name would imply the degree of “certainty,” the standard in the literature is to use the name “symmetric uncertainty.”

This usage seems to stem from the discussion of this measure in Press *et al.* (1992, pp. 634) where the main discussion is in terms of uncertainty coefficients, which already have this trend of 0 indicating independent data and a rising trend as data moves away from independence.

The calculation of symmetric uncertainty is performed using

$$SU[A, B] \equiv 2\left(\frac{H[a] + H[b] - H[a, b]}{H[a] + H[b]}\right) \tag{D.1}$$

which is defined in Press *et al.* (1992, pp. 634).

The symmetric uncertainty measure is an entropy-normalized version of the mutual information between the two values A and B.

D.2 Mutual Information

Mutual information ($MI[A; B]$) is defined to be

$$MI[A, B] \equiv H[a] + H[b] - H[a, b] \tag{D.2}$$

and may also be calculated using

$$MI[A, B] = \sum_{(a,b)\,\in\,A\times B} p(a, b) \log \frac{p(a, b)}{p(a)p(b)} \tag{D.3}$$

where a and b are probability mass functions as described in in Duda *et al.* (2001, pp. 632).

The entropy definition

$$H[A] = -\sum_{a} p(a) \log p(a) \tag{D.4}$$

formalized in Shannon (1948a,b) is extended to joint entropy (*e.g.*, Moon, 2000) by simply considering the joint $p(x, y)$ instead of a single probability mass function. This is calculated using

$$H[A, B] = -\sum_{(a,b)\,\in\,A\times B} p(a, b) \log p(a, b) \tag{D.5}$$

where $p(a, b)$ is the joint distribution of A and B.

D.3 Interdependency Redundancy

Interdependency redundancy of two random variables is calculated as the mutual information normalized by the joint entropy of the pair (from Wong and Ghahrarnan, 1975; Wong, Liu and Wang, 1976)

$$R[A,B] = \frac{MI[A,B]}{2\ N\ H[A,B]} \tag{D.6}$$

where N is the minimum number of occurrences of A or B.

D.4 Discussion

Upon evaluation, none of the entropy based statistics, namely mutual information, symmetric uncertainty, and the interdependency redundancy measure have significantly different trends to those in the simpler correlation calculations.

The mutual information, symmetric uncertainty and interdependency redundancy are all calculated over the observed confidence values after binning these values using an equal-range bin scheme with 8 bins per dimension.

An examination of the performance of equal-range and MME quantization schemes in conjunction with these statistics, as well as the effect of Q for these measures is explored in Appendix E, for readers interested in this background.

As can be seen by examining the performance of mutual information, interdependency redundancy and symmetric uncertainty, the saturation of the data at $(1, 1)$ leads to a very poor correlation. For these reasons, Spearmannrank correlation is used in the discussion in Chapter 9.

Table D.1: Mutual Information Comparison Summary: Equal Binning, Q=10

	C_{MICD}	$C_{\delta/\tau}$ Probability	$C_{\tau[0...1]}$	$C_{\text{PD-Probabilistic}}$
COVARIED				
0.125	0.260	0.274	0.352	0.149
0.250	0.274	0.280	0.352	0.170
0.500	0.206	0.224	0.296	0.182
1.000	0.236	0.254	0.268	0.257
2.000	0.178	0.136	0.150	0.157
4.000	0.067	0.073	0.086	0.188
8.000	0.000	0.000	0.001	0.000
μ	0.174	0.177	0.215	0.158
BIMODAL				
0.125	0.556	0.583	0.590	0.457
0.250	0.866	0.874	0.961	0.624
0.500	0.703	0.823	0.937	0.758
1.000	0.610	0.708	0.836	0.738
2.000	0.337	0.434	0.508	0.498
4.000	0.071	0.064	0.106	0.083
8.000	0.000	0.000	0.000	0.000
μ	0.449	0.498	0.563	0.451
SPIRAL				
0.125	0.023	0.023	0.028	0.033
0.250	0.037	0.018	0.026	0.026
0.500	0.031	0.037	0.062	0.038
0.750	0.033	0.023	0.076	0.040
1.000	0.040	0.021	0.124	0.076
μ	0.033	0.024	0.063	0.042

Table D.2: Symmetric Uncertainty Comparison Summary: Equal Binning, Q=10

	C_{MICD}	$C_{\delta/\tau\text{ Probability}}$	$C_{\tau[0...1]}$	$C_{\text{PD-Probabilistic}}$
Covaried				
0.125	0.134	0.143	0.189	0.077
0.250	0.145	0.155	0.203	0.094
0.500	0.116	0.133	0.158	0.091
1.000	0.150	0.144	0.155	0.179
2.000	0.122	0.122	0.103	0.181
4.000	0.000	0.000	0.000	0.000
μ	0.111	0.116	0.135	0.104
Bimodal				
0.125	0.000	0.000	0.000	0.000
0.250	0.000	0.000	0.000	0.000
0.500	0.000	0.000	0.000	0.000
1.000	0.000	0.000	0.000	0.000
2.000	0.000	0.000	0.000	0.000
4.000	0.000	0.000	0.000	0.000
μ	0.000	0.000	0.000	0.000
Spiral				
0.125	0.004	0.002	0.004	0.000
0.250	0.001	0.003	0.001	0.001
0.500	0.014	0.015	0.013	0.007
0.750	0.027	0.002	0.024	0.000
1.000	0.057	0.000	0.049	0.000
μ	0.021	0.004	0.018	0.001

Table D.3: Interdependency Redundancy Comparison Summary: Equal Binning, Q=10

	C_{MICD}	$C_{\delta/\tau\text{ Probability}}$	$C_{\tau[0...1]}$	$C_{\text{PD-Probabilistic}}$
COVARIED				
0.125	0.004	0.004	0.006	0.002
0.250	0.004	0.003	0.005	0.002
0.500	0.003	0.003	0.004	0.002
1.000	0.003	0.004	0.004	0.003
2.000	0.004	0.003	0.003	0.004
4.000	0.002	0.003	0.002	0.007
8.000	0.000	0.000	0.000	0.000
μ	0.003	0.003	0.003	0.003
BIMODAL				
0.125	0.012	0.012	0.012	0.009
0.250	0.015	0.016	0.015	0.012
0.500	0.015	0.016	0.015	0.012
1.000	0.014	0.014	0.015	0.012
2.000	0.009	0.011	0.009	0.010
4.000	0.003	0.004	0.003	0.003
8.000	0.000	0.000	0.000	0.000
μ	0.010	0.011	0.010	0.008
SPIRAL				
0.125	0.000	0.000	0.000	0.001
0.250	0.001	0.000	0.000	0.000
0.500	0.000	0.001	0.001	0.001
0.750	0.001	0.000	0.001	0.001
1.000	0.001	0.001	0.002	0.002
μ	0.001	0.000	0.001	0.001

Table D.4: Confidence Correlation Comparison Summary

	C_{MICD}	$C_{\delta/\tau\text{ Probability}}$	$C_{\tau[0...1]}$	$C_{\text{PD-Probabilistic}}$
Covaried				
0.125	0.496	0.525	0.593	0.378
0.250	0.495	0.528	0.593	0.398
0.500	0.423	0.453	0.549	0.399
1.000	0.438	0.461	0.499	0.424
2.000	0.333	0.357	0.373	0.404
4.000	0.166	0.117	0.140	0.113
8.000	0.001	−0.002	0.018	−0.001
μ	0.336	0.348	0.395	0.302
Bimodal				
0.125	0.743	0.486	0.715	0.656
0.250	0.715	0.406	0.701	0.628
0.500	0.665	0.375	0.672	0.585
1.000	0.583	0.270	0.579	0.511
2.000	0.426	0.383	0.389	0.354
4.000	0.205	0.099	0.132	0.094
8.000	—	—	—	—
μ	0.556	0.336	0.531	0.471
Spiral				
0.125	0.056	0.065	0.063	0.013
0.250	0.094	0.051	0.076	−0.041
0.500	0.119	0.100	0.125	0.066
0.750	0.181	0.127	0.199	0.108
1.000	0.202	0.109	0.259	0.190
μ	0.130	0.091	0.144	0.067

Appendix E

Statistical Measure Performance Under Quantization

This appendix summarizes the performance of the mutual information, symmetric uncertainty and the Spearmannranking statistics as evaluated under various quantization schemes for Gaussian and for Uniform data distributions.

Spearmannranking does not, of course, require binning, as the rank correlation will directly convert continuous values. It is included in this comparison simply to demonstrate what the effects of binning are in order to illuminate the behaviour of the other measures.

In each distribution, 1000 points are generated. The covariance in the distribution is adjusted from 0 (no covariance/independent data) to 1 (complete dependency of features).

E.1 Performance Tables

Tables E.1—E.5 display the numerical results for calculations using 1000 points for the statistics: mutual information, symmetric uncertainty and Spearmannranking using both equal width and MME binning.

Equal-width bin results are shown in Table E.1 for mutual information data, Table E.3 for symmetric uncertainty and Table E.2 for Spearmannranking.

Similar data binned using MME is displayed in Tables E.4, Tables E.6 and E.5 for mutual information symmetric uncertainly and Spearmannranking (respectively).

The mutual information statistic is described in equation (D.2) from Section D.2 in Appendix D; symmetric uncertainty is described in Section D.1 in equation (D.1). Both of these are entropy based measures.

Spearmannranking is defined in the main text in Chapter 9, in equation (9.10). Spearmannranking is simply a correlation based on relative ordinal position in the data set.

E.1.1 Entropy Statistics Versus SpearmannRanking

When comparing the results of the SpearmannRanking data in Tables E.2 and E.5, it is apparent that the Spearmannranking statistic benefits from being run on the raw data, as would be expected. When discussing Spearmannranking, no binning will be performed.

E.1.2 Choice of Binning Mechanism for Entropy Summary Statistics

Examination of the figures showing equal bin plots in Figures E.2 through E.7, we see that the equal-width bins preserves the overall shape of the underlying distribution while gathering occurrence counts regarding similar events.

A comparison with MME the based binning in Figures E.8 through E.13, demonstrates that the MME bins distort the effective shape of the distribution, leaving the distribution looking somewhat rounded as in Figure E.13.

This feature of MME, while useful for confidence generation in the PD and FIS algorithms will skew our calculation of confidence equivalency, and should thus be avoided for any discussion of these statistics.

In the discussion of confidence, we will therefore use equal-width bins in order to generate the summary statistics.

E.1.3 Choice of Q for Entropy Summary Statistics

Tables E.1 through E.6 show the effects of the number of quantization bins (Q) on the results.

As can be seen, too low a quantization (such as Q=2) results in very poor performance as there are too few distinct events in the space to adequately represent a trend. As seen by examining different correlations at Q=2 in Table E.3, there is no change in the statistic until perfect correlation is reached.

Conversely, a Q value which is too high incorrectly represents a great deal of information presence even in cases where there is independence, due to the irregularities which appear due to the relatively small N relative to the number of bins.

We must therefore balance Q and N if we are to use any of the non-ranked statistics, and choose as small a Q value as we can in order to get reasonable performance for our choice of N. As we will calculate our statistics over all jackknife runs, N will be 1000, and therefore, from these tables, it would seem that a choice of Q=8 is reasonable, as choosing Q=8 allows us to see the greatest degree of change across the symmetric uncertainty statistic in Table E.3, as well as the greatest change in mutual information as seen in Table E.1.

Table E.1: Mutual Information of Equally Binned Uniform Data (1000 Points)

	Correlation Coefficient					
	0.00	0.25	0.50	0.75	0.99	1.00
Q=2	0.000001	0.000001	0.000001	0.000001	0.000001	0.011408
Q=5	0.015014	0.141960	0.298019	0.528048	1.402589	2.006350
Q=8	0.039223	0.297239	0.388209	0.706805	1.839019	2.811652
Q=10	0.049157	0.354696	0.436370	0.752605	2.062620	3.174170
Q=15	0.159509	0.432796	0.567665	0.844415	2.356481	3.807815
Q=20	0.264482	0.518409	0.650064	0.942965	2.558243	4.247510

Table E.2: SpearmannRanking of Equally Binned Uniform Data (1000 Points)

	Correlation Coefficient					
	0.00	0.25	0.50	0.75	0.99	1.00
raw	−0.093548	0.250188	0.510996	0.777258	0.989883	1.000000
Q=2	−0.001001	−0.001001	−0.001001	−0.001001	−0.001001	1.000000
Q=5	−0.094267	0.204093	0.477265	0.711144	0.949343	1.000000
Q=8	−0.072047	0.231281	0.482814	0.762818	0.969068	1.000000
Q=10	−0.085259	0.247671	0.498017	0.769129	0.977148	1.000000
Q=15	−0.088046	0.239236	0.508771	0.774676	0.983600	1.000000
Q=20	−0.095187	0.242767	0.510054	0.772848	0.987102	1.000000

Table E.3: Symmetric Uncertainty of Equally Binned Uniform Data (1000 Points)

	Correlation Coefficient					
	0.00	0.25	0.50	0.75	0.99	1.00
Q=2	0.000127	0.000127	0.000127	0.000127	0.000127	1.000000
Q=5	0.007478	0.077203	0.160231	0.275472	0.699178	1.000000
Q=8	0.013954	0.114255	0.147840	0.263865	0.655911	1.000000
Q=10	0.015484	0.119853	0.146094	0.248715	0.651879	1.000000
Q=15	0.041904	0.121071	0.157596	0.231302	0.620990	1.000000
Q=20	0.062282	0.129200	0.160865	0.230623	0.606215	1.000000

Table E.4: Mutual Information of MME Binned Uniform Data (1000 Points)

	Correlation Coefficient					
	0.00	0.25	0.50	0.75	0.99	1.00
Q=2	0.005096	0.017626	0.116276	0.298526	0.757705	0.999997
Q=5	0.015979	0.175338	0.296970	0.621629	1.575068	2.321921
Q=8	0.035495	0.279476	0.416989	0.703978	1.960117	2.999988
Q=10	0.069933	0.325090	0.458489	0.754228	2.142002	3.321914
Q=15	0.149346	0.446866	0.550683	0.889301	2.475754	3.906105
Q=20	0.285521	0.605000	0.748973	0.986812	2.542354	4.321899

Table E.5: SpearmannRanking of MME Binned Uniform Data (1000 Points)

	Correlation Coefficient					
	0.00	0.25	0.50	0.75	0.99	1.00
raw	−0.093548	0.250188	0.510996	0.777258	0.989883	1.000000
Q=2	−0.084004	0.155997	0.395998	0.619998	0.920000	1.000000
Q=5	−0.088480	0.255892	0.487579	0.767038	0.959497	1.000000
Q=8	−0.086795	0.241088	0.509401	0.769164	0.974470	1.000000
Q=10	−0.089513	0.245647	0.505015	0.772254	0.979996	1.000000
Q=15	−0.093930	0.253125	0.509376	0.779506	0.986811	1.000000
Q=20	−0.092472	0.251797	0.510864	0.778500	0.987335	1.000000

Table E.6: Symmetric Uncertainty of MME Binned Uniform Data (1000 Points)

	Correlation Coefficient					
	0.00	0.25	0.50	0.75	0.99	1.00
Q=2	0.005096	0.017626	0.116277	0.298527	0.757707	1.000000
Q=5	0.006882	0.075514	0.127898	0.267722	0.678347	1.000000
Q=8	0.011832	0.093159	0.138997	0.234660	0.653375	1.000000
Q=10	0.021052	0.097862	0.138020	0.227046	0.644810	1.000000
Q=15	0.038234	0.114402	0.140980	0.227669	0.633817	1.000000
Q=20	0.066064	0.139985	0.173297	0.228328	0.588249	1.000000

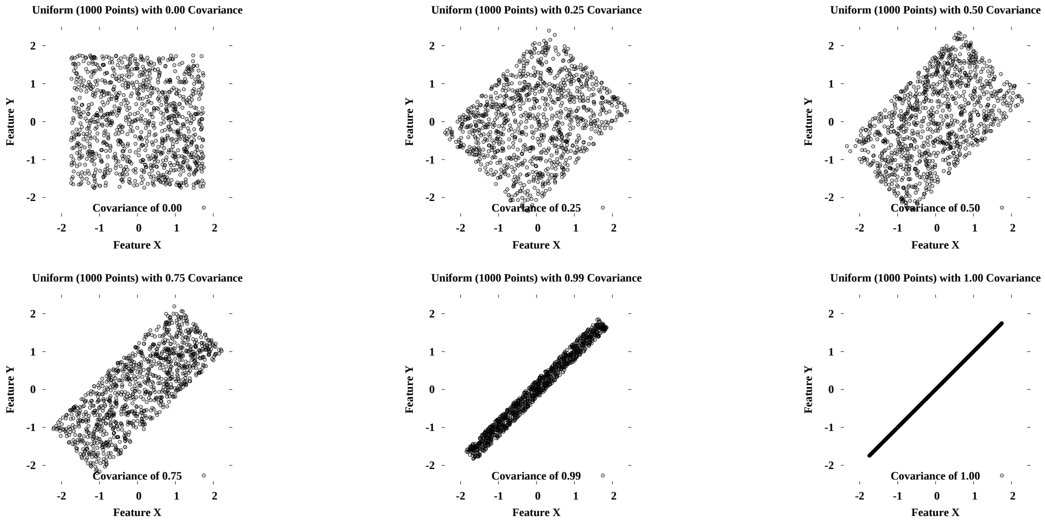


Figure E.1: Raw Uniform Data (1000 Points)

E.2 Quantization Figures

The remainder of this appendix contains figures showing the distribution of the points used for the calculations in Tables E.1 through E.6.

Figure E.1 displays the raw points, while Figures E.2 through E.13 show the effects of transforming the raw points through various types of binning strategies.

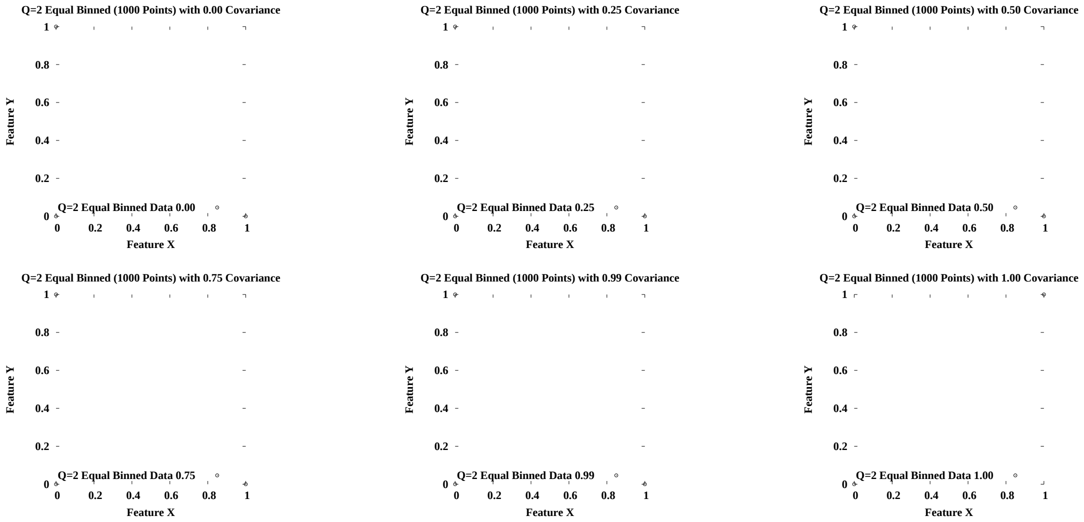


Figure E.2: Q=2 Equally Binned Uniform Data (1000 Points)

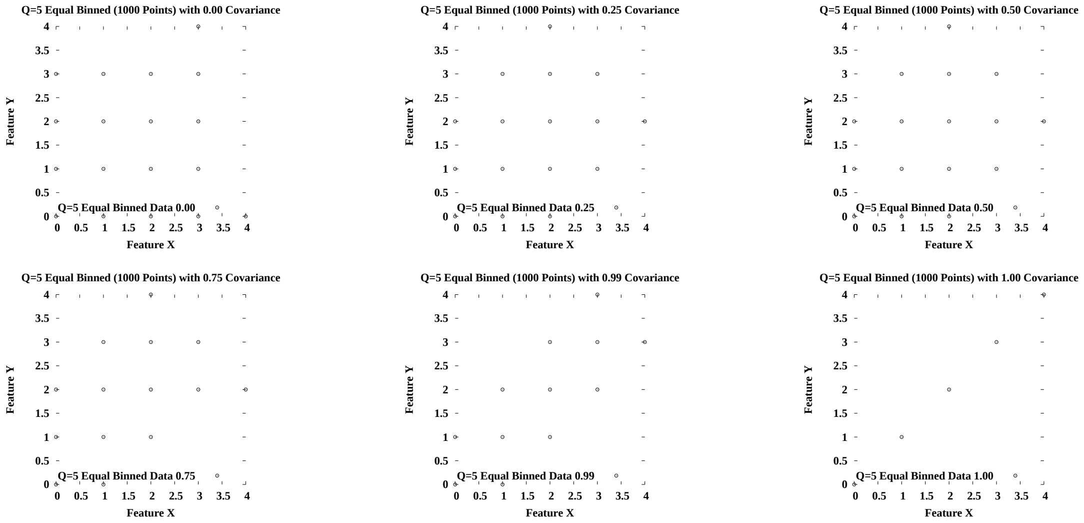


Figure E.3: Q=5 Equally Binned Uniform Data (1000 Points)

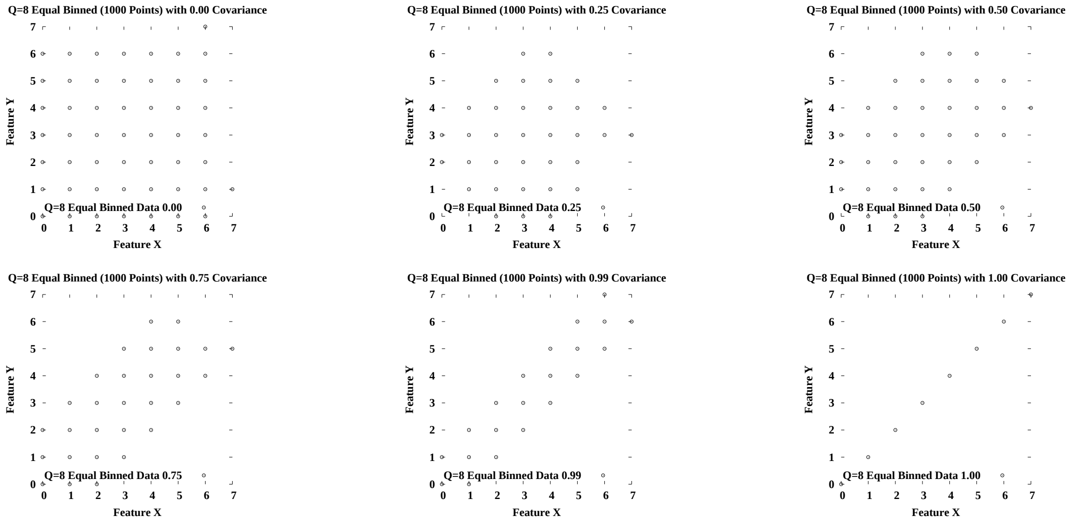


Figure E.4: Q=8 Equally Binned Uniform Data (1000 Points)

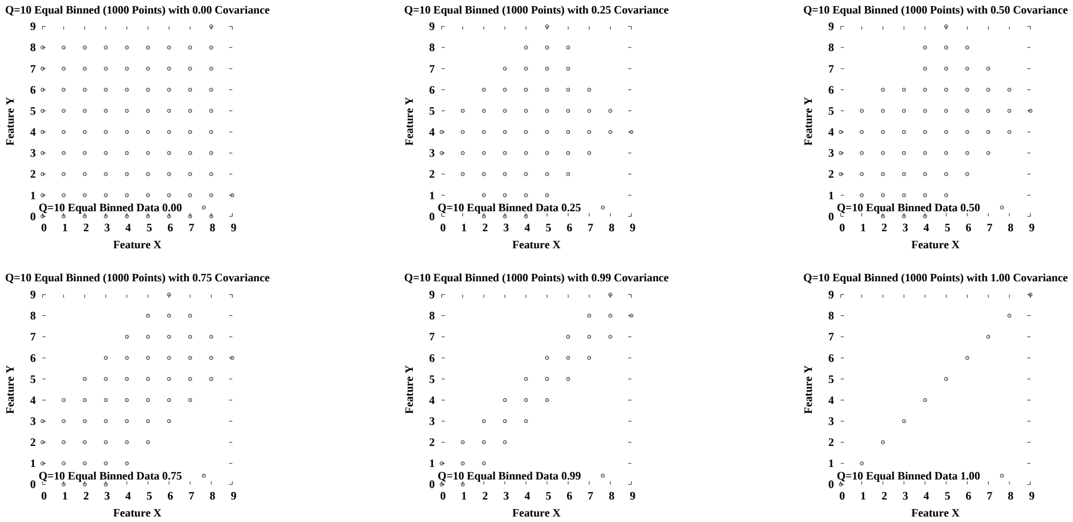


Figure E.5: Q=10 Equally Binned Uniform Data (1000 Points)

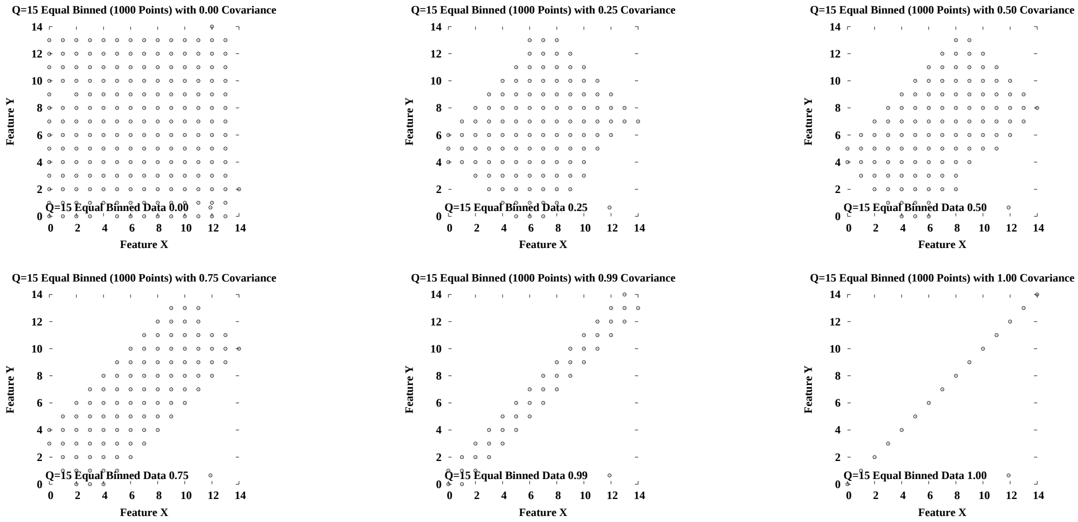


Figure E.6: Q=15 Equally Binned Uniform Data (1000 Points)

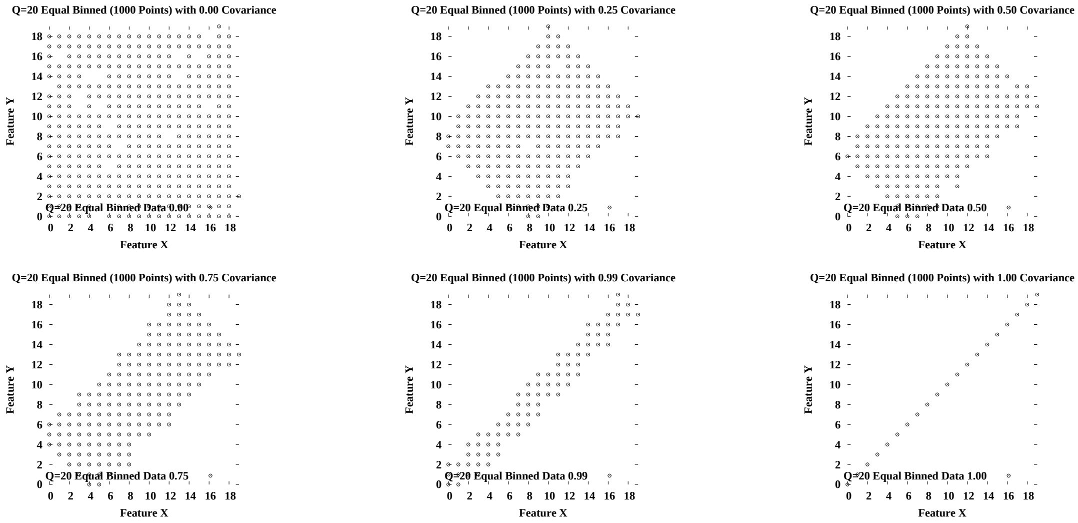
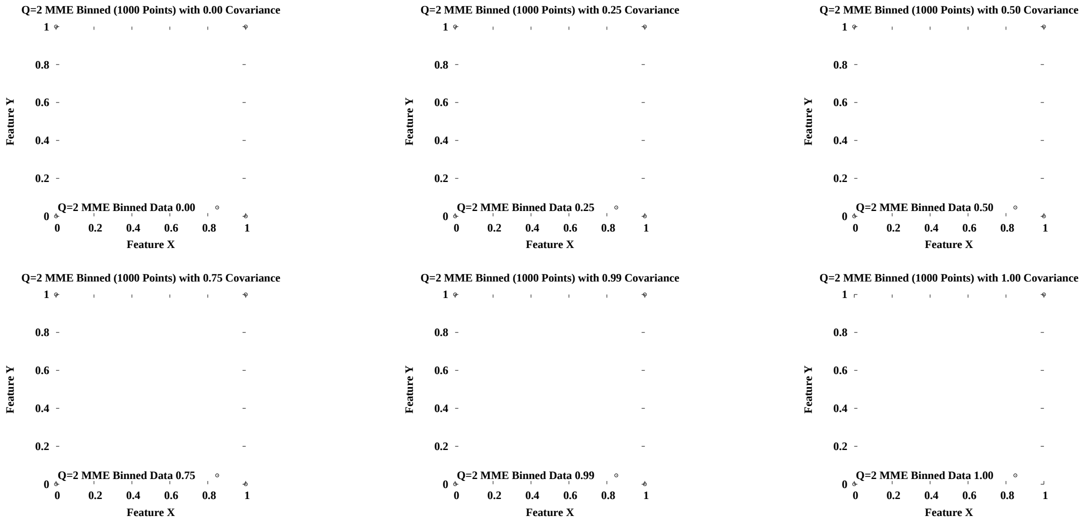


Figure E.7: Q=20 Equally Binned Uniform Data (1000 Points)


Q=2 MME Binned (1000 Points) with 0.00 Covariance

Q=2 MME Binned (1000 Points) with 0.25 Covariance

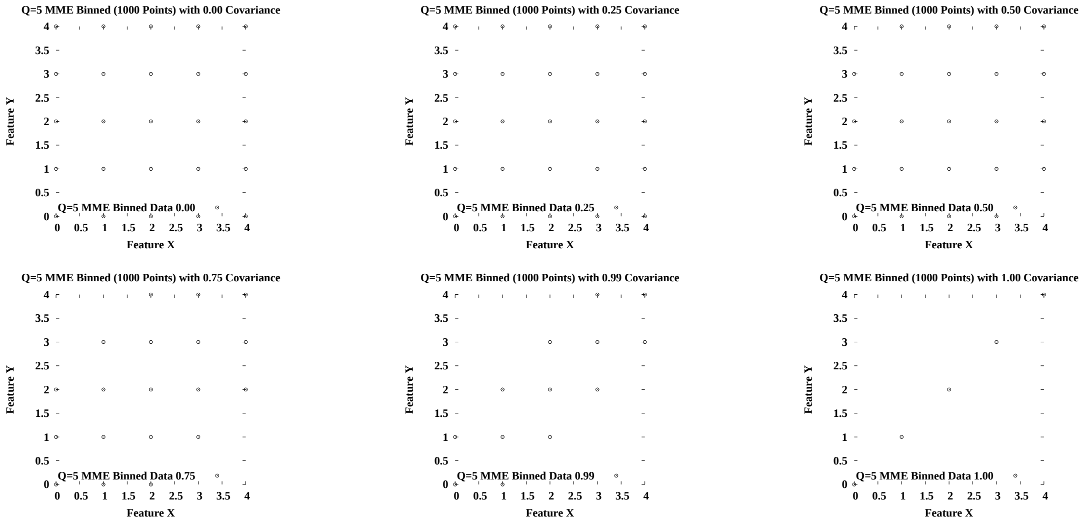
Q=2 MME Binned (1000 Points) with 0.50 Covariance

Q=2 MME Binned (1000 Points) with 0.75 Covariance

Q=2 MME Binned (1000 Points) with 0.99 Covariance

Q=2 MME Binned (1000 Points) with 1.00 Covariance

Figure E.8: Q=2 MME Binned Uniform Data (1000 Points)


Q=5 MME Binned (1000 Points) with 0.00 Covariance

Q=5 MME Binned (1000 Points) with 0.25 Covariance

Q=5 MME Binned (1000 Points) with 0.50 Covariance

Q=5 MME Binned (1000 Points) with 0.75 Covariance

Q=5 MME Binned (1000 Points) with 0.99 Covariance

Q=5 MME Binned (1000 Points) with 1.00 Covariance

Figure E.9: Q=5 MME Binned Uniform Data (1000 Points)

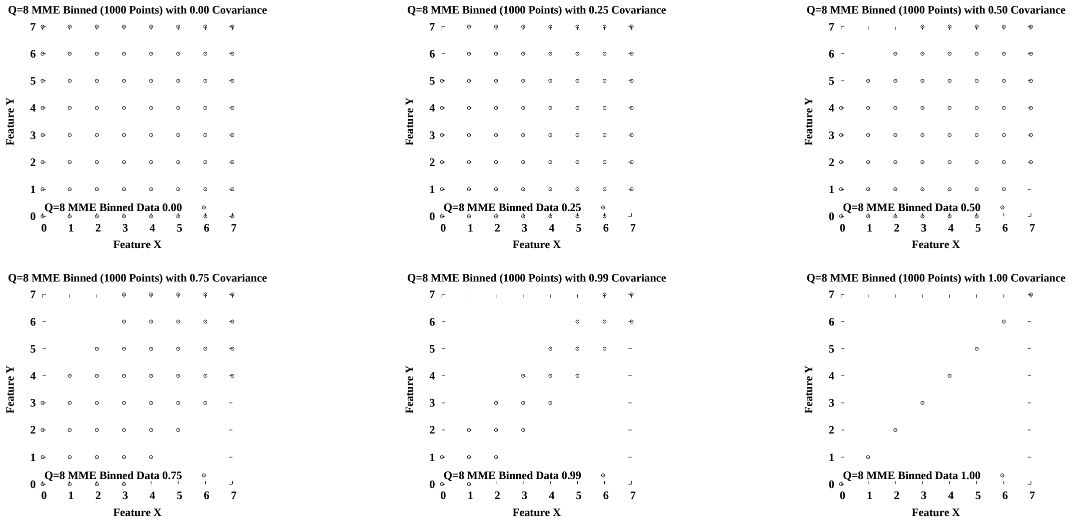


Figure E.10: Q=8 MME Binned Uniform Data (1000 Points)

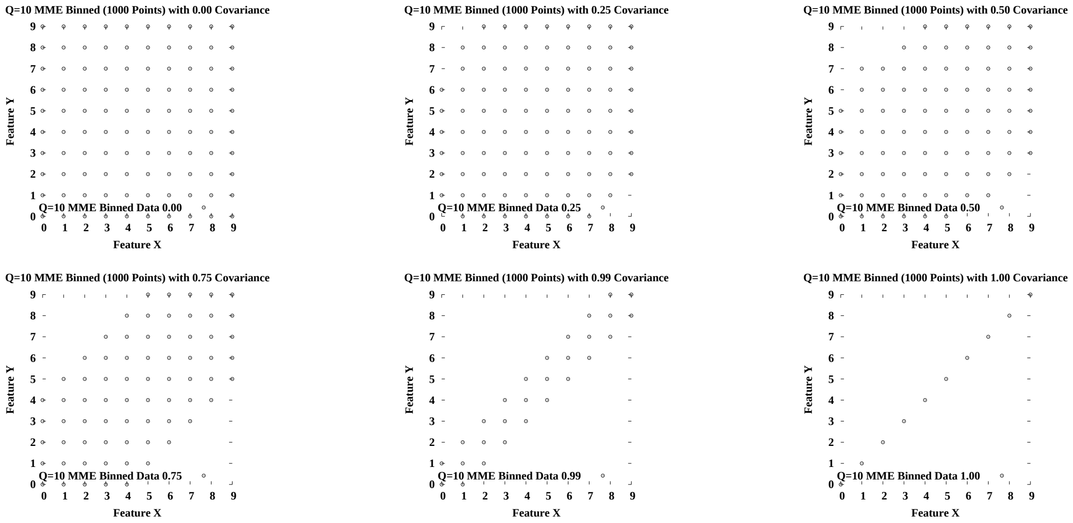


Figure E.11: Q=10 MME Binned Uniform Data (1000 Points)

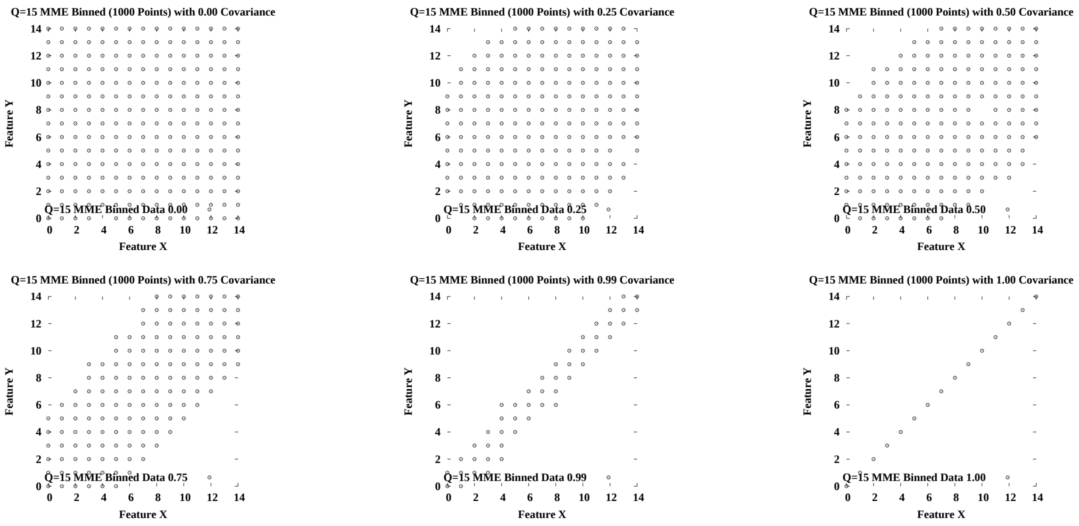


Figure E.12: Q=15 MME Binned Uniform Data (1000 Points)

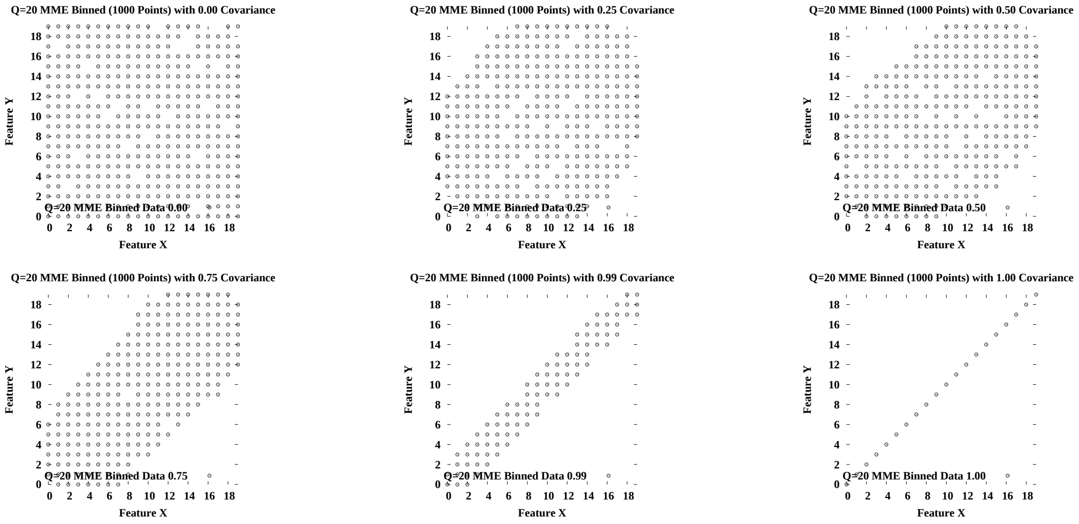


Figure E.13: Q=20 MME Binned Uniform Data (1000 Points)

Bibliography

Abu-Hanna, A. and N. de Keizer. Integrating classification treest with local lgistic regression in Intensive Care prognosis. *Artificial Intelligence In Medicine*, 29:5–23, 2003.

Adelman, L. *Evaluating Decision Support and Expert Systems*. Wiley Series in Systems Engineering. John Wiley & Sons, 1992.

Aha, D. W., D. Kibler and M. K. Albert. Instance-based learning algorithms. *Machine Learning*, 6:37–66, 1991.

Anderson, E., Z. Bai *et al.* *LAPACK Users' Guide*. Society for Industrial and Applied Mathematics (SIAM), 3rd edition, 1999. ISBN 0-89871-447-8. Software Library Available Online.
URL `http://www.netlib.org/lapack/`

Anzai, Y. *Pattern Recognition and Machine Learning*. Academic Press Inc., San Diego, 1989.

Arbib, M. A., editor. *Handbook of Brain Theory and Neural Networks*. MIT Press, Cambridge, MA, June 1995. ISBN 0-262-01148-4.

Barlow, R. E., C. A. Clarotti and F. Spizzichino, editors. *Reliability and Decision Making*. Chapman & Hall, London, 1993.

Bayes, R. T. Essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society of London*, 53:370–418, 1763.

Bean, C. L., C. Kambhampati and S. Rajasekharan. A rough set solution to a fuzzy set problem. In FUZZ-IEEE '02, pages 18–23.

Becker, P. W. *Recognition of Patterns: Using the Frequencies of Occurrence of Binary Words*. Springer-Verlag, 2nd edition, 1968.

Bellman, R. *Adaptive Control Processes: A Guided Tour*. Princeton University Press, New Jersey, 1961.

Bennett, N. L., L. L. Casebeer *et al.* Family physicians' information seeking behaviours: A survey comparison with other specialties. *BMC Medical Informatics and Decision Making*, 5(9), 2005.

Berner, E. S., editor. *Clinical Decision Support Systems: Theory and Practice*. Springer-Verlag, 1988. ISBN 0-387-98575-1.

Bezdek, J. C. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Advanced Applications In Pattern Recognition. Plenum Press, New York and London, 1981.

Booty, W. G., D. C. L. Lam *et al.* Great Lakes toxic chemical decision support system. In Denzer, Swayne and Schimak (1997). IFIP TC5 WG5.11 International Symposium on Environmental Software Systems (ISESS '97).

Boyen, X. and L. Wehenkel. Automatic induction of fuzzy decision trees and its application to power system security assessment. *Fuzzy Sets and Systems*, 102(1):3–19, 1999. ISSN 0165-0114. doi:http://dx.doi.org/10.1016/S0165-0114(98)00198-5.

Brath, R. 3D interactive information visualization: Guidelines from experience and analysis of applications. In *4th International Conference on Human–Computer Interaction*. June 1997a.

——. Metrics for effective information visualization. In *Information Visualization*. IEEE, Phoenix, Oct 1997b. doi:10.1109/INFVIS.1997.636794.

——. Paper landscapes: A visualization design methodology. In R. F. Erbacher, P. C. Chen, J. C. Roberts, M. T. Groehn and K. Boerner, editors, *Visualization and Data Analysis*, volume 5009. International Society for Optical Engineering (SPIE), Jun 2003. ISBN 0-8194-4809-5.

Brath, R. and M. Peters. Dashboard design: Why design is important. *Data Mining Review/Data Mining Direct*, October 2004.

Brillman, J. C., T. Burr *et al.* Modeling emergency department visit patterns for infectious disease complaints: Results and application to disease surveillance. *BMC Medical Informatics and Decision Making*, 5(4), 2005.

Buchanan, B. G. and E. H. Shortliffe, editors. *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*. Addison-Wesley, Reading, Massachusetts, 1984.

Camps-Valls, G., M. Martínez-Ramón *et al.* Robust γ-filter using support vector machines. *Neurocomputing*, 62:493–499, 2004. doi:10.1016/j.neucom.2004.07.003.

Carpenter, G. A. and S. Grossberg. ART 2: Self-organization of stable category recognition codes for analog input patterns. *Applied Optics*, 26(23):4919–4930, 1987a.

——. A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics, and Image Processing*, 37(1):54–115, 1987b.

——. ART 3: Hierarchical search using chemical transmitters in self-organizing pattern recognition architectures. *Neural Networks*, 3(2):129–152, 1990.

Carpenter, G. A., S. Grossberg and J. H. Reynolds. ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network. *Neural Networks*, 4(5):565–588, 1991a.

Carpenter, G. A., S. Grossberg and D. B. Rosen. Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system. *Neural Networks*, 4(6):759–771, 1991b.

Catlett, J. *Megainduction: Machine Learning on Very Large Databases*. Ph.D. thesis, Basser Department of Computer Science, University of Sydney, Sydney, Australia, 1991.

Chan, K. C. C. and D. K. Y. Wong, Andrew K. C. Chiu. Learning sequential patterns from probabilistic inductive prediction. *IEEE Transactions Systems, Man, Cybernetics*, 24(10):1532–1547, October 1994.

Chau, T. Marginal maximum entropy partitioning yields asymptotically consistent probability density functions. *IEEE Transactions on Pattern Analalysis & Machine Intelligent*, 23(4):414–417, April 2001.

Chau, T. and A. K. C. Wong. Pattern discovery by residual analysis and recursive partitioning. *IEEE Transactions on Knowledge & Data Engineering*, 11(6):833–852, Nov-Dec 1999.

Chen, L., N. Tokuda *et al.* A new scheme for an automatic generation of multi-variable fuzzy systems. *Fuzzy Sets and Systems*, 120:323–329, 2001.

Chen, M.-Y. Establishing interpretable fuzzy models from numeric data. In *Proceedings of the 4th World Congress on Intelligent Control and Automation*, volume 3, pages 1857–1861. IEEE, Jun 2002.

Chen, M.-Y. and D. A. Linkens. Rule-base self generation and simplification for data-driven fuzzy models. In FUZZ-IEEE '01, pages 424–427.

Chiang, I.-J. and J. Y.-j. Hsu. Fuzzy classification on trees for data analysis. *Fuzzy Sets and Systems*, 130(1):87–99, 2002. ISSN 0165-0114. doi:http://dx.doi.org/10.1016/S0165-0114(01)00212-3.

Ching, J. Y., A. K. C. Wong and K. C. C. Chan. Class-dependent discretization for inductive learning from continuous and mixed-mode data. *IEEE Transactions on Pattern Analalysis & Machine Intelligent*, 17(7):641–651, 1995.

Chong, A., T. D. Gedon *et al.* A histogram-based rule extraction technique for fuzzy systems. In FUZZ-IEEE '01, pages 638–641.

Coiera, E. *Guide to Health Informatics*. Arnold/Hodder & Stoughton, UK, 2nd edition, 2003. ISBN 0-340-76425-2.

Colombet, I., T. Dart *et al.* A computer decision aid for medical prevention: A pilot qualitative study of the personalized estimate of risks (EsPeR) system. *BMC Medical Informatics and Decision Making*, 3(13), 2003.

Cordella, L. P., P. Foggia *et al.* Reliability parameters to improve combination strategies in multi-expert systems. *Pattern Analasis & Applications*, 2:205–214, 1999.

Cordón, O., F. Herrera *et al. Genetic Fuzzy Systems : Evolutionary Tuning and Learning of Fuzzy Knowledge Bases*, chapter 11, pages 375–382. In Cordón, Herrera *et al.* (2001b), 2001a.

——. *Genetic Fuzzy Systems : Evolutionary Tuning and Learning of Fuzzy Knowledge Bases*. World Scientific, Singapore, 2001b. ISBN 981-02-4017-1.

Cortés, U. and M. Sànchez-Marrè, editors. *Environmental Decision Support Systems and Artificial Intelligence: Papers from the AAAI Workshop*, Report WS-99-07. AAAI Press, 1999. ISBN 1-57735-091-x.

Cortés, U., M. Sànchez-Marrè and F. Wotawa, editors. *Workshop on Environmental Decision Support Systems (EDSS'2003)*. IJCAI, Acapulco, Mexico, Aug 2003.

Costa, P. J., J. P. Dunyak and M. Mohtashemi. Models, prediction, and estimation of outbreaks of infectious disease. In *Proceedings of SoutheastCon*, pages 174–178. IEEE : Institute of Electrical and Electronics Engineers, Inc., April 2005.

Costa Branco, P. and J. A. Dente. Fuzzy systems modeling in practice. *Fuzzy Sets and Systems*, 121:73–93, Jul 2001.

Cowan, D. F., editor. *Informatics for the Clinical Laboratory: A Practical Guide*. Health Informatics Series. Springer-Verlag, 2003.

Cox, E. *The Fuzzy Systems Handbook*. Academic Press Professional, Cambridge, MA, 1994.

Cox, R. T. Probability frequency and reasonable expectation. *American Journal of Physics*, 14:1–13, 1946.

Cristianini, N. and J. Shawe-Taylor. *An Introduction to Support Vector Machines*. Cambridge University Press, 2000. ISBN 0-521-78019-5.

de Graaf, P. M. A., G. C. van den Eijkel *et al.* A decision-driven design of a decision supoprt system in anesthesia. *Artificial Intelligence In Medicine*, 11:141–153, 1997.

de Tré, G. and R. de Caluwe. Level-2 fuzzy sets and their usefulness in object-oriented database modelling. *Fuzzy Sets and Systems*, 140(1):29–49, Nov 2003.

Dempster, A. P. A generalization of Bayesian inference. *Journal of the Royal Statistical Society, Series B*, 30:205–247, 1968.

Denzer, R., D. A. Swayne and G. Schimak, editors. *Environmental Software Systems*, volume 2. Chapman & Hall, London, New York, 1997. ISBN 0-412-81740-3. IFIP TC5 WG5.11 International Symposium on Environmental Software Systems (ISESS '97).

Devadoss, P. R., S. L. Pan and S. Singh. Managing knowledge integration in a national health-care crisis: Lessons learned from combating SARS in Singapore. *IEEE Transactions on Information Technology in Biomedicine*, 9(2):266–275, June 2005.

Dick, S., A. Schencker *et al.* Re-granulating a fuzzy rulebase. In FUZZ-IEEE '01, pages 372–375.

Duda, R. O., P. E. Hart and D. G. Stork. *Pattern Classification*. John Wiley & Sons, 2nd edition, 2001. ISBN 0-471-05669-3.

Friedland, D. J., editor. *Evidence-Based Medicine: A Framework for Clinical Practice*. Appleton & Lange, 1998.

Fukumi, M. and N. Akamatsu. Evolutionary approach to rule generation from neural networks. In *Proceedings of the 1999 IEEE International Fuzzy Systems Conference*, volume III, pages 1388–1393. FUZZ-IEEE, Aug 1999.

FUZZ-IEEE '00. *Proceedings of the 9th IEEE International Conference on Fuzzy Systems*. IEEE, San Antonio, USA, May 2000. ISBN 0-7803-5877-5. ISSN 1098-7584.

FUZZ-IEEE '01. *Proceedings of the 10th IEEE International Conference on Fuzzy Systems, FUZZ-IEEE'01*. IEEE, Melbourne, Australia, 2001. ISBN 0-7803-7293-X. ISSN 1098-7584.

FUZZ-IEEE '02. *Proceedings of the 11th IEEE International Conference on Fuzzy Systems, FUZZ-IEEE'02*. IEEE, Hololulu, Hawaii, 2002. ISBN 0-7803-7279-4. ISSN 1098-7584.

FUZZ-IEEE '99. *Proceedings of the 8th IEEE International Conference on Fuzzy Systems*. IEEE, Seoul, Korea, Aug 1999. ISBN 0-7803-5406-0. ISSN 1098-7584.

Gabrys, B. Learning hybrid neuro-fuzzy classifier models from data: To combine or not to combine? *Fuzzy Sets and Systems*, 147(1):39–56, 2004.

Galassi, M., J. Davies *et al. GNU Scientific Library Reference Manual*. Network Theory, 2nd edition, March 2005. ISBN 0-954161-73-4. Software Library Available Online.
URL `http://www.gnu.org/software/gsl`

Gay, A. K. Simulating biological variation in MFAPs: Resultant effects on MUAPs and EMG signals. Graduate research term report, Department of Systems Design Engineering, University of Waterloo, Waterloo, Ontario, Canada, Dec 1999. SYDE 642: Simulation.

Gelernter, D. *Machine Beauty: Elegance and the Heart of Technology*. Basic Books (Perseus), New York, Jan 1998. ISBN 0-465-04516-2.

Gibb, W. J., D. M. Auslander and J. C. Griffin. Adaptive classification of myocardial electrogram waveforms. *IEEE Transactions on Biomedical Engineering*, 4(8):804–808, Aug 1994.

Ginsberg, M. L. *Essentials of Artificial Intelligence*. Morgan Kaufman, San Francisco, 1993. ISBN 1-55860-221-6.

Gokhale, D. V. On joint and conditional entropies. *Entropy*, 1(2):21–24, 1999.

Goldberg, D. E. *Genetic Algorithms in Search, Optimization & Learning*. Addison-Wesley, 1989.

Goldberg, D. E. and K. Deb. A comparative analysis of selection schemes used in genetic algorithms. In G. J. E. Rawlins, editor, *Foundataions of Genetic Algorithms*, pages 69–93. Morgan Kaufman Publishers, 1991.

Grossberg, S. Adaptive pattern classification and universal recoding: I. parallel development and coding of neural feature detectors. *Biological Cybernetics*, 23:121–134, 1976.

——. Adaptive resonance theory (ART). In M. A. Arbib, editor, *Handbook of Brain Theory and Neural Networks*, pages 79–81. MIT Press, Cambridge, MA, June 1995. ISBN 0-262-01148-4.

Grzymala-Busse, J. and W. Ziarko. Discovery through rough set theory. *Communications of the ACM*, 42:55–57, 1999.

——. Data mining and rough set theory. *Communications of the ACM*, 43:108–109, 2000.

Gupta, M. M., R. K. Ragade and R. R. Yager, editors. *Advances in Fuzzy Set Theory and Applications*. North-Holland, Oxford, 1979. ISBN 0-444-85372-3.

Gurov, S. I. Reliability estimation of classification algorithms I: Introduction to the problem - point frequency estimates. *Computation Mathematics and Modeling*, 15(4):365–376, 2004.

——. Reliability estimation of classification algorithms II: Point baysian estimates. *Computation Mathematics and Modeling*, 16(2):169–178, 2005.

Guthrie, G., D. A. Stacey and D. Calvert. Detection of disease outbreaks in pharmaceutical sales: Neural networks and threshold algorithms. In IJCNN '05, pages 3138–3143.

Haberman, S. J. The analysis of residuals in cross-classified tables. *Biometrics*, 29(1):205–220, Mar 1973.

——. *Analysis of Qualitative Data*, volume 1, pages 78–79,82–83. Academic Press, Toronto, 1979. ISBN 0-12-312502-2.

Harris, E. K. and J. C. Boyd. *Statistical Bases of Reference Values in Laboratory Medicine*. Marcel Dekker, Inc., New York – Basel – Hong Kong, 1995.

Hathaway, R. J. and J. C. Bezdek. Clustering incomplete relational data using the non-Euclidean relational fuzzy *c*-means algorithm. *Pattern Recognition Letters*, 23:151–160, 2002.

Heckerman, D. Probabilistic interpretation for MYCIN's certainty factors. In L. N. Kanal and J. F. Lemmer, editors, *Uncertainty in Artificial Intelligence*, pages 167–196. Elsevier/North-Holland, Amsterdam, London, New York, 1986.

Hertz, J., A. Krogh and R. G. Palmer. *Introduction to the Theory of Neural Computation*. Santa Fe Institute Studies in the Sciences of Complexity, 1991.

Heske, T. and J. N. Heske. *Fuzzy Logic for Real World Design*. Annabooks, San Diego, 1999. ISBN 0-929392-24-8.

Hipel, K. W. and Y. Ben-Haim. Decision making in an uncertain world: Information-gap modeling in water resources management. *IEEE Transactions Systems, Man, Cybernetics C*, 29(4):506–517, Nov 1999.

Hipel, K. W., L. Fang and D. M. Kilgour. Game theoretic models in engineering decision making. *Journal of Infrastructure Planning and Management*, 470(4):1–16, July 1993.

Hipel, K. W., D. M. Kilgour *et al.* Strategic decision support for the services industry. *IEEE Transactions on Engineering Management*, 48(3):358–369, Aug 2001.

Hipel, K. W., X. Yin and D. M. Kilgour. Can a costly reporting system make environmental enforcement more efficient? *Journal of Stochastic Hydrology and Hydraulics*, 9(2):151–170, 1995.

Hisdal, E. Possibilistically dependent variables and a general theory of fuzzy sets. In Gupta, Ragade and Yager (1979), pages 215–234.

Hoffmann, F. Combined boosting and evolutionary algorithms for learning of fuzzy classification rules. *Fuzzy Sets and Systems*, 141:47–58, 2004.

Hong, T.-P. and C.-Y. Lee. Induction of fuzzy rules and membership function from training examples. *Fuzzy Sets and Systems*, 84:33–47, Nov 1996.

Höppner, F., F. Klawonn *et al.* *Fuzzy Cluster Analysis*. Chichester, England, 1999.

House, W. C., editor. *Decision Support Systems: A Data-Based, Model-Oriented, User-Developed Discipline*. Petrocelli, New York/Princeton, 1983. ISBN 0-89433-225-2.

IJCNN '05. *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*. IEEE/INNS, Montréal, Québec, July 2005. ISBN 0-7803-9049-0.

Innocent, P. R. Fuzzy symptoms and a decision support index for the early diagnosis of confusable diseases. In *Proceedings of the RASC Conference*, pages 1–8. Dept. of Computing Science, De Montfort

University, Leicester, UK, Jul 2000a.
URL `http://www.cse.dmu.ac.uk/~pri/rasc.pdf`

——. A lightweight fuzzy process to support early diagnosis of confusable diseases. In FUZZ-IEEE '00, pages 516–521. doi:10.1109/FUZZY.2000.838713.

Ishibuchi, H. and T. Murata. Learning of fuzzy classification rules by a genetic algorithm. *Electronics and Communications in Japan, Part 3*, 80(3):37–46, 1997. Translated from Denshi Joho Tsushin Gakkai Ronbunshi, Vol 79-A, No. 7, 1996, pp.1289–1297.

Ishibuchi, H., K. Nozaki *et al.* Construction of fuzzy classification systems with rectangular fuzzy rules using genetic algorithms. *Fuzzy Sets and Systems*, 65(2/3):237–253, 1994.

Jain, A. K., M. N. Murty and P. J. Flynn. Data clusting: A review. *ACM Computing Surveys*, 31(3):264–323, September 1999. ISSN 0360-0300. doi:10.1145/331499.331504.

Jensen, R. and Q. Shen. Fuzzy-rough sets for descriptive dimensionality reduction. In FUZZ-IEEE '02, pages 29–34.

Joachims, T. Making large-scale svm learning practical. Technical Report LS8, Universität Dortmund, 1998.
URL `http://www.joachims.org/publications/joachims_98c.pdf`

——. Svm*light*. 2005.
URL `http://svmlight.joachims.org`

Kapler, T., R. Harper and W. Wright. Correlating events with tracked movements in time and space: a GeoTime case study. In *Proceedings of the 2005 Intelligence Analysis Conference*. Jan 2005.

Karnik, N. N. and J. M. Mendel. Applications of type-2 fuzzy logic systems: Handling the uncertainty associated with surveys. In FUZZ-IEEE '99, pages 1546–1551. doi:10.1109/FUZZY.1999.790134.

Karnik, N. N., J. M. Mendel and Q. Liang. Type-2 fuzzy logic systems. *IEEE Transactions on Fuzzy Systems*, 7(6):643–658, Dec 1999.

Kaynak, O., K. Jezernik and Á. Szeghegti. Complexity reduction of rule based models: A survey. In FUZZ-IEEE '02, pages 1216–1220.

Keller, H. and C. Trendelenburg, editors. *Data Presentation: Interpretation*. Clinical Biochemistry: Principles, Methods, Applications: 2. Walter de Gruyter, Berlin – New York, 1989.

Kennedy, J. and R. C. Eberhart. Particle swarm optimization. In *Proceedings of IEEE International Conference on Neural Networks*, pages 1942–1948. IEEE : Institute of Electrical and Electronics Engineers, Inc., 1995.

Kilgour, D. M., L. Fang and K. W. Hipel. GMCR in negotiations. *Negotiation Journal*, pages 151–156, April 1995.

Kim, M. W., J. G. Lee and C. Min. Efficient fuzzy rule generation based on fuzzy decision tree for data mining. In *Proceedings of the 1999 IEEE International Fuzzy Systems Conference*, volume III, pages 1223–1228. FUZZ-IEEE, Aug 1999.

Kirkpatrick, S., C. D. J. Gelatt *et al.* Optimization by simulated annealing. *Science*, 220(4298):671–680, May 1983.

Klir, G. J. Generalized information theory: Emerging crossroads of fuzziness and probability. In NAFIPS '05.

——. Measuring uncertainty associated with convex sets of probability distributions: A new approach. In NAFIPS '05.

Knauf, R., A. J. Gonzalez and T. Abel. A framework for validation of rule-based systems. *IEEE Transactions Systems, Man, Cybernetics B*, 32(3):281–295, June 2002.

Kononenko, I. Machine learning for medical diagnosis: History, state of the art and perpsective. *Artificial Intelligence In Medicine*, 23:89–109, 2001.

Kononenko, I. and I. Bratko. Information-based evaluation criterion for classifier's performance. *Machine Learning*, 6:67–80, 1991.

Koza, J. R. *Genetic Programming : On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.

Krishnapuram, R. and J. M. Keller. A possibilistic approach to clustering. *IEEE Transactions on Fuzzy Systems*, 1(2):98–110, 1993.

——. The possibilistic *c*-means algorithm: Insights and recommendations. *IEEE Transactions on Fuzzy Systems*, 4(3):385–393, 1996.

Kruse, R., J. E. Gebhardt and F. Klawonn. *Foundations of Fuzzy Systems*. John Wiley & Sons, New York, 1994.

Kukar, M. Transductive reliability estimation for medical diagnosis. *Artificial Intelligence In Medicine*, 29:81–106, 2003.

Kukar, M., I. Kononenko *et al.* Analysing and improving the diagnosis of ischaemic heart disease with machine learning. *Artificial Intelligence In Medicine*, 16:25–50, 1999.

Kukolj, D. Design of adaptive Takagi-Sugeno-Kang fuzzy models. *Applied Soft Computing*, 2:89–103, 2002.

Kuncheva, L. I. How good are fuzzy if-then classifiers. *IEEE Transactions Systems, Man, Cybernetics A*, 30(4):501–509, Aug 2000.

Labbi, A. and E. Gauthier. Combining fuzzy knowledge and data for neuro-fuzzy modeling. *Journal of Intelligent Systems*, 6(4), 1997.

Larsson, J. E., B. Hayes-Roth *et al.* Evaluation of a medical diagnosis system using simulator test scenarios. *Artificial Intelligence In Medicine*, 11:119–140, 1997.

Lehmann, E. L. and H. J. M. D'Abrera. *Nonparametrics: Statistical Methods Based on Ranks*. Pearson Education/Prentice-Hall, revised edition, 1998. ISBN 0-13997-735-X.

León, L. F., D. C. Lam *et al.* An environmental impact assessment model for water resources screening. In Denzer *et al.* (1997). IFIP TC5 WG5.11 International Symposium on Environmental Software Systems (ISESS '97).

Levitin, G. Evaluating correct classification probability for weighted voting classifiers with plurality voting. *European Journal of Operations Research*, 141:596–607, 2002.

——. Threshold optimization for weighted voting classifiers. *Naval Research Logistics*, 50(4):322–344, 2003.

Liang, Q. and J. M. Mendel. An introduction to type-2 TSK fuzzy logic systems. In FUZZ-IEEE '99, pages 1534–1539. doi:10.1109/FUZZY.1999.790132.

——. Interval type-2 fuzzy logic systems. In FUZZ-IEEE '00, pages 328–333. doi:10.1109/FUZZY.2000.838681.

Lin, T. Y. and N. Cercone, editors. *Rough Sets and Data Mining*. Kluwer Academic Publishers, 1997.

López de Mántaras, R. A distance-based attribute selection measure for decision tree induction. *Machine Learning*, 6:81–92, 1991.

Mahfouf, M. and D. A. Linkens. Rule-base generation via symbiotic evolution for a mamdani-type fuzzy control system. In FUZZ-IEEE '01, pages 396–399.

Majowicz, S. and D. A. Stacey. The use of clustering to analyze symptom-based case definitions for acute gastrointestinal illness. In IJCNN '05, pages 2429–2434.

Mamdani, E. H. Applications of fuzzy algorithms for simple dynamic plant. In *Procedings of the IEEE*, volume 121, pages 1585–1588. IEEE, 1974.

MathWorld. *MathWorld*, August 2005.
URL `http://mathworld.wolfram.com/`

Mendel, J. M. Fuzzy logic sytems for engineering: a tutorial. *Procedings of the IEEE*, 83(2):345–377, Mar 1995a.

——. Fuzzy logic sytems for engineering: a tutorial – errata. *Procedings of the IEEE*, 83, Sep 1995b.

——. *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*. Prentice-Hall, 2001. ISBN 0-13-040969-3.

Mendel, J. M. and R. I. B. John. Type-2 fuzzy sets made simple. *IEEE Transactions on Fuzzy Systems*, 10(2):117–127, April 2002.

Metz, C. E. Basic principles of ROC analysis. *Seminars in Nuclear Medicine*, 8:283–298, 1978.

Minsky, M. L. and S. A. Papert. *Perceptrons : An Introduction to Computational Geometry*. MIT Press, 2nd edition, 1988. ISBN 0-262-63111-3.

Mitchell, M. *An Introduction to Genetic Algorithms*. MIT Press, Cambridge, Massachusetts, 1996.

Mitra, S. and Y. Hayashi. Neuro-fuzzy rule generation: Survey in soft computing framework. *IEEE Transactions on Neural Networks*, 11(3):748–768, May 2000.

Montani, S., P. Magni *et al.* Integrating model-based decision support in a multi-modal reasoning system for managing type 1 diabetic patients. *Artificial Intelligence In Medicine*, 29:131–151, 2003.

Moon, T. K. *Electrical and Computer Engineering 7680: Information Theory Course Notes*. Utah State University, Sept 2000.
URL `http://www.engineering.usu.edu/classes/ece/7680/lecture2.pdf`

Morton, S. *Management Decision Systems: Computer-Based Support for Decision Making*. Harvard University Press, 1971. ISBN 0-87584-090-6.

Muresan, L., K. T. László and K. Hirota. Similarity in hierarchical fuzzy rule – base systems. In FUZZ-IEEE '02, pages 746–750.

Murphy, A. K. G. *Effective Information Display and Interface Design for Decomposition-Based Quantitative Electromyography*. Master's thesis, University of Waterloo, Systems Design Engineering, 2002.

NAFIPS '05. *Proceedings of the 2005 International Joint Conference of the North American Fuzzy Information Processing Society Biannual Conference*. NAFIPS, Jun 2005. ISBN 0-7803-9188-8.

Nauck, D., F. Klawonn and R. Kruse. *Foundations of Neuro-Fuzzy Systems*. John Wiley & Sons, New York, 1997.

Newman, D. J., S. Hettich *et al.* UCI repository of machine learning databases. 1998.
URL `http://www.ics.uci.edu/~mlearn/MLRepository.html`

Norman, D. A. *The Design of Everyday Things*. Basic Books (Perseus) or MIT press (UK), New York, 1998/2002. ISBN 0-262-64037-6. Formerly *The Psychology of Everyday Things*.

O' Carroll, P. W., W. A. Yasnoff *et al.*, editors. *Public Health Informatics and Information Systems*. Health Informatics. Springer-Verlag, 2002. ISBN 0387954740.

Øhrn, A. *Discernability and Rough Sets in Medicine: Tools and Applications*. Ph.D. thesis, Department of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway, 1999. Institutt for datateknikk og informasjonvitenskap IDI-rapport 1999:14.

Pal, N. R., J. C. Bezdek and R. J. Hathaway. Sequential competitive learning and the fuzzy c-means clustering algorithms. *Neural Networks*, 9(5):787–796, 1996.

Pal, N. R., K. Pal and J. C. Bezdek. A mixed c-means clustering model. In *Proceedings of the Sixth IEEE International Conference on Fuzzy Systems*, volume 1, pages 11–21. FUZZ-IEEE, Jul 1997.

Pal, S. K. and S. Mitra. *Neuro-Fuzzy Pattern Recognition : Methods in Soft Computing*. Wiley Series on Intelligent Systems. Wiley-Interscience, 1999. ISBN 0-471-34844-9.

Pawlak, Z. Rough sets. *International Journal of Computing and Information Sciences*, 11(5):341–356, 1982.

——. *Rough Sets: Theoretical Aspects of Reasoning about Data*. Studies in Fuzziness and Soft Computing. Kluwer Academic, Norwell, MA, 1992. ISBN 0-7923-1472-7.

Pedrycz, W. *Fuzzy Sets Engineering*. CRC Press, 1995. ISBN 0-8493-9402-3.

Penaloza, M. A. and R. M. Welch. Infectious disease and climate change: A fuzzy database management system approach. In *Remote Sensing - A Scientific Vision for Sustainable Development*, volume 4, pages 1950–1952. IGARSS: Geoscience and Remote Sensing, Aug 1997.

Pino, L. Discussion on PD conditional probability derivation from WOE measure. Personal Communication, July 2005.

Polkowski, L. and A. Skowron, editors. *Rough Sets in Knowledge Discovery 1: Methodology and Applications*, volume 18 of *Studies in Fuzziness and Soft Computing*. Physica-Verlag, 1998. ISBN 3-7908-1119-X.

Press, W. H., S. A. Teukolsky *et al.* *Numerical Recipes in C : The Art of Scientific Computing*. Cambridge University Press, 2nd edition, 1992. ISBN 0-521-43108-5.

Quinlan, J. R. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986.

——. *C4.5 : Programs for Machine Learning*. Morgan Kaufman, 1993.

——. Learning first-order definitions of functions. *Journal of Artificial Intelligence Research*, 5:139–161, 1996.

Rajabi, S., D. M. Kilgour and K. W. Hipel. Modeling action-interdependence in multiple critera decision making. *European Journal of Operations Research*, 110(3):490–508, Nov 1998.

Rumelhart, D. E., G. E. Hinton and R. J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

Russell, S. and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2003. ISBN 0-13-790395-2.

Sage, A. P. *Decision Support Systems Engineering*. Wiley Series in Systems Engineering. John Wiley & Sons, 1991.

Sage, A. P. and W. B. Rouse, editors. *Handbook of Systems Engineering and Management*, chapter Operations Research and Refinement of Courses of Action. John Wiley & Sons, Apr 1999. ISBN 0-471-15405-9.

Sayood, K. *Introduction to Data Compression*. Morgan Kaufmann, 2nd edition, 2000.

Schniederjans, M. *Case Studies in Decision Support*. Petrocelli, Princeton, 1987.

Scott, A. C., J. E. Claton and E. L. Gibson. *A Practical Guide to Knowledge Acquisition*. Addison-Wesley, 1991.

Shafer, G. Perspectives on the theory and practice of belief functions. *International Journal of Approximate Reasonong*, 3:1–40, 1990.

Shafer, G. and J. Pearl. *A Mathematical Theory of Evidence*. Princeton University, Princeton, New Jersey, 1976.

Shafer, G. and J. Pearl, editors. *Readings in Uncertain Reasoning*. Morgan Kaufmann, 1990. ISBN 1-55860-125-2.

Shannon, C. E. A mathematical theory of communication: Part 1. *Bell Systems Technical Journal*, 27:379–423, July 1948a. Reprinted in Slepian (1974).
URL `http://cm.bell-labs.com/cm/ms/what/shannonday/paper.html`

——. A mathematical theory of communication: Part 2. *Bell Systems Technical Journal*, 27:623–656, October 1948b. Reprinted in Slepian (1974).
URL `http://cm.bell-labs.com/cm/ms/what/shannonday/paper.html`

Shen, Q. and A. Chouchoulas. A rough-fuzzy approach for generating classification rules. *Pattern Recognition*, 35:2425–2438, 2002.

Shortliffe, E. H. *Computer-Based Medical Consultations: MYCIN*. Elsevier/North-Holland, Amsterdam, London, New York, 1976.

Shortliffe, E. H. and L. E. Perreault, editors. *Medical Informatics : Computer Applications in Health Care and Biomedicine*. Springer-Verlag, 2nd edition, 1990.

Silver, M. S. *Systems That Support Decision Makers: Description and Analysis*. John Wiley Information Systems Series. John Wiley & Sons, 1991. ISBN 0-471-91968-3.

Simpson, P. K. *Artificial Neural Systems*. Windcrest/McGraw-Hill, 1991. ISBN 0-07-105355-7.

Slepian, D., editor. *Key Papers in the Development of Information Theory*. IEEE Press, New York, 1974.

Spiegel, D. and T. Sudkamp. Sparse data in the evolutionary generation of fuzzy models. *Fuzzy Sets and Systems*, 138:363–379, 2003.

Sprague, R. and E. Carlsons. *Building Effective Decision Support Systems*. Prentice-Hall, 1982. ISBN 0-13-086215-0.

Sugeno, M., M. F. Griffin and A. Bastian. Fuzzy hierarchical control of an unmanned helicopter. In *Proceedings of the 5th IFSA World Congress (IFSA '93)*. Seoul, 1993.

Sugeno, M. and G. T. Kang. Structure identification of fuzzy model. *Fuzzy Sets and Systems*, 28:15–33, 1988.

Sundararajan, C. R. *Guide to Reliability Engineering: Data Analysis, Applications, Implementation, and Management*. Van Nostrand Reinhold, New York, 1991.

Syswerda, G. A study of reproduction in generational and steady-state genetic algorithms. In G. J. E. Rawlins, editor, *Foundataions of Genetic Algorithms*, pages 94–112. Morgan Kaufman Publishers, 1991.

Takagi, T. and M. Sugeno. Fuzzy identification of systems and its application to modeling. *IEEE Transactions Systems, Man, Cybernetics*, 15:116–132, 1985.

Tape, T. G. Interpreting diagnostic tests. Technical report, University of Nebraska Medical Center, 2005. URL `http://gim.unmc.edu/dxtests`

Toscano, R. and P. Lyonnet. Parameterization of a fuzzy classifier for the diagnosis of an industrial process. *Reliability Engineering & System Safety*, 77:269–279, 2002.

Tsumoto, S. Statistical evidence for rough set analysis. In FUZZ-IEEE '02, pages 757–762.

Tufte, E. R. *The Visual Display of Quantitative Information*. Graphics Press, Cheshire, Conneticut, 2nd edition, 1983. ISBN 0-961-39214-2.

——. *Envisioning Information*. Graphics Press, Cheshire, Conneticut, 1991. ISBN 0-961-39211-8.

——. *Visual Explanations: Images and Quantities, Evidence and Narritive*. Graphics Press, Cheshire, Conneticut, 1997. ISBN 0-961-39212-6.

Türkşen, I. B. Interval valued fuzzy sets and fuzzy measures. In *Proceedings of the First International Joint Conference of the North American Fuzzy Information Processing Society Biannual Conference*, volume 3, pages 317–321. NAFIPS, Dec 1994.

——. Knowledge representation and approximate reasoning with type II fuzzy sets. In *International Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and The Second International Fuzzy Engineering Symposium*, volume 4, pages 1911–1917. IEEE : Institute of Electrical and Electronics Engineers, Inc., March 1995a.

——. Linguistics and uncertainty in intelligent systems. In *Proceedings of the International Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and The Second International Fuzzy Engineering Symposium*, volume 3, pages 2297–2304. FUZZ-IEEE, 1995b.

——. Type I and type II fuzzy system modeling. *Fuzzy Sets and Systems*, 106(1):11–34, 1999. ISSN 0165–0114.

Vapnik, V. N. *The Nature of Statistical Learning Theory*. Springer-Verlag, 1995. ISBN 0-262-01148-4.

Wanas, N. M. and M. S. Kamel. Weighted combination of neural network ensembles. In D. B. Fogel, M. A. El-Sharkawi, X. Yao, G. Greenwood, H. Iba, P. Marrow and M. Shackleton, editors, *Proceedings of the 2002 World Congress on Computational Intelligence WCCI 2002*, pages 1748–1752. IEEE Press, Honolulu, 2002. ISBN 0-7803-7278-6.

Wang, L. X. Fuzzy systems are universal approximators. In *Proc. IEEE Int. Conf. on Fuzzy Systems*, pages 1163–1169. San Diego, 1992.

Wang, X. Z., Y. D. Wang *et al.* A new approach to fuzzy rule generation: Fuzzy extension matrix. *Fuzzy Sets and Systems*, 123:291–306, 2001.

Wang, Y. *High Order Pattern Discovery and Analysis of Discrete-Valued Data Sets*. Ph.D. thesis, University of Waterloo, Systems Design Engineering, 1997.

Wang, Y. and A. K. C. Wong. From association to classification: Inference using weight of evidence. *IEEE Transactions on Knowledge & Data Engineering*, 15(3):764–767, May-June 2003.

Waterman, D. A. and F. Hayes-Roth, editors. *Pattern-Directed Inference Systems*. Academic Press, New York, 1978.

Wiener, N. *Cybernetics: or Control and Communication in the Animal and the Machine*. MIT Press, 1948, 1961. ISBN 0-262-73009-X.

——. *The Human Use of Human Beings: Cybernetics and Society*. Da Capo Series in Science. Da Capo Press, 1950. ISBN 0-306-80320-8.

Witten, I. H. and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, chapter WEKA : Machine Learning Algorithms in Java. Morgan Kaufman, 2000.

Wong, A. K. C. and D. Ghahrarnan. A statistical analysis of interdependence in character sequences. *Information Sciences*, 8(2):173–188, April 1975.

Wong, A. K. C., T. S. Liu and C. C. Wang. Statistical analysis of residue variability in cytochrome C. *Journal of Molecular Biology*, 102(2):287–295, April 1976.

Wong, A. K. C. and Y. Wang. High-order pattern discovery from discrete-valued data. *IEEE Transactions on Knowledge & Data Engineering*, 9(6):877–893, Nov-Dec 1997.

——. Pattern discovery: A data driven approach to decision support. *IEEE Transactions Systems, Man, Cybernetics C*, 33(1):114–124, Feb 2003.

Wright, W. Multi-dimensional representations — how many dimensions? In *New Paradigms in Information Visualization and Manipulation*. ACM, Nov 1997.

Wrigley, N. *Categorical Data Analysis for Geographers and Environmental Scientists*. Longman, 1985.

Xing, H., S. H. Huang and J. Shi. Rapid development of knowledge-based systems via integrated knowledge acquisition. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 17:221–234, 2003.

Xu, L., J. Neufeld *et al.* Maximum margin clustering. In L. K. Saul, Y. Weiss and L. Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 1537–1544. Neural Information Processing Systems Foundation NIPS, MIT Press, Cambridge, MA, 2005.

Yager, R. R., M. Fedrizzi and J. Kacprzyk, editors. *Advances in the Dempster-Shafer Theory of Evidence*. Wiley, 1994.

Yager, R. R. and D. P. Filev. Relational partitioning of fuzzy rules. *Fuzzy Sets and Systems*, 80:57–69, 1996.

Yager, R. R., S. Ovchinnikov *et al.*, editors. *Fuzzy Sets and Applications: Selected Papers by L.A. Zadeh*. John Wiley & Sons, 1987.

Young, W. J., D. C. L. Lam *et al.* Development of an environmental flows decision support system. In Denzer *et al.* (1997). IFIP TC5 WG5.11 International Symposium on Environmental Software Systems (ISESS ’97).

Yyrdusev, M. An environmental impact assessment model for water resources screening. In Denzer *et al.* (1997). IFIP TC5 WG5.11 International Symposium on Environmental Software Systems (ISESS ’97).

Zadeh, L. Reviews of books: A mathematical theory of evidence. *AI Magazine*, 5(3), Fall 1984.

Zadeh, L. A. Fuzzy sets. *Information and Control*, 8:338–353, 1965. Reprinted in Yager *et al.* (1987).

——. Probability measures of fuzzy events. *Journal of Mathimatical Analysis and Application*, 23:421–427, 1968. Reprinted in Yager *et al.* (1987).

——. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions Systems, Man, Cybernetics*, 3:28–44, 1973. Reprinted in Yager *et al.* (1987).

——. The concept of a linguistic variable and its application to approximate reasoning – part 1. *Information Sciences*, 8:199–249, 1975a. Reprinted in Yager *et al.* (1987).

——. The concept of a linguistic variable and its application to approximate reasoning – part 2. *Information Sciences*, 8:301–357, 1975b. Reprinted in Yager *et al.* (1987).

——. The concept of a linguistic variable and its application to approximate reasoning – part 3. *Information Sciences*, 9:43–80, 1975c. Reprinted in Yager *et al.* (1987).

——. A fuzzy-algorithmic approach to the definition of complex or imprecise concepts. *International Journal of Man-Machine Studies*, 8:249–291, 1976. Reprinted in Yager *et al.* (1987).

——. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 1:3–28, 1978. Reprinted in Yager *et al.* (1987).

——. The role of fuzzy logic in the management of uncertainty in expert systems. In J. Hayes, D. Michie and L. I. Mikulich, editors, *Machine Intelligence*, volume 9, pages 149–194. Halstead Press, New York, 1979. Reprinted in Yager *et al.* (1987).

——. The role of fuzzy logic in the management of uncertainty in expert systems. *Fuzzy Sets and Systems*, 11:199–227, 1983. Reprinted in Yager *et al.* (1987).

Zhang, X., R. Fiedler and M. Popovich. A biointelligence system for identifying potential disease outbreaks. *IEEE Engineering in Medicine & Biology Magazine*, 23(1):58–64, Jan–Feb 2004.

Ziarko, W. Decision making with probabalistic decision tables. In *Proceedings of the 7th International Workshop on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing*, pages 463–471. RSFD-CrC'99, Yamaguchi, Japan, 1999.

——. Optimal decision making with data-acquired decision tables. In *Proceedings of the International Symposium on Intelligent Information Systems*, pages 75–85. IIS, Bystra, Poland, 2000.

——. Probabilistic decision tables in the variable precision rough set model. *Computational Intelligence*, 17(3):593–603, 2001.

——. Acquisition of hierarchy-structured probabalistic decision tables and rules from data. In FUZZ-IEEE '02, pages 779–784.

Index