Event Level Pattern Discovery in Multivariate Continuous Data

by

Tom Chau

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

Waterloo, Ontario, Canada, 1997

0-612-30594-5

The University of Waterloo requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

Abstract

Patterns naturally arise in all types of data. The universal drive to uncover and understand these patterns has generated a wide range of pattern discovery tools and algorithms. The result is that patterns are sought after in many different forms, including rules, network weights, topologies, hierarchical trees, hypergraphs, membership functions, probability density functions and functional relationships. This diversity of pattern instantiations raises the question as to what is the fundamental information in the data.

In this thesis, the event is promoted as the fundamental information bearing entity in continuous data. Events, event associations and patterns are defined for the continuous sample space. From the event perspective, pattern discovery is viewed as the search for statistically significant events, where significance is judged according to the objective of the discovery. Hence, event-based pattern discovery is formulated as a mathematical optimization problem with statistical objective functions. A novel sequential and recursive methodology is proposed as the solution technique to the optimization task.

For two or three dimensional data, an approximation based on selective recursive partitioning is developed. The application of the discovered events to multivariate density estimation, smoothing and classification demonstrate the versatility of the event framework. An event-based measure of significant temporal change forms the basis for a time-dependent discovery algorithm. A new event synthesis procedure facilitates the analysis of high-dimensional data by selectively constructing high dimensional events. Parallel event plots serve as an interpretative visualization tool.

Experiments illustrate that on a classical front. event-based classification can be comparable to existing methodologies. From a discovery standpoint, the event approach offers unprecedented interpretability of complicated multivariate continuous data. Local dependencies are easily revealed and locally significant features are immediately identified. Temporal changes can be objectively assessed and elusive high-dimensional outliers can be detected. Subdimensional clusters. while traditionally challenging to unravel, are handled confidently with event-based pattern discovery.

Acknowledgements

To you O God, I sing praise and thanksgiving.

I would like to thank my parents Frank and Irene, who have supported me financially, emotionally and spiritually throughout my studies. Their constant love has pulled me through every hardship along the way.

I am also deeply grateful for the love and care that my wife Grace has incessantly poured into my life. I thank her for tolerating the intolerable.

Thanks is also due for my brothers and sisters who have provided encouragement and have instilled within me an ever-growing sense of service.

To my friends, your benevolence has been a beacon of hope. I pray that I'll be able to reciprocate your kindness in the years to come. My gratitude extends especially to Hassan Refae, Murray Rice, Brian Lim, Steve Wilkinson, Clem Yip, Carolyn Kwan, Cassandra Luhur, Dr. Jeff Pang, Dan Davison, Murugavel Mahalingam and John Peng.

A gesture of gratitude goes out also to my supervisor, Prof. Wong and to my committee members, especially Prof. Paul Calamai.

Contents

List of Figures

List of Tables

Symbol	Meaning
A	projection matrix
$B(\Re^d)$	Borel σ-field
c_j	asymptotic variance of residual r_j
d	dimensionality
D	diagonal matrix (Chapter 4) or dimension set (Chapter 5)
E	event
E_j	j^{th} event in a partition
E_{J+1}	background event
$\hat{f}(\mathbf{x})$	continuous probability density function
$\hat{p}(\mathbf{x})$	discrete probability density function
$g(\cdot)$	test statistic corresponding to a discovery objective
H	entropy
I	set of integers
I_x, I_y, I_z	one-dimensional intervals in $\Re$
J	discovery objective
K	number of classes
l_i	length of interval in dimension i
M	number of significant events in a partition
n_j	observed frequency of an event E_j
N	sample size
pdf	probability density function
$\hat{P}_j$	estimated probability of an event E_j
$\hat{p}_J$	estimated probability density of an event E_j
q	number of bits in genetic algorithm coding
Q	partition size
r_j	residual value for event E_j
R_i	relaxed event
$\Re^d$	d-dimensional Euclidean space
S	subspace of the sample space Ω
$S(t)$	subspace at time t
SS_{reg}	regression sum-of-squares
SS_{err}	error sum-of-squares
T	time interval between observations
$X(\cdot)$	random variable mapping
z	critical value for standard normal random variable
$\lvert \cdot \rvert$	cardinality

Greek Symbol	Meaning
α	significance level
α_0	significance level at $d = 2$
η^2	statistical strength of dependency
γ	positive scaling constant
Γ	image space
κ	index set for insignificant events
Λ	proportion of events with expected frequency < 5
ω	observation or outcome in the sample space
Ω	sample space
ρ_{obs}	observed density
ρ_{unif}	uniform density
θ_c^α	critical value of test statistic
$\tilde{\theta}$	optimization parameters
ξ	Minimum number of points per cell
ζ	Mean value point

Chapter 1

Introduction

In this introductory chapter, the emphasis is on an intuitive overview of the thesis. Terminology that is introduced here will be followed by more precise definitions in later chapters.

1.1 What is pattern discovery?

A pattern is a collection of observations or measurements whose occurrence in an ensemble is statistically significant. The stipulation of statistical significance implies that the observations are not just a chance occurrence, but rather indicative of some underlying process or law. Each observed or measured entity constitutes a variable in the pattern description.

Patterns can be described on 2 different levels. At the variable level, a pattern is a mathematical relation among variables while at the event level, a pattern is a subset of variable values. For now, we can think of an event as a subset of

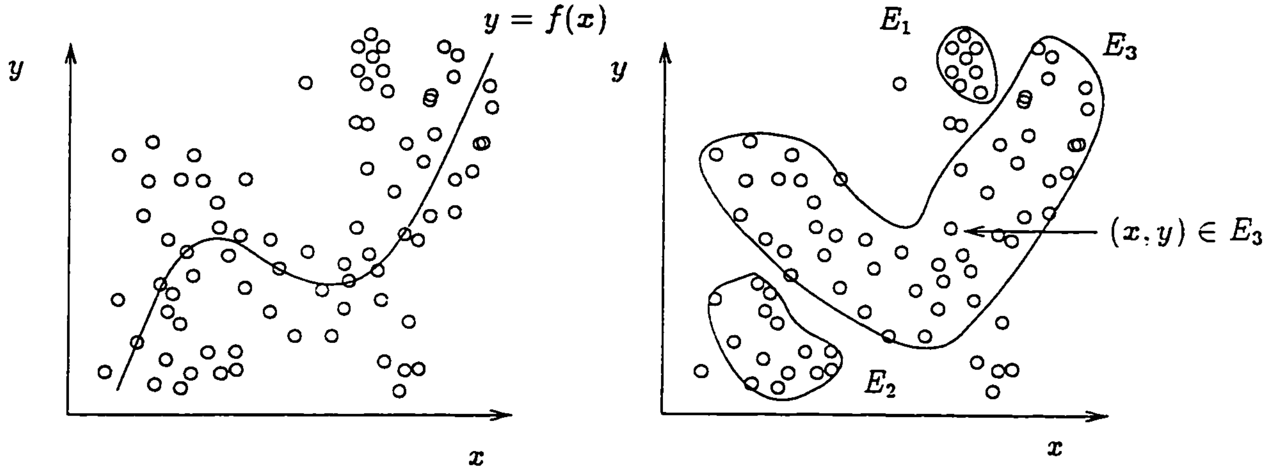


Figure 1.1: Variable and event level descriptions

observations or measurements. On the left-hand side of Figure 1.1, a data set is described at the variable level by a functional relationship, $y = f(x)$. On the opposite side, the same data set is described by 3 subsets or events, E_1, E_2 and E_3.

Broadly speaking, pattern discovery is the search for significant patterns in data and in particular, event-based pattern discovery is the search for significant subsets of variable values. To appreciate the central role that pattern discovery plays in the problem of intelligent data analysis, I will briefly outline each component in Figure 1.2.

The collection stage involves the gathering of data and one typically encounters the difficulties of excessive data, insufficient data or fractured data [44]. In light of these problems with experimental measurements and observations, the second stage of intelligent data analysis usually involves some preprocessing. A broad range of techniques for preprocessing are available, including filtering, editing, dimensionality reduction and sampling. The common goal of these techniques is to

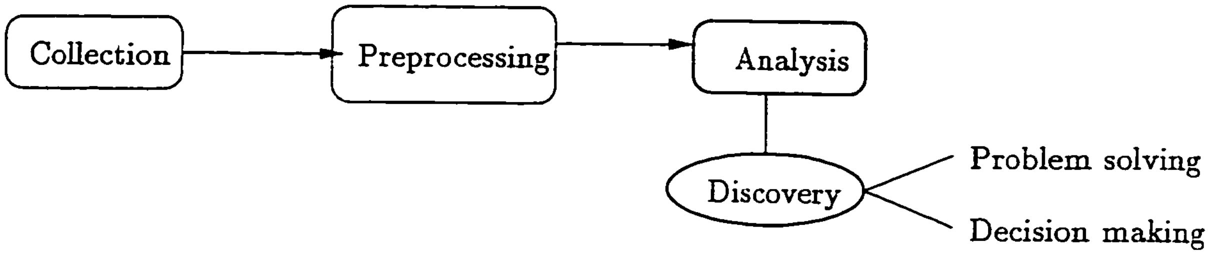


Figure 1.2: The general data analysis problem adapted from Famili [44]

better prepare the data for analysis and interpretation. In the analysis stage, the objective is "*to discover patterns that will be used to solve problems or make decisions*" [44]. I have thus further subdivided this stage into 2 groups of tasks, namely, pattern discovery and the subsequent applications of the discovered information. These applications include classification, prediction, planning, diagnosis, structural modeling and tracking.

Patterns naturally occur in all types of real-life data, gathered from disciplines as diverse as economics, finance, medicine, demography, electronics and astronomy. Whenever patterns may exist, pattern discovery can assist in the interpretation of the underlying structure of data. Considering its place in the intelligent data analysis framework along with its broad spectrum of applicability, the importance of pattern discovery is undisputed. Human visual and perceptual abilities continue to play a major part in data analysis, as evidenced by graphical exploratory analysis and visualization. However, *automatic* (machine executed) pattern discovery techniques are needed in many real-world situations where, due to the aforementioned problems with the collected data, the analysis becomes too laborious and overwhelming for the human faculties. In recent literature, the term discovery is used synonymously with the term "knowledge discovery". Since there is still debate among experts as to what constitutes "knowledge" arising from data, I will exclusively use the more classical terminology of "pattern" and hence discovery will mean pattern discovery.

1.2 Ongoing issues and challenges

Despite widespread research, a number of challenges in pattern discovery still persist.

Fundamental information The first is a theoretical issue concerning the level (variable or event) at which information is extracted and described. Many methods used for pattern discovery, such as neural networks, cater only to variable level descriptions and more fundamental information is difficult to extract. This additional information often leads to a better overall interpretation of the data. To date, there is little theory for discovering patterns at the elemental level of events.

High-dimensionality The discovery of high-dimensional patterns with little prior information is an ominous task. An immediate problem is the sparsity of data in high dimensional space [104]. Small sample situations invalidate the wealth of asymptotic results that have been developed for statistical data analysis. Limited ability to explore surfaces of more than 5 dimensions [68] and the curse of dimensionality [13], i.e. the exponential growth of computational effort with increasing dimensionality, are some of the other dilemmas encountered.

Dynamic discovery Oftentimes, patterns change dynamically over time and a static description no longer suffices as the evolution of patterns becomes the object of interest. Inability to objectively detect and describe temporal changes have hampered the development of stochastic pattern discovery. Among the obstacles encountered are complicated update rules, unreasonably lengthy retraining sessions and restrictive parametric assumptions.

1.3 An event level approach

In this thesis, the emphasis is on the pattern discovery stage of intelligent data analysis. Preprocessing is not directly addressed although some of the developed techniques do provide preprocessing functions. Unlike many existing methods, an event rather than a variable level approach is adopted. The present research aims to establish the theoretical groundwork for event-based pattern discovery. Once the foundation is laid, specializations and extensions to discovery applications such as classification and density estimation, are investigated from an event perspective. Throughout the development, flexibility of discovery and transparency of the results are key requirements. The objectives of this thesis are formally stated in Chapter 3.

1.4 Summary of contributions

To draw this chapter to a close, I briefly outline what I feel are the contributions of this work towards the field of pattern discovery. A more detailed list appears at the end of this thesis.

A) Theoretical contributions. The main theoretical contribution is the development of a framework of events for continuous pattern discovery. This framework supports the existence of structure in less than the full dimensionality and is general enough to assist in the interpretation of other data analysis methods. Two new test statistics are derived for a unique application of a 2-way contingency table. Discovery in continuous data is cast as an optimization problem, a general formulation which holds for any discovery objective.

B) Methodological contributions. An enhanced recursive partitioning scheme is developed with self-adaptive parameters, avoiding ad-hoc settings of its predecessors. The novel ideas of recursive and sequential discovery are proposed as practical solution techniques to the optimization problem. High-dimensional analysis is addressed with a new bottom-up approach of event synthesis from low-dimensional projections. Under the common umbrella of events. a general kernel method is put forth and an objective measure of temporal change is developed. A variation of the parallel axes plot is conceived in order to enhance the visualization of multidimensional organization.

The bulk of the above items revolve around the common framework of events.

1.5 Chapter roadmap

Following this introductory chapter, there will be a survey of related methods which have been employed as pattern discovery tools. This review will span from classical to modern methods, emphasizing their merits and shortcomings as pattern discovery mechanisms. A formal statement of the thesis objectives and rationale constitute Chapter 3. In Chapter 4, the theoretical foundations of the pattern discovery framework are laid down. This chapter begins by refining some of the terminology introduced here, and moves on to formulate the discovery problem conceptually and mathematically. Chapter 5 extends and specializes the theory discussed in Chapter 4 to various scenarios. Throughout the presentation, a unified framework is maintained. The chapter concludes with an explanation of how discovered information can be easily interpreted. Some demonstrative and comparative experiments are reported in Chapter 6. Basic pattern discovery properties are highlighted along with a number of case studies. The final chapter concludes the thesis with a summary of contributions and indications for future ventures.

Chapter 2

Review of Discovery-related Methods

2.1 Overview

The thrust towards understanding data by uncovering hidden patterns and embedded knowledge is not a new research incentive. Indeed a multitude of techniques, variants and hybrids have arisen over the years in response to the universal need for data interpretation. Contributions to this area stem from a diversity of disciplines, including statistics, geology, psychology, economics, finance, medicine, engineering, astronomy and physics.

Categorizing the existing discovery methods is an arduous task, largely due to the different interpretations of what constitutes discovered information. Figure 2.1 is an arrangement of some generally accepted methods of discovering interesting information in data. The list is by no means comprehensive. With the exception of

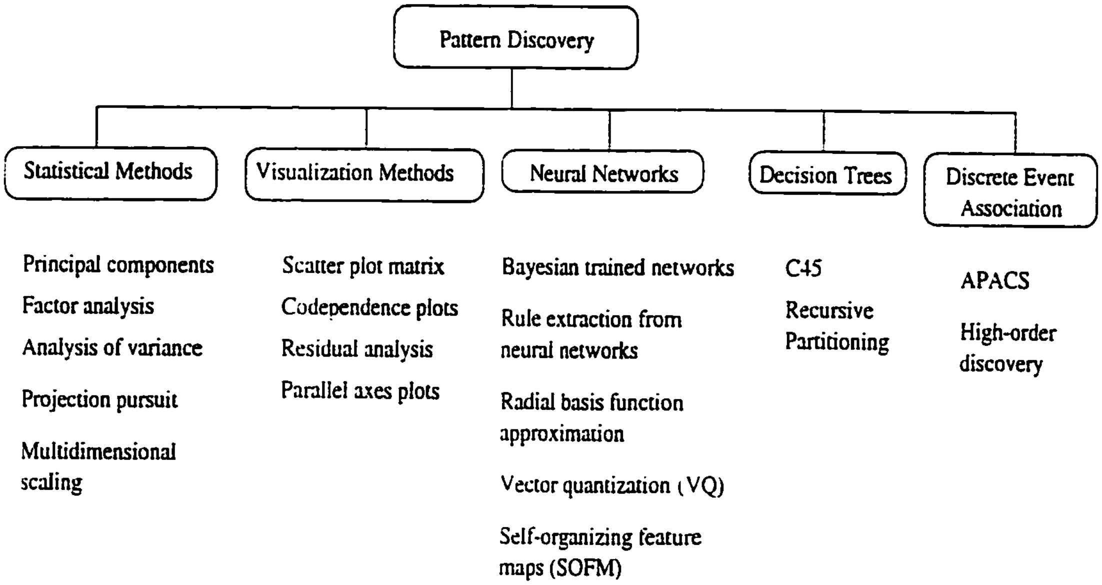


Figure 2.1: Pattern discovery methods classified by underlying techniques and tools

discrete event association methods, the categorized techniques focus on uncovering variable type dependencies. An alternative and insightful classification, shown in Figure 2.2, is obtained by considering the research question that is addressed, rather than the commonalities among the techniques. Some methods occur in more than one category as they are capable of fulfilling multiple pattern discovery objectives. The discovery objectives are also ranked according to the complexity of information sought after, with "functional relationship" [119] being the most complex, and "outliers" being the most elementary. Although the ranking is not absolute, it does provide a general idea of the hierarchy of information that is generally of interest. Again, the methods largely extract variable relationships, although some have straightforward event interpretations (Section 4.2.4).

Although this chapter reviews some related work, certain ideas are more appropriately presented along with the proposed theory and methods. Therefore, in the

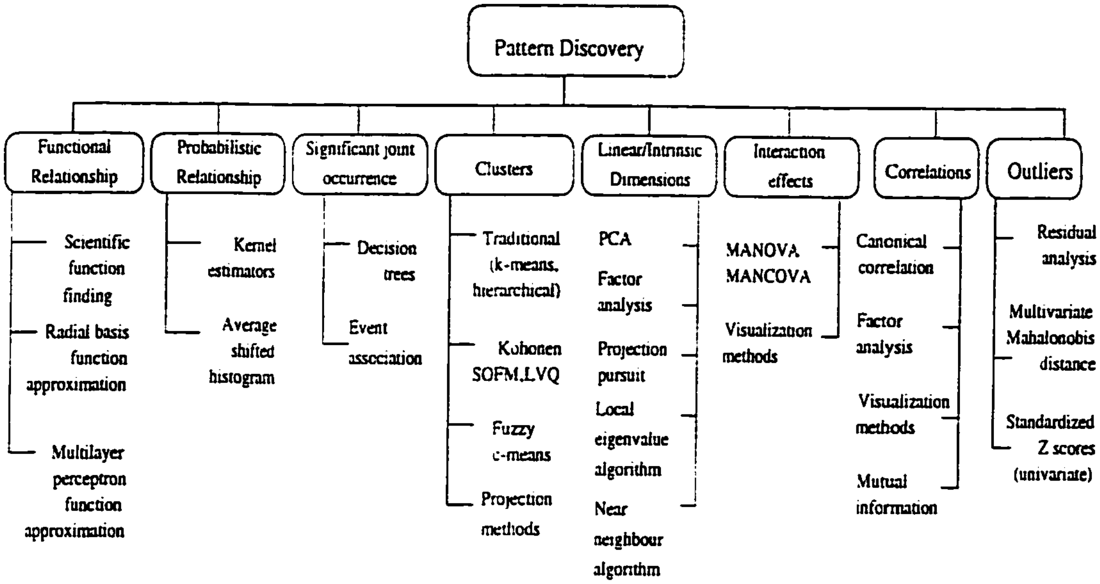


Figure 2.2: Pattern discovery methods classified by research question addressed

ensuing chapters, there will also be intermittent sections which will again revisit existing methodologies. Throughout the thesis, the reviews focus specifically on discovery methods for continuous data.

2.2 Visualization methods

Visualization is indispensable in exploratory data analysis. There is information revealed through graphical tools that cannot be easily detected by other means [30].

2.2.1 Scatterplot matrix

A scatterplot matrix is obtained by first graphing each pair of variables in a scatter plot. These plots are then arranged in a matrix so that along a given row or column, each variable is graphed against all the others. The key characteristic of scatterplot matrices is the redundancy of information achieved by duplicating graphs in the upper and lower triangles of the matrix. This redundancy is conducive to visual linkage of features among scatter plots. Correlations among variables and limited high-dimensional structure can be easily detected via a scatterplot matrix. As a starting point of analysis, the scatterplot matrix can also suggest potential relationships that may be investigated by another method.

The scatterplot matrix can be enhanced by *brushing* [32] or highlighting points in corresponding plots to reveal interplot linkage. However, pairwise scatter plots are limited to 10 variables, beyond which viewing becomes difficult. Another enhancement has been the *Grand Tour* [7], a smooth sequence of scatter plots produced by applying a continuous sequence of projection matrices. Unfortunately, it may take hours to unveil interesting projections in this manner. The main advantage of scatter plots is the symmetrical treatment of variables. Some 2-dimensional structures and clusters can also be detected. An obvious disadvantage is the difficulty of interpretation when categories of data overlap or when classes vary in sample size. Overplotting occurs with voluminous data and obscures structure. Geometrically, many different dissimilar d-dimensional data sets can give rise to visually similar scatter plots. Without further verification, a statement of the high-dimensional structure cannot be made conclusively. Due to distortions of high dimensional geometry, scatter plots exhibit an undue bias on tails of data [104]. Despite its popularity, simplicity and interpretability, scatter plots are not completely clairvoyant and might actually yield deceptive information about high-dimensional patterns.

2.2.2 Codependence plots

Codependence plots (coplots) are used to discover conditional dependencies among variables in a visually efficient manner [30]. Some of the variables are conditioned to lie within specific intervals and are represented in the *given panels.* In the *dependence panel* is a scatterplot matrix of the 2 variables under study. Each scatter plot in the dependence panel satisfies the conditioning imposed by the given panels. In this way, the effect of one variable on another can be isolated for analysis. Interaction among variables is also easily detected.

2.2.3 Residual analysis

Residuals are extremely informative in statistics and exploratory data analysis. Generally, residuals represent the variation in the data which is not accounted for by the assumed model. This interpretation of residuals is naturally conducive to pattern discovery. Indeed, by a careful choice of the model, deviations, as indicated by the residuals, can represent the discovery of useful information or patterns. The residual arises from the fundamental relationship,

$$\text{data} = \text{fit} + \text{residual} \tag{2.1}$$

where the fit is established by an assumed model [53]. In regression analysis, residual dependence plots convey valuable information about the adequacy of the fit. A sloping residual band indicates the presence of additional linearities in the data while a curved band suggests the prevalence of a nonlinear relationship [53]. Monotone spread in the data is conveyed by a wedge-shaped residual distribution.

The residual is also employed in other graphical and analytical tools. Spread-location plots are used to detect monotone spread in the data [30]. Residual-fit spread plots clearly compare the variation explained by the fit against that remaining in the residuals. Normal probability plots or quantile-quantile plots exploit the ordered sequence of residuals to detect skewness and outliers [53]. In categorical data analysis, residuals are used to detect outliers and to check for normality [26]. Particularly, in the former role, residuals can pinpoint cells which significantly depart from the assumed model. In pattern discovery, it has been recognized that model departures can often point towards interesting patterns in the data [23, 125, 25, 126]. In this light, residuals have been engaged in the discovery of sequential patterns in data [125], the estimation of missing measurements in power systems [36], the discovery of associations in continuous data [116] and the detection of high-order patterns in discrete data [128]. An advantage of residuals over other visualization techniques is that in addition to the transparency of residual plots, quantitative information is conveyed in the actual residual values.

Visualization methods are inarguably powerful discovery tools as they effectively exploit human pattern recognition abilities. However, with high-dimensional or extremely dense data, visualization may be severely hampered. Further, there is no guarantee that the human analyst will not overlook certain structure, due to fatigue, information overload, or the application of inappropriate tools. Clearly, considerable expertise is required in interpreting the observed graphs. Nonetheless, visualization provides a valuable initial step in many discovery problems.

2.3 Multivariate statistical methods

2.3.1 Analysis of variance: effects, interactions and strength of associations

A wide variety of variance analysis methods exists. These include the analysis of variance (ANOVA) and analysis of covariance (ANCOVA) and their multivariate counterparts, the factorial multivariate analysis of variance (MANOVA) and factorial multivariate analysis of covariance (MANCOVA). Basically, these methods rely on the computation of variances and covariances among variables to answer a number of research questions. Of particular interest to pattern discovery, is the question of whether independent variables (IVs) or their interactions exhibit significant effects on a dependent variable (DV) or group of DVs. Further, if these significant effects exist. then how strong are the associations. I will briefly discuss how analysis of variance methods answer these two questions.

The analysis hinges upon the fact that the total variance in the data set can be attributed to different sources of variation. In the multivariate case, variation is measured by a sum-of-squares and cross-products matrix, denoted as S. Generally, in the absence of covariates, the total sum-of-squares, S_{total} or the total variation in the data, can be decomposed as follows,

$$S_{total} = \sum S_{IV} + \sum S_{interact} + S_{error} \qquad (2.2)$$

where $\sum S_{IV}$ is the variation due to the IVs, $\sum S_{interact}$ is the variation due to the interaction of IVs, and S_{error} represents the within-group or error variation.

To test whether or not the IVs or their interactions significantly affect the DV or combination of DVs, one typically uses an approximate F-test based on Wilks Lambda [114]. Wilks Lambda, Λ, has the general form,

$$\Lambda = \frac{|S_{error}|}{|S_{effect} + S_{error}|} \tag{2.3}$$

where $|\cdot|$ denotes determinant. The determinant is the multivariate analog of univariate variance. The matrix S_{effect} is the effect under investigation. It could be the variation due to an independent variable (S_{IV}) or interaction of independent variables ($S_{interact}$). An approximate F-value can be computed from Λ via a rather intricate equation. See Tabachnick [114, p.387]. If the F-value exceeds the critical value at the chosen level of significance, then the effect in question significantly influences the DV.

Once a statistically significant effect is identified, we can gauge the strength of the association using the measure η^2, given as,

$$\eta^2 = 1 - \Lambda \tag{2.4}$$

This represents the fraction of the variance in the DVs that is accounted for by the effect under consideration. Hence, by examining variances alone, the MANOVA technique can reveal significant main and interaction effects on the DVs and give a quantitative assessment of the strength of these effects.

From a pattern discovery standpoint, multivariate analysis of variance answers the preliminary question of which variables influence each other and to what extent.

Although this is fairly low-level information, the analysis can point to directions for more in-depth study. The main limitation is the assumption that groups of data arise from normal populations with equal variances. Nonetheless, MANOVA is robust to mild violations of this assumption.

2.3.2 Principal components: discovering structure

Two closely related methods for revealing information about the latent structure in a multivariate data set are Principal component analysis (PCA) and Factor Analysis (FA). These are discussed in turn.

Principal component analysis is concerned with finding a set of uncorrelated indices that sufficiently describe the observed variation in the data. Ideally, the number of requisite indices for a sufficient description is much less than the original number of variables, and hence dimensionality reduction is achieved. Since these indices or components hint at the topological dimensionality of the data, they often allow easier interpretation of the data's structure, than in the original dimensions.

Suppose that the data is given in terms of the variables, $X_1, X_2, \ldots, X_p$. The analysis begins by computing the sample covariance matrix for the p variables. Let the eigenvalues of the matrix be $\lambda_1, \lambda_2, \ldots, \lambda_p$ and the corresponding eigenvectors, $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_p$. There will be p principal components Z_i, each expressed as a linear combination of the original variables X_i,

$$Z_i = \sum_{j=1}^{p} a_{ij} X_i \qquad i = 1, \ldots, p \tag{2.5}$$

where a_{ij} is the j^{th} element of the i^{th} eigenvector. The variance of the i^{th} principal component is $\mathrm{var}(Z_i) = \lambda_i$. This variance is the proportion of the variation in the data that is accounted for by the component. Components which account for small proportions of the overall variation in the data are discarded. In this way, PCA seeks out the "dimensions" of greatest variation. Note that if the original variables were coded to have 0 means and unit variances, then we would be dealing with the correlation rather than covariance matrix. Recently, it has been shown that multilayer perceptron autoassociators with one hidden layer also perform linear PCA [40]. Some neural network implementations of PCA are reviewed in [83].

From a pattern discovery perspective, PCA is advantageous in that the number of dimensions to be simultaneously considered is reduced. The extracted principal components are useful for understanding the underlying structure in the data. However, if the variables are nonlinearly related, interpretation of the principal components can be difficult. Because PCA effectively projects the original data onto dimensions of maximal variance, it is often used as a preprocessor to classification where separability is of prime interest. However, due to the emphasis on variance, principal components critically depend on the units used. Generally, all features need to be sphered to have unit variance.

2.3.3 Factor analysis: discovering structure

Factor analysis (FA) is similar to principal components in that it uncovers a reduced set of common "factors" to explain the data. Unlike PCA which analyzes all the variance in the observed variables, FA only considers the variance that is shared among the variables.

Suppose the original variables are $X_1, X_2, \ldots, X_p$. The observed values for each variable are standardized to have mean 0 and unit variance. Factor extraction yields an initial set of factors, described by a factor loading matrix A. This key matrix contains the correlations between each factor and each variable. Factor rotation tries to enhance high correlations while diluting low correlations. The resulting factor model is given by.

$$X_i = \sum_{j=1}^{m} a_{ij} F_j + e_i, \qquad i = 1, \ldots, p \tag{2.6}$$

where a_{ij} are the factor loadings for the i^{th} variable, X_i and the F_j are m uncorrelated common factors with 0 mean and unit variance. The last term, e_i represents a factor that is specific to the i^{th} variable. Note that the number of factors m, is less than the number of original variables, p. Hopefully, the m factors are easier to interpret than the original p variables. By looking at the final factor loadings, the analyst would attempt to assign meaningful interpretations to the factors. Ideally, a factor is easily interpretable when several observed variables correlate highly with it but do not correlate with other factors. In fact, the effect of factor analysis is to group together variables that are correlated.

From a pattern discovery perspective, FA attempts to unveil the underlying structure in data. Like PCA, it is not very effective when the data is not well approximated by a linear model. The ultimate evaluation of a factor analysis is the interpretability of the final factors. Hence, the success of this mode of discovery hinges heavily on human creativity in deriving meaningful interpretations of the factors.

2.3.4 Projection pursuit: discovering structure

Projection pursuit (PP) [48] searches for the most "interesting" low-dimensional linear projection of high-dimensional data. It is a numerical optimization problem with the objective of finding a projection axis $\hat{k}$ to maximize a projection index. The original index of Friedman and Tukey [48] was of the form,

$$I(\hat{k}) = s(\hat{k})d(\hat{k}) \tag{2.7}$$

where the spread of the data is measured by $s(\hat{k})$ and $d(\hat{k})$ describes the local density of points after projection onto $\hat{k}$, the projection axis. Maximizing I corresponds to finding a projection in which the data are simultaneously concentrated locally (large $d(\hat{k})$) and expanded globally (large $s(\hat{k})$). Equivalently, projection pursuit finds a direction of maximum variance of the data while trying to preserve their interpoint distances.

Several different projection indices have been proposed, including the standardized Fisher index, negative Shannon entropy, L_1 index and the Hellinger index. All incorporate the dual local-global optimization objective. The accepted meaning of "interestingness" is the departure from multivariate normality [68].

Unlike PCA, PP gives some consideration to local variation. However, it too has several drawbacks. PP is sensitive to the scaling of the data and experiences difficulty in detecting structure in highly curved surfaces [48]. As a constrained nonlinear optimization problem, it is computationally intense and often encounters suboptimal local maxima [101]. Huber [68] remarked that in high dimensions, a large number of points is required to ensure that discovery reveals the underlying structure rather than quirks of the sample space. Many projection indices are difficult to evaluate when the projection space is of more than 1 or 2 dimensions and indices such as the Legendre index and Hermite index are very sensitive to outliers [101]. To detect high-dimensional structure, many careful decisions must be made: the dimensionality of the projection subspace, a characteristic radius which determines the algorithms sensitivity to local variations and the metric to measure interpoint distances. In terms of advantage, the PP algorithm does not rely on a single projection as in PCA. It can be applied recursively to decreasing subspaces to uncover multiple levels of structure. However, the choice of subspace must be made manually. In many applications, projection pursuit serves primarily as a powerful visualization tool for seeking outliers [101] and natural groupings in data. The only disadvantage is that the potentially large number of different views can be overwhelming.

2.4 Decision tree methods

Tree-based methods have mainly been used for classification problems. The rules generated from tree-construction are often interpretable and thus as discovery mechanisms, they offer insight into the structure of the data. Generally, a classification tree consists of a top node or root and branches off to many subsequent nodes at which decisions are made. Decisions continue until a terminal node or leaf is reached. A classification tree partitions the sample space into sub-regions corresponding to the leaves. Hence, a tree is a naturally hierarchical way to describe the partitioning of the sample space.

The task of constructing a decision tree from a set of examples is known as tree induction. The key considerations in tree induction are pruning strategy and prescription for splitting nodes. The process of splitting at a node is driven by an impurity measure. The impurity measure is an indication of the class composition at each node, with a single class composition being 0 impurity. Common impurity measures include the entropy and the Gini index. Pruning is needed to efficiently choose a rooted tree from among the large number of possibilities. The pruning process is typically propelled by some measure of the classification error rate and the number of leaves in the tree. The aim is to simultaneously minimize error rate and tree size.

Trees have been applied in 3 general areas [101] outside of its social science origins, namely, statistics, machine learning and engineering. In machine learning, the use of trees is closely tied to discovery. Specifically, the objective is to induce a set of rules from a data set, either directly, or by way of an induced tree. Examples of such methods are ID3 [96], its descendant C4.5 [97], ASSISTANT and the family of CLS [94] systems. Only recent versions of some of these algorithms can accommodate continuous data. In engineering, a discovery-related application of trees is the recursive partitioning of the sample space for the generation of invariant decision rules. Some prominent works include Henrichon and Fu's frequency equalization partitioning [64] and Friedman's binary, recursive scheme [47]. Common strengths of past partitioning methods are scale invariance and local representation of the feature space. These properties offer solutions to nonlinearly separable decision regions. The underlying shortcomings include the need to determine problem dependent parameters and the inability to effectively screen out irrelevant data.

The natural ability to handle missing data and the relative ease of interpreting the acquired "knowledge" are the key strengths of trees as discovery tools. However, for noisy data sets, trees tend to become very large, very difficult to prune [101] and classification performance degrades.

2.5 Neural network rule extraction

The literature on neural networks is of titanic proportions. Fortunately, here I will only focus on recent *discovery* approaches, namely the excavation of interpretable rules from an artificial neural network trained in a supervised fashion.

It has been shown that, assuming an unlimited number of sigmoidal hidden units, a 3 layer network can approximate any continuous function to arbitrary accuracy [78]. In a similar spirit, radial basis functions have also been proven to be universal function approximators [60, 91]. In other words, these networks can discover any arbitrary functional mapping between independent and dependent variables in a given data set. However, from the standpoint of readability [89], the trained network parameters offer little understanding and interpretation into the "knowledge" acquired. Narazaki [89] attributes this lack of readability to the inherent distributed representation of the neural network. This deficiency in "explanation capability" [5] has spawned the recent work in rule extraction from trained networks.

In a recent survey, Andrews et. al. [5] proposed expressive power and translucency as the principal criteria for classifying strategies for rule extraction from trained neural networks. Three categories were identified:

1. Boolean rule extraction using decompositional approaches

   Techniques in this category seek Boolean rules at the level of the individual hidden and output units. Particularly, for a unit under study, a rule is generated from a connection whose summed weights guarantee that the unit's activity exceeds its bias. The rules from individual units are aggregated to form a composite rule base.

2. Boolean rule extraction using pedagogical approaches

   According to Andrews et. al., this set of techniques treat rule extraction as a learning task, with the network function as the target concept and the inputs as the network's inputs. The extracted rules directly map network inputs into network outputs.

3. Fuzzy rules in neurofuzzy systems

   Strategies in this group perform fuzzy rule refinement by tuning membership functions and modifying connection weights. Two prominent examples are the tuning of fuzzy associative memories [76] and fuzzy logic controllers [130].

Andrews et. al. observed that rule extraction occurs only after network training but not during training. Hence, there may be some redundancy in the 2 processes. Unfortunately, like many training algorithms, rule extraction algorithms are also computationally expensive. Furthermore, the majority of rule extractors rely on some form of ad-hoc heuristics to constrain the rule space. A number of studies have reported certain inconsistencies between the classification performance of the extracted rules and that of the network itself [117, 50]. These unexpected results have raised some theoretical queries [5].

In more recent work, Setiono and Liu [106] developed a neural rule generator, NeuroRule, that alleviated some of the aforementioned complications. Their technique consisted of 4 phases: the building of a weight decay network, pruning the network, discretizing hidden unit activations and rule generation. The main advantages of NeuroRule are as follows.

- Unlike its predecessors, NeuroRule does not place restrictions on the hidden unit activation values. Theoretically, rules can then be a more faithful reflection of the network operation.
- Through experiments on machine learning data sets, it was shown that extracted rules maintained classification accuracy of the network itself, and generally exceeded that of equivalent decision tree classifiers.
- The number of antecedents in the extracted rules are not limited, as in some of the decompositional approaches classified by Andrews et. al.

Exploiting both local and distributed representations, Narazaki et. al. [89] developed a fuzzy rule generation system that directly analyzes the network function rather than the network weights. Rules were derived by first identifying homogeneous, "monotonic" regions of the input space and secondly by projecting a hyperrectangle estimate of the region onto the individual axes. The proposed idea of relating local intervals for class discrimination is philosophically akin to event level analysis. Unfortunately, apart from 2 simple pedagogical examples, no experimental results were presented.

In short, neural rule extraction seems to be a promising alternative for pattern discovery and data-mining. Indeed, there would be a tremendous impact on data analysis in general, if the internal representation of the neural network could be fully interpreted. Nonetheless, consistency between the rules and the trained network have yet to be theoretically established.

2.6 Nonlinear projection methods

Nonlinear projection methods, like their linear counterparts (PCA and PP), attempt to map the original data or some computed characteristic (e.g. interpoint distances) from the original high dimensional space into a lower dimensional, more easily interpretable space. The advantage of nonlinear projections is the potential of revealing more general relationships in the data. Both statistical and neural methods are used to this end. Some popular examples are briefly introduced here.

A common nonlinear projection neural network is the bottleneck or autoencoder network. This is typically a five layer network with linear outer and middle layers while the second and fourth layers have sigmoidal transfer functions. The two outer layers have p units while the middle layer has q units, with $p \gg q$. By simultaneously applying the p dimensional original data to both outer layers, the network is forced to learn the mappings,

$$\Re^p \longrightarrow \Re^q \quad \text{and} \quad \Re^q \longrightarrow \Re^p \tag{2.8}$$

Ideally, the q dimensional result will be more revealing than the original data.

A self-organizing feature map [75] projects similar examples onto contiguous locations in a one or two dimensional topological output space. The mechanism of projection is soft competitive learning. When an example (data point in $\Re^d$) is presented to the network, the connection weights of the closest representative (neuron) along with its neighbours in a predefined vicinity are adjusted. In brief, the attained mapping is

$$\Re^d \longrightarrow \Re^1 \quad \text{or} \quad \Re^2 \tag{2.9}$$

In a similar vein to the self-organizing map, learning vector quantization [74, 75] achieves nonlinear dimensionality reduction by hard competitive learning. In this approach, d dimensional data is mapped to q dimensional vectors or codes, where $q \ll d$ and q need not be restricted to one or two. Unlike soft learning, only the neuron with the nearest code is updated, leaving its neighbours unaffected. The achieved mapping is thus,

$$\Re^d \longrightarrow \Re^q \tag{2.10}$$

The Sammon mapping [103] is a special case of multidimensional scaling [82]. The idea is to compute interpoint distances in the original data and to map this set of distances into a corresponding set of distances in a lower dimensional space. The objective of the projection is to minimize stress, defined as,

$$E = \frac{1}{\sum_{i<j}^{N} d_{ij}^{*}} \sum_{i<j}^{N} \frac{(d_{ij}^{*} - d_{ij})^2}{d_{ij}^{*}} \tag{2.11}$$

where d_{ij}^{*} is the distance between the i^{th} and j^{th} points in the original dimensionality,

d_{ij} is the distance between the same two points in the lower dimensional space and N is the total number of data points. This type of mapping maximally preserves interpoint distances.

2.7 Spatial analysis

Spatial data analysis is employed in the detection of clustering or regularity in data distributed in $\Re^d$. The general spatial model [35] is specified by spatial locations $\{s_1, \ldots, s_n\}$ and data $\{Z(s_1), \ldots, Z(s_n)\}$, where $Z(\cdot)$ is often a random mapping. Depending on the definition of the locations $\{s_I\}$ (e.g. continuously varying over $\Re$ or restricted to lattice points), this model can accommodate the analysis of geostatistical, lattice or point data. The analysis of point patterns is most relevant to pattern discovery in a continuous sample space.

With point data, complete spatial randomness is defined as a homogenous Poisson process in $\Re^d$. This spatial process has the property that the events in a bounded region $E \subset \Re^d$ are independently uniformly distributed over E. In other words, the events are equally likely to occur anywhere within E, without any interaction. For example, suppose a bounded region A is under analysis. The probability that a point $x \in \Re^d$ falls within a subregion $E \subset A$ is

$$P(x \in E) = \frac{\text{volume of } E}{\text{total volume } A}. \tag{2.12}$$

This definition of randomness closely coincides with the concentration discovery hypothesis formulated later in this thesis.

The goal of the analysis is to summarize the spatial data by way of descriptive statistics, which in turn suggest appropriate models for the data. Two main mechanisms for obtaining summary statistics of spatial data are quadrats [118, 102] Moreover, inter-event distances. In both cases, point patterns are defined as regularities or clusterings in the location of spatial events. These patterns are detected by testing the data against the complete spatial randomness hypothesis. If the hypothesis is rejected, the subsequent step is to measure the magnitude of deviation from randomness.

Quadrats are well defined geometrical shapes (usually rectangles), placed either randomly or contiguously over the study region. The number of events falling within each quadrat is enumerated yielding observed frequencies. Using a χ^2 test, these observed counts are statistically compared to a set of expected frequencies obtained under the assumption of the complete spatial randomness model. A significant difference indicates departure from randomness. Numerous indices have been developed to gauge the amount of deviation from randomness, including the relative variance, David-Moore index, ICS, mean-crowding and Morisita's index [100]. From a pattern discovery perspective, quadrat-based analysis only detects global but not local patterns. Further, the definition of pattern or useful information is restricted to spatial clustering. Functional and general relational behaviour cannot be detected. In addition, the quadrat size and shape are chosen arbitrarily and spatial information is inevitably lost in the summary statistics.

Distance methods for spatial data are founded upon the definition of a handful of different nearest-neighbour distances. For example, the distance D between events and nearest neighbouring events has a known distribution under the complete spatial randomness hypothesis. In particular, the density is

$$p(D) = 2\pi\lambda D \exp(-\pi\lambda D^2), \quad D > 0 \tag{2.13}$$

where λ is the intensity of the homogenous Poisson random process. With knowledge of the distance distributions, complete spatial randomness can be tested using a number of test statistics [35, p.604], [118, p.242-263]. The shortfall of spatial distance methods for pattern discovery is that the definition of pattern relies heavily on the number of nearest neighbours considered. With a single neighbour, only small scale patterns are considered while larger scale information is unavailable [35]. Again, spatial information is lost in retaining only a nearest neighbour summary. The nearest neighbour statistics can only indicate the direction of departure with no additional information as to the nature or location of the discrepancy.

Comment on Discrete Discovery Methods

Although, with some preprocessing, discrete data may be analyzed, the techniques reviewed above are all geared towards continuous data. There exist a whole realm of methods designed specifically for discrete-valued, categorical and ordinal data. Belief networks (See Ripley [101] for a review) such as Markov networks or Causal networks, APACS [23], discovery of association rules [1], AQ17 [20] and CN2 [29] are among the many existing discrete discovery methodologies. Here, I restrict the review to the treatment of continuous data, which is the principal concern of this thesis.

2.8 Summary

In this chapter, I have briefly reviewed a sampling of techniques from the main areas of visual exploratory data analysis, multivariate statistics, decision trees and neural networks. Visualization is limited to a small number of dimensions and statistical methods often impose a set of limiting parametric assumptions. Decision trees are not rooted in deep theory [101] but rely on many ad-hoc heuristics. Rule extraction from neural networks are computationally intensive tasks and the number of rules can often be unmanageable. Nonetheless, each genre of analysis offers some insight into the discovery problem and potential solutions. Visualization emphasizes the need for interpretability and transparency. Statistical approaches underscore the presence of intricate associations and interactions among variables and offer ways to overcome the curse of dimensionality. Hierarchical partitioning of the sample space is an intriguing prospect for local discovery that is suggested by the decision tree methods. Finally, the generation of readable rules from complex neural networks points towards the existence of a fundamental level of data organization. Some of these ideas and implications will be explored in the development of an event-based pattern discovery framework.

Chapter 3

Problem Definition

From the review, it is evident that the scope of the pattern discovery problem is quite broad. In this brief chapter, I attempt to succinctly define the focus of the present research.

3.1 Data specification

The main specification is with regards to the type of data targeted for analysis. The event level theory should be applicable to both discrete and continuous data. In terms of algorithmic development, the data should meet the following specifications.

Type The data should be continuous. If it is discrete, the variables should have a large number of values such that they can be effectively treated as continuous (e.g integer test scores from 0 to 100).

Range The range of possible data values is unrestricted.

Missing Values Missing values are tolerated if they are few and infrequent. Grossly incomplete data is not admissible. [1]

Format Typically, with multidimensional data, an observation or measurement is arranged into a vector of numbers or symbols. Each element of the vector represents the value of a physical entity that has been observed or measured. The physical entities of interest are called features in pattern recognition, attributes in machine learning and factors or variables in statistics. Throughout this thesis, these terms will be used interchangeably and I will assume that data is organized in a vector format.

3.2 Rationale for present research

Traditionally in pattern recognition, random variables and random vectors have been used as mathematical representations of patterns [49, 131, 41]. From probability theory, we know that associated with each random variable value is an event [90]. Since the relationships among random variables are used to express the overall structure of the data, it is natural to turn to events for a deeper view of that structure.

With discrete data, an event level theory for pattern discovery is straightforward since the individual observations constitute elementary events. The developments of [23, 125, 128] exemplify the maturity of the discrete event theory. For continuous data, an event formulation for discovery is not yet established.

[1] Many strategies are dedicated to the handling of more troublesome and incomplete data. See for example [71, 85].

A discovery strategy and theory should accommodate multivariate data. In the analysis of multidimensional data, one usually relies on several independent sources of evidence to validate hypotheses one may have about the data's structure. Hence, an additional perspective is always welcome.

To be useful for decision making and problem solving, the detected patterns should be easily understood. Transparency also permits easy verification with domain knowledge and comparison to the findings of other methods.

We have seen that numerous different criteria are of interest in data analysis, for example, the maximal local data concentration, the maximum data variance and the minimum class impurity. A useful discovery algorithm must therefore have the flexibility to support different discovery objectives.

If a framework of discovery is to be general, it should be extensible to many problems in data analysis which directly or indirectly make use of patterns in the data.

3.3 Objectives

Formally, the objectives of the present research are:

1. To propose a theoretical framework for pattern discovery at the event level for multivariate continuous data.

2. To develop a transparent pattern discovery algorithm for multivariate continuous data. The algorithm should yield interpretable patterns while accommodating different discovery objectives, and

3. To apply the discovered patterns to problems of multivariate density estimation, multicategory classification, dynamic pattern discovery and high-dimensional discovery.

Chapter 4

Pattern Discovery Foundations

In this chapter, I will lay down the definitions and conceptual arguments which comprise the proposed event-based theoretical construct. The generality of the formulation will be argued by exemplifying its interpretation in the context of some existing methods. Different discovery objectives are introduced along with appropriate test statistics. The chapter will close with a mathematical statement of the discovery problem.

4.1 Generalized events

The notion of an event was vaguely alluded to in the introductory chapter. Here, the concept will be formalized. Note that the definitions that I invoke here are generally consistent with the terminology of probability theory, particularly in the axiomatic definition of an experiment [90]. Minor departures will be noted. In pattern discovery, we are concerned, in general, with d-dimensional data. To draw an equivalence with probability theory, we simply consider a d-dimensional data point, either an observation or a measurement, as the outcome of an imaginary experiment. Denote an experimental outcome as ω. The set of all possible outcomes is the sample space, Ω. and may be finite or uncountably infinite. In each case, events are defined differently. Although continuous data is assumed in this thesis, the discussion of events for discrete data enforces the generality of the event concept.

4.1.1 Events - finite sample space

When the sample space is finite, an event is defined as a subset of Ω. This subset may consist of a single event in which case it is called an elementary event [90]. If the subset is empty, it is called a null event. For events in a finite sample space, probabilities are assigned to elementary events. The axioms of probability are then easily satisfied. In pattern discovery, this definition of events applies when the data of interest is discrete. The discrete nature of the data may be manifested in 2 ways, a finite number of numerical values or a finite number of categorical levels. In the latter case, the data is symbolic. The following example illustrates an event for discrete-valued data which consists of both numeric and symbolic features.

Example 1 *Events in discrete data*

Consider a study on birds where the features of interest are habitat, the number of toes, the colour of the bird's chest and its ability to fly. These features can take on values from a finite set, as indicated below.

Feature	*Allowable values*
Habitat	*urban area, inland waters, grassland, desert, brushy areas, woodlands*
# of toes	*2,3,4*
Breast colour	*red, white, grey, spotted, blue, yellow*
Flight	*yes, no*

Examples of events from the sample space of possible combinations of feature values are listed below.

Event	*Habitat (Symbolic)*	*# of toes (Numeric)*	*Breast colour (Symbolic)*	*Flight (Binary)*
1	*urban area*	*4*	*red*	*yes*
2	*inland waters*	*3*	*white*	*yes*
3	*grassland*	*2*	*grey*	*no*
4	*desert*	*4*	*white*	*yes*

Random variable

In pattern analysis, tools have been developed to manipulate random variables rather than elementary events or outcomes. The outcomes are mapped into a space in which the operations of discovery are more easily performed. The chosen mapping becomes the random variable for the problem at hand. Since there are a finite number of outcomes, the resulting random variable can only assume a finite number of distinct values, meaning that it is a discrete random variable. There are 3 possible mappings depending on the nature of the data and the processing ability of the analysis method.

1. When the discrete data is numerical, the random variable can simply be the unity mapping.

2. If the data is categorical and the pattern discovery method can directly handle symbolic variables, the unity mapping can also be applied.

3. On the other hand, if the data is categorical but the discovery method can only manipulate numbers, we require a transformation from the events to a finite set of integers. For categorical data, a mapping of the form

$$X : \omega \longrightarrow i, \quad \omega \in \Omega, \quad i \in \Gamma \subset I \tag{4.1}$$

   is usually employed. Here, ω is an elementary event and the image space Γ is a finite subset of the set of integers, I.

In Example 1, if the symbolic variables were mapped into a set of integers, each event would be a distinct point in I^4.

Table 4.1 summarizes the different discrete random variable mappings discussed. In general, the choice of $X(\cdot)$ only needs to satisfy the usual axiomatic conditions [90, p.66]. However, for an event level formulation, we also demand that the inverse transformation exists. In practice, this requirement stipulates that $X(\cdot)$ should be 1-to-1. To understand the rationale for this additional constraint, we need to realize that discovery could be performed in either the sample space, Ω, or the image space, Γ, depending on the discovery algorithm. The existence of $X^{-1}(\cdot)$ ensures that information about events is always available, regardless of the space in which discovery is performed.

Table 4.1: Summary of random variable mappings for discrete-valued data

Discrete data type	Variable type that discovery method can can handle	Random variable mapping
Numeric	Numeric	Unity mapping $X : \omega \rightarrow \omega$
Symbolic	Symbolic	Unity mapping $X : \omega \rightarrow \omega$
Symbolic	Numeric	General, invertible $X : \omega \rightarrow i \ , i \in \Gamma \subset I$

4.1.2 Events - infinite sample space

In this thesis, I am mainly concerned with continuous data, for which the sample space is generally taken to be the d-dimensional Euclidean space, $\Re^d$. Clearly, this sample space consists of an uncountable number of outcomes. The use of elementary events becomes intractable, especially when defining probabilities [90]. Instead, events are defined to be subsets of Ω which form a Borel σ-field, $B(\Re^d)$. To clarify, we need a number of definitions. Let $\mathcal{Z}$ be a nonempty set of arbitrary elements. A class of sets in $\mathcal{Z}$ is a collection of subsets of $\mathcal{Z}$. A field is a nonempty class closed under all finite set operations (union, intersection and complementation). A σ-field is a nonempty class closed under all countable set operations. The canonical σ-field for $\Re^d$ is the Borel σ-field.

Definition 1 *Borel σ field of $\Re^d$ [95]*

Consider the sample space $\Re^d$. Let $\infty < a_i < b_i < \infty$, $i = 1, \ldots, d$. Let $I_i = (a_i, b_i]$.

A subset A of $\Re^d$ is called a rectangle *if it has the form,*

$$A = I_1 \times \cdots \times I_d = \{\mathbf{x} : x_i \in I_i, 1 \leq i \leq d\} \tag{4.2}$$

The σ-field generated by the collection of all rectangles in $\Re^d$ is the Borel σ-field of $\Re^d$, $B(\Re^d)$.

Equivalently, the Borel σ-field is the smallest σ-field containing all rectangles. The sets in the Borel field are called Borel sets. Borel sets include rectangles and countable unions and intersections of rectangles. We are now ready to define an event in the continuous sample space. $\Re^d$.

Definition 2 *An event in $\Re^d$ is a Borel set.*

There are two advantages of defining events in this way. First of all, there is a nice geometric perspective. Figure 4.1 depicts examples of events in $\Re^3$. We see that events are just hyper-rectangles or countable unions or intersections of hyper-rectangles. Secondly, a probability measure can now be assigned to events without violating the axioms of probability.

Random variable

As in the case with discrete data, technically we cannot talk about random variables until we have mapped the outcomes into numerical values. Fortunately, for the sample space $\Re^d$, the outcomes are already continuous numerical values. Hence, we can simply employ the identity mapping,

$$X : \omega \longrightarrow \omega, \quad \omega \in \Re^d \tag{4.3}$$

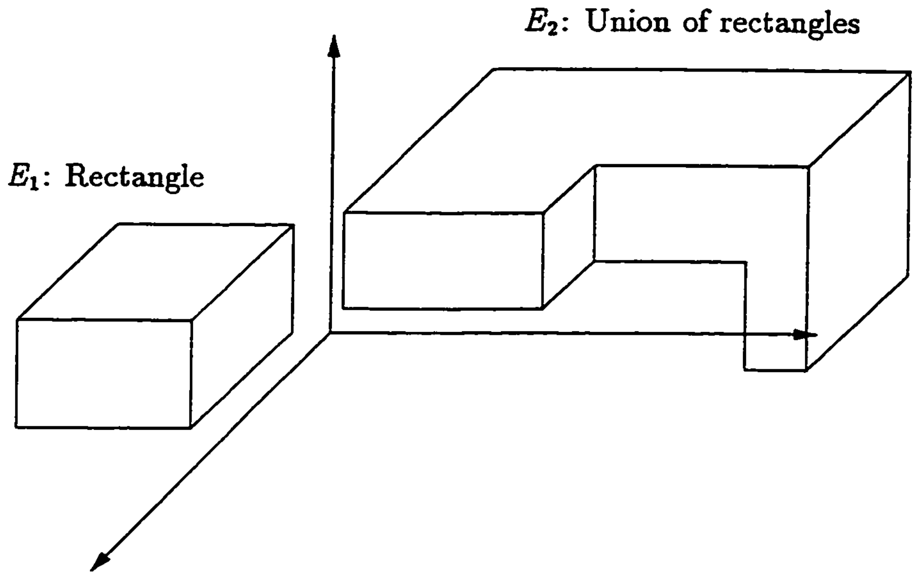


Figure 4.1: Example of events in $\Re^3$

to obtain a continuous random vector X. The identity mapping has an important implication for pattern discovery, namely that ω is both the outcome of the imaginary experiment and the corresponding value of the random vector. This nicety will be exploited in formulating the idea of event-based discovery.

To summarize, Figure 4.2 pictorially represents the general mapping from the sample space Ω to an image space Γ. Note that typically, discovery algorithms operate in the image space Γ, so that random variable terminology and properties can be employed. For discrete data, Γ is a subset of integers whereas for continuous data, Γ is the real line. The event E in Ω is an elementary event for discrete data and a Borel set for continuous data.

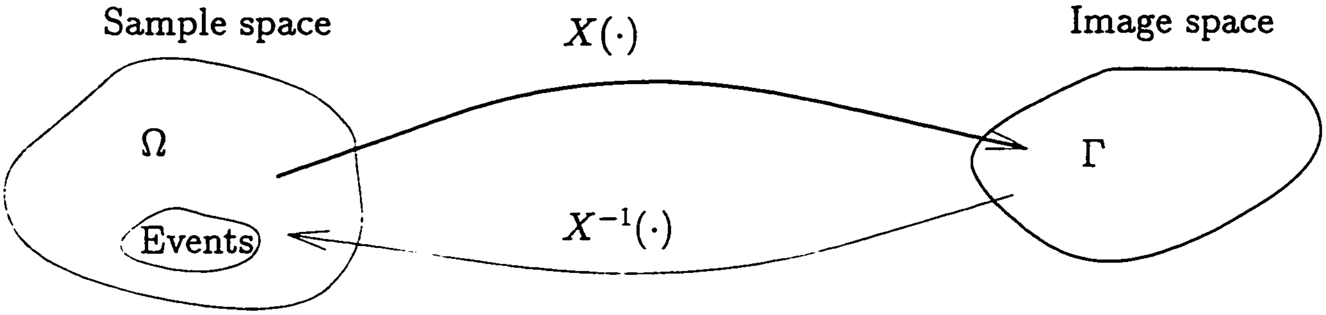


Figure 4.2: General random variable mapping for pattern discovery

4.1.3 Event characterizations - infinite sample space

Apart from its location in space, a few other quantities will be used to characterize events in $\Re^d$.

Definition 3 *Volume*

The volume of an event is the hypervolume of the d-dimensional subspace occupied by the event.

From a measure theoretic viewpoint, if we consider the event to be a set, then the volume is a measure of the set.

The next definition deviates slightly from the analogous concept in probability theory. Suppose we have a data set $\{\omega\}$ from a sample space, $\Omega \subset \Re^d$.

Definition 4 *Observed Frequency*

The observed frequency. n_E, of an event E in the sample space Ω is the number of data points that fall within the volume of E. If we denote $\{x\} \subset \{\omega\}$ as the finite set of points falling inside the volume of E, then $n_E = |\{x\}|$, where $|\cdot|$ denotes cardinality.

For the sake of notational simplicity, I will assume that the event E is synonymous with the set of points that it contains. Hence, the observed frequency will be directly written as, $n_E = |E|$. I will interchangeably refer to n_E as the cardinality of event E and as the observed frequency of event E.

In probability theory, the observed frequency of an event is the number of times the event is observed in an experiment. To consolidate the 2 definitions conceptually, consider a hypothetical experiment where each observed data point in Ω induces an event. Now suppose that there is a set of points $\{\mathbf{x}\}$, each of which induces the same event E. If we consider events rather than data points as the experimental outcomes, the number of times E will be observed is exactly equal to the number of points in $\{\mathbf{x}\}$. Therefore, $n_E = |\{\mathbf{x}\}|$.

Definition 5 *Probability*

The probability of an event E is intuitively estimated by the proportion of data points contained in the event.

$$\hat{P}_E = \frac{n_E}{N} \tag{4.4}$$

Equivalently, this is the probability of finding an outcome within the subspace defined by E. The fact that events have been defined as Borel sets allows us to construct this probability measure without fear of violating probability axioms.

4.2 Pattern

With the formal definition of an event in place, the notion of a pattern can now be made precise.

Definition 6 *Pattern*

Let Ω denote the sample space and let $g(\cdot)$ be a test statistic corresponding to a specified discovery criterion. Let θ_c^α be the critical value of a statistical test at a significance level of α. A pattern is an event E that satisfies the condition,

$$g(E) \geq \theta_c^\alpha \tag{4.5}$$

The test statistic $g(\cdot)$ measures the degree to which an event satisfies the objective of our discovery. Different discovery objectives and their corresponding test statistics will be discussed in Section 4.3. In this section, some general remarks are made about the test statistic and its interpretation.

If the test statistic is 2-tailed, 3 types of events can be identified, according to the value of the test statistic.

1. $g(E) \geq \theta_c^\alpha$

   When the test statistic equals or exceeds the critical value, the event E adequately satisfies the discovery objective and is called a *positive significant event.* A pattern, as defined above, is a positive significant event.

2. $|g(E)| < \theta_c^\alpha$

   Here the test statistic indicates that the event does not satisfy the discovery objective at the chosen level of significance. We label the event as an *insignificant event.*

3. $g(E) \leq -\theta_c^\alpha$

   The last type of event is called a negative significant event. These events are contrary to the discovery objective.

With a 1-tailed test statistic, only insignificant and positive significant events are applicable. When there is no ambiguity, the term, "*significant event*" will be taken to mean a positive significant event. What follows is another useful interpretation of a pattern which I will refer to on occasion.

Event association

Any event E of dimension $d > 1$ can be interpreted as the joint occurrence of lower dimensional events. For example, in Figure 4.3, the event E can be viewed as the joint occurrence of three 1-dimensional intervals, I_x, I_y and I_z. These 1-dimensional intervals are themselves Borel sets in $\Re$ and therefore valid events. This leads to the following simple definition.

Definition 7 *Event Association [128]*

An event association is a significant joint occurrence of low-dimensional events. In particular, any d-dimensional event ($d \geq 2$) can be considered an event association, composed of d 1-dimensional events.

According to defintions 4.2 and 2, an event E uniquely determines a set of one-dimensional intervals. However, the event E may also be interpreted as the joint occurrence of a nonunique set of lower dimensional events, each with $d > 1$.

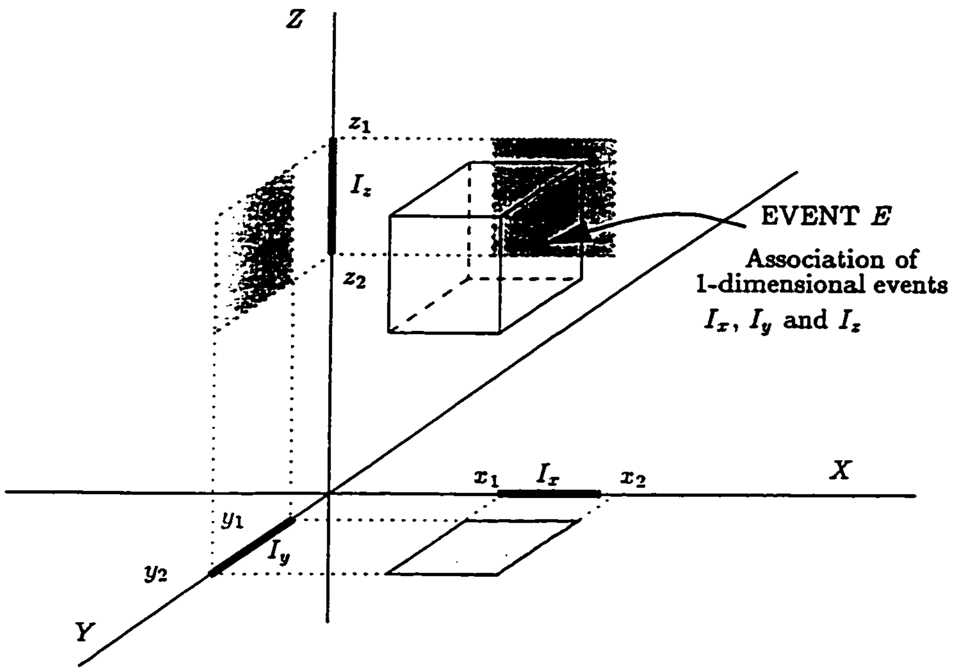


Figure 4.3: A 3-dimensional event association

Note that the term "association" should not be confused with the association between dependent and independent variables as measured by η^2 in statistical analysis of variance (ANOVA) studies. Nonetheless, both usages do imply an inherent relationship between objects: variables in ANOVA, and events in pattern discovery.

We appreciate that the jargon, "pattern", "significant event" and "event association" all share the same meaning, with only variations in interpretation. While the word "pattern" has intuitive appeal, its statistical basis is intimately implied by the term "significant event". On the pragmatic front, "event association" offers a geometric perspective, which will be useful in interpreting discovery results.

4.2.1 Geometry of a significant event

The definition of a pattern suggests that pattern discovery involves the search for significant events. In this light, the geometry of an event naturally demands further clarification. Theoretically, the event can be any Borel set. To simplify computations, I have chosen the most elementary configuration, a rectangular event, which only requires $2d$ parameters for complete specification. To specify a hyperellipsoid for instance, one would need in the order of d^2 parameters. At the onset, this may seem to be a colossal compromise in the accuracy of the discovery process. However, it will be shown that since discovery is performed *sequentially* and *recursively*, the loss in accuracy is effectively mitigated. Furthermore, from a strictly theoretical viewpoint, representational power is not jeopardized since any geometrical configuration can be represented by a countable number of hyper-rectangles to arbitrary accuracy.

4.2.2 Advantage of the identity mapping

In the discussion of random variables in Section 4.1.2, I hinted that the identity mapping is advantageous for event-based discovery. To appreciate this advantage, first consider a general random variable mapping, $X(\cdot)$. Suppose that we are given a data set with sample space Ω. Upon applying the random variable mapping to events in Ω, we have an image space $\Gamma = \Re^d$ (Figure 4.4). Discovery is applied in the image space Γ and a subspace $B \subset \Gamma$, is discovered as significant, i.e., $g(B) > \theta_c^\alpha$. This subspace B induces an event E in Ω,

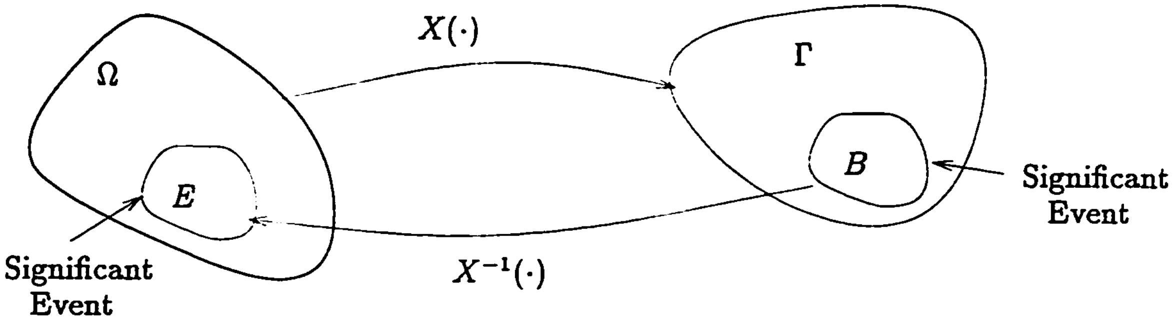


Figure 4.4: Identity mapping

$$E = \{\omega | X(\omega) \in B\} \subset \Omega \tag{4.6}$$

In general, after discovery is performed in the image space Γ, we cannot directly make any conclusions about the corresponding events in Ω. In fact, we would need to apply the inverse transformation to points in B to obtain outcomes contained in E. Fortunately, with the identity mapping, we have that $E = \{X^{-1}(b) | b \in B\} = B$ trivially. Discovery can thus be performed directly in the sample space while random variables, typically reserved for use in the image space, can be employed without restriction. Alternately, we can say that under these circumstances, discovery can be performed in either the image space or the sample space. In short, we see that continuous data naturally lends to event level discovery.

4.2.3 Pattern discovery

Theoretically, candidate events may lie anywhere in the sample space. However, in practice, we restrict the search to a compact subspace of the sample space, as demarcated by the available samples. The following definition summarizes the above discussions.

Definition 8 *Pattern discovery (Continuous data)*
Suppose we have a continuous data set with sample space Ω. Suppose further that the identity mapping is invoked to produce the image space Γ. Pattern discovery is then the search for significant (rectangular) events in a compact subspace of either the sample space Ω or the image space Γ.

Here is a quick recap of the last two sections. For continuous data, an event is a rectangular Borel subset of $\Re^d$. It is characterized by its observed frequency, volume, probability and statistic value. A pattern is an event that is significant according to its statistic value. Pattern discovery is the search for significant events.

Before discussing some important discovery objectives, I want to demonstrate the generality of the proposed event framework by interpreting a few existing methodologies from an event perspective.

4.2.4 Application to existing methods

With continuous data, trees and neural networks can be viewed as inherently event discovery mechanisms.

Decision trees

Recall that decision trees partition the sample space into subregions each time a node splits. Consider the construction of a tree from a data set of continuous variables. Suppose that at a given node i, the range of a variable X undergoes a binary split, say, $X < a$ and $X > a$, where a is some scalar value. The subregion corresponding to node i is then partitioned into 2 smaller regions. The continuation of this process demarcates successively smaller regions of space. The subspace delineated by the terminal leaf nodes represents a subspace with maximally homogeneous class composition. This subspace is simply a hyper-rectangular event in the above formulation. Indeed, tree construction can be considered the discovery of events with the impurity measure as the discovery criterion.

Neural classifiers

That neural networks discover events is suggested by the recent attempts to interpret neural learning in terms of rules. Particularly. the work of Narazaki et al. [89] lends support to this theory. In developing their explanatory mechanism, "monotonic regions" of the sample space are identified. These are basically regions where the class membership is fairly uniform. Once a "monotonic region" is identified, it is projected onto each axis to form a rule with the following structure,

If x_1 is around c_1, and x_2 is around c_2, $\ldots$, and x_d is around c_d, then $x = \{x_1, \ldots, x_d\}$ belongs to class P

where $(c_1, \ldots, c_d)$ are the coordinates of the center of the monotonic region. The fuzzy concept, "around c_i" is approximated by a closed interval around c_i. Clearly, this rule simply identifies a hyper-rectangle in the sample space. It is an event where the discovery criterion is uniformity of class membership.

Neural function approximators

White [123] argues that standard backpropagation learning is equivalent to finding the set of weights, W^*, such that the network function, $f(X, W)$, is the mini-

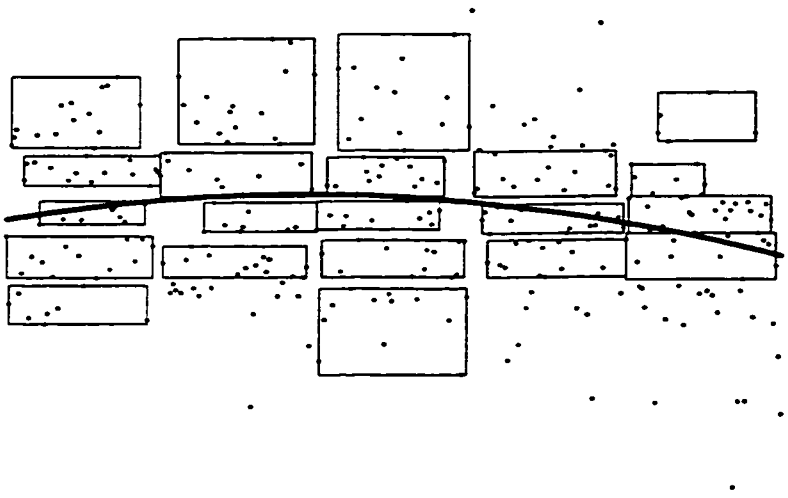


Figure 4.5: Back propagation learning as event averaging

mum mean-squared error approximation to the conditional expectation function, $g(X) = E(Y|X)$. Here. X and Y are respectively the network inputs (independent variables) and outputs (dependent variables). Other theorists have offered similar interpretations of neural network learning [81, 39. 120]. The summary statistic $E(Y|X)$ can be interpreted as the average value of Y over events which contain X. Figure 4.5 is a two-dimensional example showing, as a solid line, the values of Y learned by backpropagation. These Y values can be thought of as the average Y value of the events, shown as rectangles.

4.3 Discovery objectives

The definitions of pattern and pattern discovery are general in that they are valid for any discovery objective. The specification of this objective is important since the discovery criterion drives the discovery process and determines what type of information will be uncovered. In theory, there could be an infinite number of possible discovery criteria, reflecting the unlimited information which may be of interest. Here, I will only develop a few basic criteria. In general, a discovery hypothesis is associated with each discovery criterion. This hypothesis implies a model for the data. By measuring the amount of deviation or adherence to the model, we can gauge the degree of violation or satisfaction of the discovery objective.

In discussing each criterion, I will proceed by motivating its consideration, summarizing the basic idea in the formulation and stating the relevant mathematics. Note that the mathematical objectives presented here only serve to illustrate the different discovery incentives and are not yet suitable for pattern discovery. The discussion on test statistics in Section 4.4 will develop the proper mathematical expressions.

4.3.1 Concentration and clustering

Motivation

In many data sets, natural grouping tendencies, high frequency observations and regions of unusually high concentration are of interest. These are typical problems tackled by unsupervised learning algorithms.

Basic idea

The main premise in formulating the concentration objective is that the uniform distribution bears no information. Hence, the search for clusters and regions of high concentration becomes the search for regions where the density is significantly higher than that of the uniform density. The discovery hypothesis in this case is that the data is uniformly distributed throughout the compact subspace.

Mathematical objective

Let ρ_{obs} represent the density of points observed within the rectangular event under scrutiny. We have that.

$$\rho_{obs} = \frac{n}{v} \tag{4.7}$$

where n is the cardinality of the event and v is the volume of the rectangle. Let ρ_{unif} represent the density of points assuming a uniform distribution over the space of interest. Hence,

$$\rho_{unif} = \frac{N}{V} \tag{4.8}$$

where N is the sample size of the data set and V is the volume of the compact subspace under consideration. The discovery criterion J can then be written as,

$$J = \rho_{obs} - \rho_{unif} \tag{4.9}$$

The discovery process would attempt to find an event to maximize J.

4.3.2 Dependency

Motivation

Oftentimes, in deciphering a data set. we are interested in local interdependencies in the data. The presence of interdependencies suggests further investigation into possible interactions among variables. In addition, the detection of independence usually allows drastic simplifications of the model.

Basic idea

If the data under analysis is independent, then the joint probability density function (pdf) can be expressed as the product of the marginal densities. Let $f(\mathbf{x})$ represent the joint pdf and let $f_i(\mathbf{x})$ represent the marginal pdf for the i^{th} variable. The condition of independence is then,

$$f(\mathbf{x}) = \prod_{i=1}^{d} f_i(\mathbf{x}) \tag{4.10}$$

The discovery hypothesis for the dependency objective is that the data is independent throughout the compact subspace.

Mathematical objective

The joint pdf for a rectangular region can be estimated as [41],

$$\hat{p} = \frac{n}{Nv} \tag{4.11}$$

where n is the number of points in the region, N is the sample size and v is the volume of the region. In a similar vein, the estimated marginal pdf for the i^{th} variable is,

$$\hat{p}_i = \frac{n_i}{Nl_i} \tag{4.12}$$

Here, n_i is the number of points within the i^{th} interval l_i, obtained by projecting the region onto the i^{th} axis. The dependency objective can then be expressed as,

$$\begin{aligned} J &= \hat{p} - \prod_{i=1}^{d} \hat{p}_i && (4.13) \\ &= \frac{n}{Nv} - \prod_{i=1}^{d} \frac{n_i}{Nl_i} && (4.14) \end{aligned}$$

To search for regions of interdependencies, the discovery process would find events to maximize J.

4.3.3 Outliers

Motivation

The identification of outliers or influential observations is an important step in data analysis. In multivariate methods such as multiple regression, outliers are removed or transformed to prevent them from unduly biasing the analysis [114]. In pattern analysis, filtering of random noise and outlying observations allows the algorithms to focus on relevant underlying patterns.

Basic idea

To detect outliers, either the concentration or the dependency criterion can be used. The basic idea is that observations which are not covered by significant events are considered as outliers.

4.3.4 Linear dependence

Motivation

The presence of linear relationships among variables often simplifies analysis and modeling. Further, the presence of linearity allows the use of many well-developed statistical methods such as linear regression. Most importantly, the local analysis of pairwise linear dependencies alone. completely reveals all higher order linear dependencies within the same local subspace.

Basic idea

If the data is governed by a linear relationship, the variation in the data can be adequately captured by a line or a hyperplane. Equivalently, linearity is present when the amount of variation captured by the line or hyperplane far exceeds the variation which is not accounted for by the linear model. This premise is subject to the usual assumptions of linear regression, such as, normality of residuals, linearity and homoscedasticity. Violation of these assumptions weaken but does not invalidate the regression [114]. Thus, from a discovery standpoint where we are only checking for the presence of linearity, these assumptions are not restrictive. A suitable discovery hypothesis is that the data under examination does not have a linear relationship.

Mathematical objective

Let k be the number of parameters in the linear model. For d-dimensional data, $k = d + 1$, with one parameter for each of the d variables and 1 extra parameter for the constant term. Let N denote the total sample size under examination. The linear model is,

$$Y = \mathbf{X}\mathbf{b} \tag{4.15}$$

where $\mathbf{X}$ is a $N \times k$ matrix and $\mathbf{b}$ is a $k \times 1$ vector of parameters. The variation captured by the linear model is expressed as the regression sum-of-squares, $SS_{reg} = \sum(\hat{Y} - \overline{Y})^2$, where $\hat{Y}$ is the predicted value of Y and $\overline{Y}$ is the mean of Y. Likewise, the variation which is not accounted for by the linear model is given by the error-sum-of-squares, $SS_{err} = \sum(Y - \hat{Y})^2$. A suitable objective would be in the form,

$$J = \frac{SS_{reg}}{SS_{err}} \tag{4.16}$$

To discover regions of strong linear dependence, the discovery process would seek events which maximized J. The various criteria and their associated discovery hypotheses are summarized in Table 4.2. Note that the discovery hypotheses are simply the null hypotheses in hypothesis testing.

Table 4.2: Summary of discovery objectives and associated hypotheses

Discovery objective	Discovery hypothesis
Concentration	Uniform distribution
Outliers	Uniform distribution/Independence
Dependencies	Independence
Linear dependence	No linear dependence

4.4 Statistical test

Although the aforementioned discovery objectives successfully capture the specific goals of discovery, their present forms are not conducive to measuring significant differences. The deficiencies are as follows.

1. The value of J is sensitive to the magnitude of the numbers involved. Some sort of standardization is required.

2. Arbitrary thresholds need to be set in order to delineate significant from insignificant differences. These thresholds would also be problem dependent.

3. There is no prescription on how to adapt the thresholds to increasing dimensionality.

Clearly, we need a robust measure of significant differences. Fortunately, the discipline of statistics provides a wealth of tools for addressing these 3 deficiencies. The approach is to express the discovery objective as a test statistic with a known asymptotic distribution. To determine the degree of significance of an event, we simply compare the value of the test statistic for that event to a critical value

θ_c^α, at a chosen level of significance α. For all the test statistics considered, this significance level is determined in a similar manner. A prudent choice is detailed below.

4.4.1 Significance level

The significance level, α, represents the probability of a Type I error in hypothesis testing [90]. In the present context. it is the probability of detecting an event as significant, when it is actually insignificant. Typically, a 5% significance level indicates a reasonably substantial deviation from the null hypothesis, while a 1% significance level is used for very stringent testing. A Type II error is committed when an event is dismissed as insignificant, when in fact it contains organized data.

For $d = 2$, the value of α and the location of the critical region can be determined by a standard protocol discussed at the end of this section. The difficulty in selecting α arises when the dimensionality d increases. Consider the following example.

Example 2 *Let $\alpha = 0.05$ and $d = 2$. Suppose we randomly generate a 2-dimensional data set in a rectangle, R. Now divide each side of R into $Q = 5$ units, for a total of 25 regions. If we were to repeatedly generate such data sets and then test each region for significant differences, we would on average, err on $0.05 \times 25 \approx 1$ of these regions. Now suppose we have a 6 dimensional data set. We will now be $Q^6 = 5^6 = 15,625$ regions. Upon repeated data generation and testing, we would on average, err on $0.05 \times 15,625 \approx 782$ of the possible regions. Note that we do not intend to search the space exhaustively. Regardless of dimensionality, we wish to maintain the number of searches within the same order of magnitude. Hence, 782* *is an unacceptably large number of false positives.*

From the example. we realize that as dimensionality increases, the volume of the space increases exponentially, as does the number of possible regions for testing. However, most of the high-dimensional space is sparse [104], and many false claims would turn up regions with little or no data. To limit the number of possible discovery errors, we demand that the significance level decrease exponentially with increasing dimensionality. The simple function below possesses the desired behaviour,

$$\alpha(d) = \alpha_0 \exp(-(d-2)), \qquad d \geq 2 \tag{4.17}$$

where α_0 is the base value of the significance level, chosen for $d = 2$. This ensures that the number of discovery errors does not rise with dimensionality.

Justification for choice of α_0

The critical region of the hypothesis test is determined by a standard procedure [90]. Suppose the statistical test has a density $f(z, m)$ when the null hypothesis is true. Here z is a normal random variable and m is the parameter whose value is being tested. Under the null hypothesis H_0, $m = m_0$.

First, a value is assigned to α_0, the Type I error probability. Subsequently, one searches for a critical region in $\Re$ such that the Type II error probability β, is minimized for a given parameter value, $m_a \neq m_0$. The value m_a is the "true value" of the parameter. If β is too large, then α_0 is increased and the minimization is repeated. If β is still too large, more samples are collected.

In Section 4.4.2, the residual statistic is introduced as the chosen test statistic

Table 4.3: Operating characteristic and Power curve for $\alpha_0 = 5\%$.

m_a	Operating characteristic β	Power curve $1-\beta$
$m_0 \pm 3$	0.149	0.851
$m_0 \pm 4$	2.07×10^{-2}	0.9793
$m_0 \pm 5$	1.18×10^{-3}	0.99882
$m_0 \pm 6$	2.67×10^{-5}	0.999973

and in Appendix Section A.1, it is shown that this statistic is normally distributed under the null and alternate hypotheses. Hence, for a given level α and parameter value m_a, β is expressed as,

$$\beta(m_a) = \int_{-z_{1-\alpha/2}}^{z_{1-\alpha/2}} f(z, m_a) \mathrm{d}z \tag{4.18}$$

To validate statistical testing in contingency tables, it is recommended that in general, individual expected frequencies must exceed a value of 5 [77], i.e. $m_0 > 5$. In the present application, an even more conservative lower bound was usually employed, i.e. $m_0 > 25$, to ensure validity of the asymptotic assumptions (See Appendix Sections A.1 and A.2). Consequently, for significant differences, typically $|m_0 - m_a| > 5$. Using Equation 4.18 we arrive at Table 4.3. It is clear that with $\alpha_0 = 5\%$, the Type II error is negligible for $m_a \geq m_0 \pm 5$ and the corresponding power of the test approaches unity. Hence, Type II errors do not threaten the test and $\alpha_0 = 5\%$ is a plausible choice.

Having addressed the issue of significance level, we can now proceed with the discussion of individual test statistics. The first statistic applies to the concentration and dependency objectives, while the second statistic is appropriate for the linearity objective.

4.4.2 Residual analysis

In statistics, residual analysis provides valuable *local* information about a data set. In linear regression, the assumptions of normality, homoscedasticity and linearity can be checked by examining residual plots [114]. In exploratory data analysis, residuals aid in the uncovering of underlying structure [53]. The study of contingency tables also commissions residuals for the purpose of detecting outliers and for verifying normality [26]. This last application will be the one adopted for pattern discovery.

There are several advantages to the application of residuals in pattern discovery. The problems of unstandardized, wildly varying quantities and arbitrary thresholds are overcome by the use of the residual. Furthermore, the residual is easily interpreted in terms of the degree of satisfaction of the discovery objective. Residual expressions will be easily developed for the concentration and dependency objectives. In fact, these residuals strongly correlate to the values of the original objectives in Section 4.3, ensuring that the intent of the discovery is realized.

In this section, I will digress a little from the main theme in order to provide the theoretical background for applying residuals to pattern discovery.

Contingency table set-up

Consider the $I \times J$ contingency table shown in Table 4.4. The row and column labels can be the levels of the factors under study or alternatively, the rows can signify different populations while the columns can represent the possible categorizations. In either case, the observed frequency, n_{ij}, is the number of individuals with the i^{th} row label and j^{th} column label. The marginal totals are represented as n_{i+} and n_{+j}, for summation across row i and down column j, respectively. Since all the table entries are frequencies, this is also called a table of counts. Without loss of generality, we can assume multinomial sampling, i.e., the counts arise from a multinomial distribution [26].

Once we have constructed such a table, we usually wish to test some null hypothesis, H_0, for example, independence or homogeneity of proportions. The null hypothesis implicitly implies a model for the counts and thus determines how we will compute the expected values m_{ij}. To evaluate how well the model fits or equivalently, whether to accept or reject the null hypothesis, a global measure such as χ^2 or G^2 is invoked. In contrast, to identify *local* departures from the assumed model, the residual is employed. This type of *local* analysis is in tune with the theme of event level discovery.

The basic residual is defined as the difference between the observed frequency count, n, and the *estimated* expected value, $\hat{m}$, under the null hypothesis.

$$\hat{e} = n - \hat{m} \tag{4.19}$$

To use this as a statistic for hypothesis testing, we need to know the asymptotic

Table 4.4: Standard form of an $I \times J$ 2-dimensional table

		Factor 2 (Categories)				
		1	2	...	J	Totals
	1	n_{11}	n_{12}	...	n_{1J}	n_{1+}
Factor 1	2	n_{21}	n_{22}	...	n_{2J}	n_{2+}
(Populations)	⋮	⋮	⋮		⋮	⋮
	I	n_{I1}	n_{I2}	...	n_{IJ}	n_{I+}
Totals		n_{+1}	n_{+2}	...	n_{+J}	n_{++}

distribution of $\hat{e}$.

Large-sample results

The key property of interest is that when the assumed null hypothesis is satisfied, the residual asymptotically approaches a normal distribution. Therefore, deviations from the assumed model can be easily detected as cells with residual values that are unlikely to arise from the normal distribution.

The large-sample distribution of the residual is derived from the basic multinomial result, usually using the delta method. See for example [2, 26]. A derivation, in the spirit of Christensen [26] is provided in the Appendix Section A.1. Let $\hat{\mathbf{m}}$ be the vector of estimated expected frequencies. As the number of observations N increases to infinity, the residual converges to a normal distribution [26],

$$N^{-1/2}(\mathbf{n} - \hat{\mathbf{m}}) \xrightarrow{\mathcal{D}} \mathcal{N}(0, D(I - A)) \tag{4.20}$$

The A matrix is defined as $A = X(X'DX)^{-1}X'D$ where X is the model matrix.

The model matrix arises from the fitting of a log-linear model, $\log(\hat{\mathrm{m}}) = X\mathrm{b}$, with parameter vector b, to the table of counts. The matrix D is a diagonal matrix with the elements of $\hat{\mathrm{m}}$ along the main diagonal. It is denoted as $D(\hat{\mathrm{m}})$. Several equivalent forms of this result exist in the literature [34, 2, 57], but the form of the covariance matrix in (4.20) is most suitable for the present discussion.

The adjusted residuals [57] of individual cells can be obtained by dividing the simple residual by the square root of the large-sample variance. These variances are the main diagonal elements of $D(I - A)$. Therefore,

$$\hat{r}_i = \frac{n_i - \hat{m}_i}{\sqrt{\hat{m}_i(1 - \hat{a}_{ii})}} \tag{4.21}$$

where a_{ii} is the i^{th} diagonal element of the matrix A.

Conditions of applicability

When considering asymptotic results for multinomial sampling, the sample size, N, approaches infinity in a specific manner. A summary of various conditions mentioned by different authors is provided below.

(i) The number of cells should remain constant as the sample size, N, goes to infinity [2].

(ii) For multinomial or product-multinomial sampling, the probabilities in each cell should remain constant [26], or equivalently, the expected frequencies must grow at the same rate [2].

(iii) For product-multinomial sampling, the sample sizes of each population should remain in fixed proportions [26].

Agresti [2] further cautions that asymptotic results obtained by the delta method become dubious in the presence of small samples. highly sparse data or complex sampling designs.

Closed form estimate of the covariance

For hierarchical log-linear models which are decomposable, highly streamlined formulas have been developed for the elements of the covariance matrix, $D(I - A)$. The following theorem. due to Haberman [57], embodies the simplifications.

Theorem 1 ***Asymptotic covariance for the simple residual***

Suppose the estimated *expected frequency can be written in the closed form*[1]*,*

$$\hat{\mathbf{m}} = \frac{\mathbf{n}^{S_1} \times \mathbf{n}^{S_2} \times \ldots \times \mathbf{n}^{S_G}}{\mathbf{n}^{T_1} \times \mathbf{n}^{T_2} \times \ldots \times \mathbf{n}^{T_{G-1}}} \tag{4.22}$$

where $\mathbf{n}^{S_j}$ *and* $\mathbf{n}^{T_j}$ *are marginal sums.* [2] *The asymptotic variance of the simple residual of the* i^{th} *cell can be estimated by,*

$$\hat{c}_i = \hat{m}_i \left(1 - \hat{m}_i \sum_{j=1}^{G} \frac{1}{\mathbf{n}^{S_j}} + \hat{m}_i \sum_{j=1}^{G-1} \frac{1}{\mathbf{n}^{T_j}} \right) \tag{4.23}$$

[1] *See Haberman [57] for existential proofs of such estimates.*

[2] *The superscripts S_j and T_j actually denote elements of special classes which are not needed in the present discussion. We can simply think of these superscripts as denoting the marginal sums in the numerator and denominator of the estimated expected frequency $\hat{\mathbf{m}}$.*

The derivation of this theorem uses an elegant but intricate coordinate-free approach and can be found in Haberman [57, Chapter 4]. However, the relationship between (4.23) and the more common variance expressions based on projection matrices [34, 2, 26], as in (4.20), is elusive. To the best of my knowledge, no attempt has been made at consolidation. In the Appendix (Section A.2), I attempt to clarify these important but subtle relationships.

4.4.3 Pattern discovery as residual analysis

With the results of the previous section in mind. pattern discovery can now be posed as a residual analysis problem. I will first build a contingency table for pattern discovery and subsequently derive the associated residual equation. The residual will then be specialized to the concentration and dependency objectives.

Suppose that we would like to discover patterns in a given data set $\{\mathbf{x}_i, i = 1, \ldots, N\}$ where each $\mathbf{x}_i \in \Re^d$. Suppose further that the continuous subspace $S \subset \Re^d$ in which this data lies, has been partitioned into J events. [3] Let us label them as $E_j, j = 1 \ldots J$. We are interested in discovering which events are significant according to a discovery criterion.

Consider the $2 \times J$ contingency table shown in Table 4.5. The columns of the table are labeled with the events, E_j, $j = 1 \ldots J$. The first population represents the unknown distribution of the sample $\{\mathbf{x}_i\}$ under consideration. Across this first row, we enter the actual observed frequency, n_{1j}, $j = 1 \ldots J$ of each event E_j, $j = 1 \ldots J$. The last column in the right is the row total, n_{1+}, where the + sign

[3] Partitions of the sample space can be obtained by an optimization search (Section 4.8) or by maximum entropy partitioning (Section 5.1)

Table 4.5: 2 × J Contingency Table for Pattern Discovery

	Events				
Population	E_1	E_2	...	E_J	Totals
1 (unknown distribution)	n_{11}	n_{12}	...	n_{1J}	n_{1+}
2 (assumed distribution)	n_{21}	n_{22}	...	n_{2J}	n_{2+}
Totals	n_{+1}	n_{+2}	...	n_{+J}	n_{++}

indicates summation over j. This notation is consistent with that of standard contingency table analysis [26, 56, 43, 2].

The second row represents the assumed distribution. The entries here are the frequencies we would expect in each event, assuming one of the discovery hypotheses (Table 4.2) to be true. Note that summing across the second row yields $n_{2+} = n_{1+}$ as required by construction. The bottom row of totals is the sum of the frequencies for each event.

We now have a 2 × J table, consisting of 2 independent multinomial populations. This is a case of product-multinomial sampling. We would like to compare the 2 populations for significant local differences. In contingency table parlance, this amounts to a test for homogeneity of proportions. The corresponding null hypothesis, H_0, is given by,

$$H_0 \ : \ p_{1j} = p_{2j}, \quad j = 1 \dots J \tag{4.24}$$

where p_{ij} is the probability of event j for population i. This null hypothesis is for testing homogeneity of proportions and is not to be confused with the discovery hypotheses.

To detect local departures from this hypothesis, we require estimates of the expected values, m_{ij}, under the assumption that H_0 is true. We now draw upon a number of standard results for a 2-dimensional table with product-multinomial sampling. Since each n_{ij} has a multinomial distribution, the expected value is, $m_{ij} = n_{i+}p_{ij}$ [26, 46, 2]. Assuming the null hypothesis, H_0, to be true, the estimate of the common value of p_{ij} is,

$$\hat{p}_{ij} = \frac{n_{+j}}{n_{++}} \tag{4.25}$$

From this we obtain as the estimated expected value,

$$\hat{m}_{ij} = n_{i+}\frac{n_{+j}}{n_{++}} \tag{4.26}$$

Using the definition $n_{+j} = n_{1j} + n_{2j}$, and exploiting the symmetry relations, $n_{1+} = n_{2+}$ and $n_{++} = 2n_{1+}$, we may specialize Equation (4.26) to the present table. Doing so, we arrive at,

$$\begin{aligned} \hat{m}_j &= \frac{1}{2}n_{+j} && (4.27) \\ &= \frac{1}{2}(n_{1j} + n_{2j}) && (4.28) \end{aligned}$$

where n_{2j} is computed according to the discovery hypothesis. Equation (4.28) is the estimated expected value for the cells in the j^{th} column of the contingency table, assuming H_0 is true. Note that the i subscript has been dropped, as the expected value $\hat{m}$ is independent of the population i, by virtue of the hypothesis of homogeneity.

To detect local deviations from H_0, we invoke the adjusted residual test statistic [57]. From Equation (4.21), we can write the adjusted residual for the cell corresponding to the i^{th} population and j^{th} event,

$$\hat{r}_{ij} = \frac{n_{ij} - \hat{m}_{ij}}{\hat{c}_{ij}^{1/2}} \tag{4.29}$$

where $\hat{c}_{ij}$ is the estimated asymptotic variance of the numerator.

Since all the underlying models of 2-dimensional tables are trivially hierarchical and decomposable, we can compute the asymptotic variance using Haberman's formula (4.23).

By direct application of Equation (4.26), we arrive at,

$$\begin{aligned} \hat{c}_{ij} &= \hat{m}_{ij}\left(1 - \hat{m}_{ij}\left(\frac{1}{n_{i+}} + \frac{1}{n_{+j}}\right) + \hat{m}_{ij}\left(\frac{1}{n_{++}}\right)\right) && (4.30) \\ &= \hat{m}_{ij}\left(1 - \frac{n_{+j}}{n_{++}}\right)\left(1 - \frac{n_{i+}}{n_{++}}\right) && (4.31) \end{aligned}$$

This is the asymptotic variance applicable to our model under the assumptions of product-multinomial sampling and homogeneity of proportions.

By applying Equation (4.27) and the relations, $n_{++} = 2n_{1+}$ and $n_{1+} = n_{2+}$, Equation (4.31) is further simplified to,

$$\hat{c}_j = \frac{1}{2}\hat{m}_j\left(1 - \frac{\hat{m}_j}{n_{1+}}\right) \tag{4.32}$$

where again the i subscript has been dropped as there is no dependence on the population i.

Recall that only the first row of the contingency table (Table 4.5) describes the actual data. Hence, we only need to compute residuals for the *first* row of cells. For simplicity of notation, these residuals will be written as r_j without the population subscript. Substituting (4.28) into the numerator of the residual definition (4.29), we arrive at the adjusted residual, tailored for Table 4.5,

$$\hat{r}_j = \frac{\frac{1}{2}(n_{1j} - n_{2j})}{\hat{c}_j^{1/2}} \tag{4.33}$$

with $\hat{c}_j$ defined by Equation (4.32). The above statistic is suitable for detecting differences between the event frequencies of the unknown distribution against those of an assumed distribution.

Interpreting residuals

The residual is an asymptotically normal, 2-tailed test statistic. The interpretation in terms of events is just a special case of the definitions in Section 4.2. Let z_β be the value of the standard normal deviate, $Z \sim N(0,1)$, such that $P(Z \leq z_\beta) = \beta$. Let $\hat{r}_j$ be the estimated value of the adjusted residual.

- An event E_j is significant at a significance level α, if $\hat{r}_j \geq z_{1-\alpha/2}$.
- An event E_j is negatively significant at a significance level α, if $\hat{r}_j \leq -z_{1-\alpha/2}$.
- An event E_j is insignificant at a significance level α, if $|\hat{r}_j| < z_{1-\alpha/2}$.

4.4.4 Residual statistic: concentration objective

To specialize (4.33), we simply need to specify the estimate of n_{2j}, the frequency for the assumed distribution. For the concentration objective, the discovery hypothesis is that the data is uniformly distributed throughout the volume of the bounded subspace, $S \subset \Omega$, under consideration. Hence, within an event E, of volume v_j, the expected number of observations would be,

$$n_{2j} = \frac{v_j}{V_{TOT}} n_{1+} \tag{4.34}$$

where V_{TOT} is the volume of S and n_{1+} is the total number of observations. In accordance with our intuitive understanding of a uniform distribution over a volume, n_{2j} is proportional to the fraction of the total volume occupied by the event.

In the context of concentration-driven discovery, a significant event identifies a part of the subspace which contains data with some structure. In contrast, insignificant events demarcate sections of the subspace where the data is lacking in organization. Lastly, negatively significant events indicate a region of space where the data is notably sparse.

4.4.5 Residual statistic: dependency objective

Similar to the previous section, the residual is tailored to the dependency objective by specifying the estimate of n_{2j}, the frequency of the assumed distribution. Under the discovery hypothesis of independence, we assume that for an event E_j, its joint probability is equal to the product of the marginal probabilities. Therefore, the expected number of observations in E_j, under the assumption of independence is,

$$n_{2j} = n_{1+} \prod_{i=1}^{d} \hat{P}_i \tag{4.35}$$

The marginal probability is estimated as,

$$\hat{P}_i = \frac{n_i}{n_{1+}} \tag{4.36}$$

with n_i being the cardinality of the relaxed event R_i which is obtained by relaxing every dimension of E_j except for the i^{th} dimension. The relaxed event is more precisely defined in the final formulation of the dependency objective (Section 4.5.2).

When considering dependency-driven discovery, a significant event marks a subspace where the different dimensions of the data exhibit strong relationships or interactions. These dependencies might be general relations or functional relations. Regions populated with data but void of dependencies are detected as insignificant events. Negatively significant events also indicate a region of space that contains no dependencies. In addition, negative significance implies that the space is sparser than its surroundings, which may support some type of dependency.

4.4.6 F-statistic: linearity objective

The linearity objective (4.16) can be easily cast as a test statistic. In fact, the ratio

$$\hat{F} = \frac{SS_{reg}/k}{SS_{err}/(N-k-1)} \tag{4.37}$$

has an asymptotic F-distribution [114]. The computed value of $\hat{F}$ is compared against an F-distribution with k and $N-k-1$ degrees of freedom. Recall that the null hypothesis is the absence of a linear relationship between the dependent variable (DV) and the independent variables (IVs), i.e. all the regression coefficients and all correlations between DV and IVs are zero. Hence, if $\hat{F}$ exceeds the critical F-value, we reject the null hypothesis and conclude that there is an underlying linear relationship.

In this section, I have set forth 2 test statistics for discovery, the adjusted residual and the F-statistic. These provide objective, problem-independent measures of deviation from the discovery hypotheses. The significance level is the only threshold to be set and has an intuitive appeal. The problems of determining thresholds and oversensitive objectives are alleviated by using these standard statistics.

4.5 Discovery as optimization

Now that we have established pattern discovery as a process of searching for events which maximize a discovery criterion (test statistic), it is natural to formulate discovery as a mathematical optimization problem.

4.5.1 Parameters

Recall that discovery is restricted to rectangular events. There are several ways to parameterize a d-dimensional rectangle. Some alternatives are listed below. All require $2d$ parameters.

1. Specify all the vertices.
2. Specify the coordinates of 1 corner (reference point) and d lengths.
3. Specify the coordinates of 1 corner (reference point), the length of the longest diagonal and $d-1$ angles from the diagonal to the edges.

I have chosen the second parameterization as it offers a number of advantages in terms of incorporating parameter constraints. This will be discussed in Section 4.5.3. With this parameterization, the parameters are,

$$\tilde{\theta} = [\theta_1, \ldots, \theta_d, \theta_{d+1}, \ldots, \theta_{2d}] \tag{4.38}$$

where,

$$\theta_i = \begin{cases} i^{th} \text{ coordinate of the reference point,} & i = 1, \ldots, d \\ \text{length of the } i - d \text{ side from the reference point,} & i = d+1, \ldots, 2d \end{cases} \tag{4.39}$$

4.5.2 Objective functions

The objective functions are simply the test statistics written in terms of $\tilde{\theta}$. Each objective function is listed in turn. Note that to avoid confusion of notation for joint and marginal frequencies, I have used $\mathcal{N}_1$ and $\mathcal{N}_2$ to represent the observed and assumed frequencies, formerly denoted by n_{1j} and n_{2j}. Marginal frequencies are still written as n_i.

Concentration objective

The objective function is

$$\hat{r}_c(\tilde{\theta}) = \frac{\frac{1}{2}\left(\mathcal{N}_1(\tilde{\theta}) - \mathcal{N}_2(\tilde{\theta})\right)}{\sqrt{\hat{c}(\tilde{\theta})}} \tag{4.40}$$

where $\mathcal{N}_1(\tilde{\theta})$ is the cardinality of the event E,

$$\mathcal{N}_1(\tilde{\theta}) = |E| \tag{4.41}$$

and $\mathcal{N}_2(\tilde{\theta})$ is the assumed frequency,

$$\mathcal{N}_2(\tilde{\theta}) = \frac{v(E)N}{V_{TOT}} \tag{4.42}$$

The event E is defined by the parameters $\tilde{\theta}$ as,

$$E = \{\omega | \theta_i < \omega_i \leq \theta_i + \theta_{i+d}, \;\; i = 1, \ldots, d\} \subset \Omega \tag{4.43}$$

where $\omega = \{\omega_1, \ldots, \omega_d\}$ is a sample in Ω. The volume of the event is also expressed in terms of $\tilde{\theta}$,

$$v(E) = \prod_{i=1}^{d} \theta_{i+d} \tag{4.44}$$

Finally, the asymptotic variance of $n(\tilde{\theta}) - \hat{m}(\tilde{\theta})$ is given by (4.31),

$$c(\tilde{\theta}) = \frac{1}{2}\hat{m}(\tilde{\theta})\left(1 - \frac{\hat{m}(\tilde{\theta})}{N}\right) \tag{4.45}$$

with $\hat{m}$ expressed by

$$\hat{m} = \frac{1}{2}(\mathcal{N}_1 + \mathcal{N}_2) \tag{4.46}$$

Dependency objective

The form of the dependency objective is similar to the concentration objective. It is given by,

$$\hat{r}_D(\tilde{\theta}) = \frac{\frac{1}{2}(\mathcal{N}_1(\tilde{\theta}) - \mathcal{N}_2(\tilde{\theta}))}{\sqrt{\hat{c}(\tilde{\theta})}} \tag{4.47}$$

where again $\mathcal{N}_1(\tilde{\theta})$ is the cardinality of the event. The frequency $\mathcal{N}_2$ is now given as,

$$\mathcal{N}_2(\tilde{\theta}) = N \prod_{i=1}^{d} \hat{P}_i(\tilde{\theta}) \tag{4.48}$$

The marginal probability estimate, $\hat{P}_i(\tilde{\theta})$ is computed as

$$\hat{P}_i(\tilde{\theta}) = \frac{n_i(\tilde{\theta})}{N} \tag{4.49}$$

with $n_i(\tilde{\theta}) = |R_i|$ denoting the cardinality of the relaxed event R_i. Relaxing an event in a given dimension simply means to extend the boundaries in that dimension to the limits of the subspace under consideration. The event R_i is obtained by relaxing all but the i^{th} dimension of the event E. Particularly, R_i is given by,

$$R_i = \left\{ \omega \;\middle|\; \begin{array}{l} \theta_i < \omega_i \leq \theta_i + \theta_{i+d} \\ L_j \leq \omega_j \leq U_j, \;\; j = 1, \ldots, d \;\; j \neq i \end{array} \right\} \subset \Omega \tag{4.50}$$

where L_j and U_j are the lower and upper boundaries of the j^{th} dimension, respectively. As with the concentration objective, the asymptotic variance is expressed by (4.45).

Linearity objective

The linearity objective is the ratio,

$$\hat{F}(\bar{\theta}) = \frac{SS_{reg}(\bar{\theta})/k}{SS_{err}(\bar{\theta})/(N-k-1)} \tag{4.51}$$

where $k = d + 1$ is the number of regression coefficients and N is the size of the sample under consideration. The sums-of-squares are computed for the points in E, by selecting one variable as the DV and using the remaining variables as IVs. The event E is defined by (4.43).

4.5.3 Constraints

To complete the mathematical formulation as an optimization problem, we need to specify the constraints. Let L_i and U_i represent the lower and upper bounds of the i^{th} dimension of the bounded subspace. Three sets of constraints are identified.

1. $L_i \leq \theta_i \leq U_i$, $i = 1, \ldots, d$

   The first constraint states that the reference point for the hyper-rectangle must fall within the bounded subspace.

2. $0 \leq \theta_i \leq U_i$, $i = d+1, \ldots, 2d$

   The constraint $\theta_i \geq 0$ ensures that the lengths of the hyper-rectangle are non-negative. The constraint $\theta_i \leq U_i$ limits the maximum length. However, this implies that the event could extend beyond the boundary U_i whenever the i^{th} coordinate of reference point is greater than L_i. Fortunately, since there is no data beyond U_i, $i = 1, \ldots, d$, events which do extend beyond U_i are naturally not favoured.

3. $\prod_{i=d+1}^{2d} \theta_i \geq \delta V_{TOT} = V_{min}$

   The last constraint demands that the event volume exceed a minimum volume, expressed as a percentage of the total volume. The purpose is to avoid volumes which produce expected frequencies which are too small for valid statistical testing. It is important to note that the minimum volume ought to depend only on the total volume and not on the dimensionality of the data.

4.5.4 Statement of optimization problem

With the parameters, objective function and constraints in place, we can now formally state the optimization problem. The pattern discovery problem is to,

$$\begin{array}{lll} \text{Maximize} & J(\tilde{\theta}) & \\ \text{subject to} & 0 \leq \theta_i \leq U_i & i = d+1, \ldots, 2d \\ & L_i \leq \theta_i \leq U_i & i = 1, \ldots, d \\ & \prod_{i=d+1}^{2d} \theta_i \geq V_{min} & \end{array} \tag{4.52}$$

The objective function $J(\tilde{\theta})$ is,

$$J(\tilde{\theta}) = \begin{cases} \hat{r}_c(\tilde{\theta}) & \text{concentration} \\ \hat{r}_d(\tilde{\theta}) & \text{dependence} \\ \hat{F}(\tilde{\theta}) & \text{linearity} \end{cases} \tag{4.53}$$

All the theoretical formulations of this chapter are now complete. The final sections present a solution technique to the optimization problem.

4.6 Genetic algorithm approach

The objective functions for concentration and dependency discovery are not smooth so that gradient-based methods cannot be readily applied. Further, the objective function is ill-posed in that the cardinality, $n(\tilde{\theta})$, cannot be expressed as an analytical function of the parameters, $\tilde{\theta}$.

Genetic algorithms (GA) have proved to be useful tools in the optimization of non-smooth objective functions [52]. Although there is debate as to whether GAs can locate a global optimum, this is not of immediate concern for the present application. In discovery, we can tolerate suboptimal and crude local solutions. *Recursive discovery* will provide the needed refinement to these inexact solutions. There are a number of advantages to applying GAs to the discovery problem. First of all, no derivatives are required and the objective functions can be used in their present form. For typical problem sizes, the GA approach requires less computations than a standard simplex direct search with penalty functions[4]. Moreover, the aforementioned constraints can be directly embedded into the GA coding scheme.

Coding scheme

The GA coding scheme can directly incorporate a variety of parameter constraints. Note that the volume constraint can be roughly approximated by a set of parameter

[4] Parallel direct searches such as [37] were not investigated

constraints. Let $R_i = U_i - L_i$. We can write the volume constraint as,

$$\prod_{i=d+1}^{2d} \theta_i \geq \delta V_{TOT} = \delta \prod_{i=1}^{d} R_i \tag{4.54}$$

where θ_i are the lengths and $\prod_{i=1}^{d} R_i = V_{TOT}$. It is clear that the volume constraint is satisfied if the following set of conditions are satisfied,

$$\theta_{j+d} \geq \delta^{1/d} R_j, \quad j = 1, \ldots, d \tag{4.55}$$

This may be a conservative bound on the lengths of the hyper-rectangle and is in fact only a sufficient but not necessary condition. The approximate bound is nonetheless tolerable since extremely small events are generally not reliable for statistical testing.

For the GA coding scheme. I have elected to use binary representation. Although not as sophisticated as gray-coding, it should be sufficient for the present purposes. I will summarize the coding procedure in five steps.

1. The first step is to determine the binary variable range. Suppose we code each variable into q bits. The representable range of values is from 0 to $2^q - 1$.

2. The second step is to compute the resolution of the binary variables. Define the resolution ρ_i, of the i^{th} binary variable as,

$$\rho_i = \frac{\max(\theta_i) - \min(\theta_i)}{2^q - 1} \tag{4.56}$$

This is simply the range of the i^{th} parameter divided by the range of the binary variable. For the present parameterization, the resolution of the binary variables are thus.

$$\rho_i = \begin{cases} \frac{U_i - L_i}{2^q - 1}, & i = 1, \ldots, d \\ \frac{U_i - \delta^{1/d} R_i}{2^q - 1}, & i = d+1, \ldots, 2d \end{cases} \tag{4.57}$$

3. Next, a linear transformation scales the actual parameter values to numbers within the binary representation range. Suppose the parameter values are $\tilde{\theta} = \{\theta_1, \ldots, \theta_{2d}\}$ and let the transformed values be denoted as $T = \{t_1, \ldots, t_{2d}\}$. The transformed values are given by,

$$t_i = \begin{cases} \text{round}\left(\frac{\theta_i - L_i}{\rho_i}\right), & i = 1, \ldots, d \\ \text{round}\left(\frac{\theta_i - \delta^{1/d} R_i}{\rho_i}\right), & i = d+1, \ldots, 2d \end{cases} \tag{4.58}$$

4. The individual genes are now obtained by converting the numbers t_i into base 2 representation. Let g_i represent the binary gene, and let Decimal_to_Binary$(\cdot)$ be a function which converts a decimal number to its binary equivalent.

$$g_i = \text{Decimal\_to\_Binary}(t_i) \tag{4.59}$$

5. The final step is to assemble the chromosome C of length $2d \times q$.

$$C = [g_1 \, g_2 \, \ldots \, g_{2d}] \tag{4.60}$$

In Steps 2 and 3 above. the parameter constraints are directly incorporated into the binary code. In fact, with this coding procedure, the GA can only represent solutions which satisfy the prescribed parameter bounds.

A standard genetic optimization routine [52] is used with a resolution of $q = 10$ bits. A population size of 30 is initiated at each optimization and a maximum of 30 generations is permitted. These were found to be empirically reasonable values to obtain crude, approximate solutions. Other coding schemes and variant GA designs are subject of much research. See Beasley [12] for an overview.

4.7 Sequential discovery

Since the subspace under analysis may have more than 1 significant event, we need to repeat discovery sequentially. Discovery should stop either when we encounter a threshold number of insignificant events or when no more events can be found. The number of insignificant events is typically set at 5 or more, to allow for a handful of poor random initializations. The general procedure is as follows. Let $Number\_insignificant$ be the number of insignificant events detected.

1. Set the discovery space S to be a subset of Ω.

2. Perform discovery in S.

3. If an insignificant event is found. check if the number of insignificant events has exceeded the specified tolerance. If exceeded, then STOP. Otherwise, increment $Number\_insignificant$ and goto Step 2.

4. If a significant event E is found,

   (a) Remove the set E from further consideration, i.e. $S = S \setminus E$.

   (b) Check if the cardinality of S is sufficient to warrant further discovery, i.e. $|S| >$ minimum # of points? If no. then STOP. Otherwise, goto STEP 2.

5. If no event is found, then STOP.

Sequential discovery reflects our interest in identifying the many possible local optima which significantly satisfy the discovery objective. The idea of sequential discovery is analogous to the projection pursuit principle of removing known structure [68, 101].

4.8 Recursive discovery

Clearly even with sequential discovery. the detected events or patterns can be quite crude. In order to refine the events. we need to extend discovery in a recursive manner. The basic idea is to take each discovered significant event and re-examine the contained data in isolation. Recursion should be stopped when there is an insufficient number of points to warrant further analysis or when the remaining subspace is insignificant. The basic recursive procedure is as follows,

1. Set S to be a subspace of Ω.

2. Perform sequential discovery on S to obtain a set of events $\{E_j, j = 1, \ldots, J\}$. If no events are detected, then EXIT.

3. For the j^{th} significant event E_j, check if $|E_j| >$ minimum # of points. If affirmative, set $S = E$ and goto Step 2. Otherwise, goto Step 4.

4. Have all events been checked? If not, then $j = j + 1$ and goto Step 3. Otherwise, EXIT.

From a measure theoretic viewpoint, the set of events forms a covering of the sample space and thus recursive discovery seeks a refinement of the initial covering. The action of recursive discovery can be interpreted in both the sample space and the parameter space.

Sample space interpretation

In the sample space, recursive discovery yields events with successively smaller volumes. If we view events as sets of real numbers, then events from recursive discovery satisfy a containment property,

$$E_0 \supset E_1 \supset E_2 \supset \dots \qquad (4.61)$$

where E_i is the event detected at the i^{th} level of recursion. Figure 4.6 illustrates the sample space interpretation.

Parameter space interpretation

In the parameter space, recursive discovery can be interpreted by way of Figure 4.7. The key observation is that each local solution induces a smaller search space for the next level of recursive discovery. Let U_0 denote the bounded parameter space.

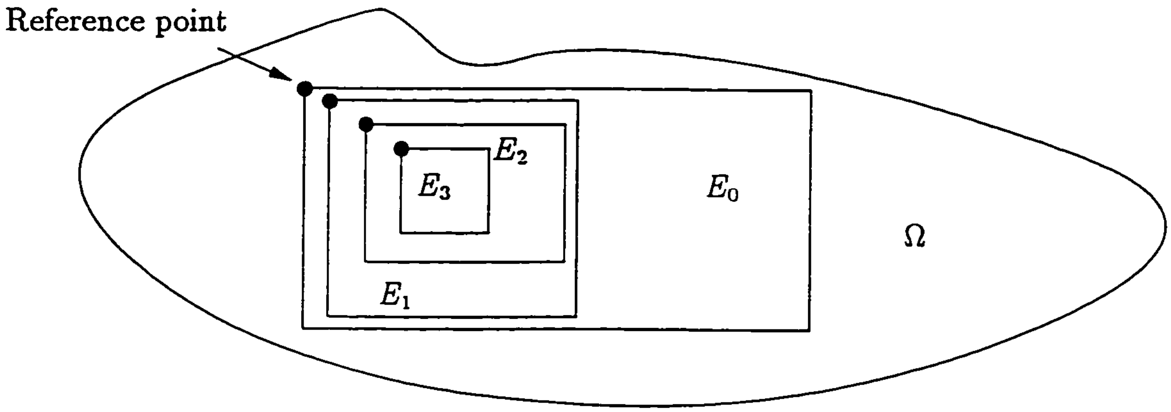


Figure 4.6: Sample space interpretation of recursive discovery

Suppose discovery yields a local solution $\tilde{\theta}_0^*$ corresponding to an event E_0. This solution induces a search space U_1 which contains all the possible solutions for the next level of discovery. When discovery is performed, a solution $\tilde{\theta}_1^*$ is obtained, corresponding to the event E_1. In like fashion, this solution induces a search space, U_1.

The induced space can be specified in general terms. If $\tilde{\theta}^*$ is the solution at the r^{th} level of recursion $r \geq 0$, then the induced search space U_r is given by,

$$U_r = \left\{ \tilde{\theta} \left| \begin{array}{ll} \theta_i^* \leq \theta_i < \theta_i^* + \theta_{i+d}^* & i = 1, \ldots, d \\ \theta_i \leq \theta_i^*, & i = d+1, \ldots, 2d \end{array} \right. \right\} \tag{4.62}$$

Note that θ_i^* is the lower boundary of event E_r in the i^{th} dimension while $\theta_i^* + \theta_{i+d}^*$ is the corresponding upper boundary. Hence, the first condition on U_r in (4.62) simply states that the reference point of the next solution must lie within the confines of E_r.

The second condition in (4.62), $\theta_i \leq \theta_i^*, \;\; i = d+1, \ldots, 2d$ stipulates that the

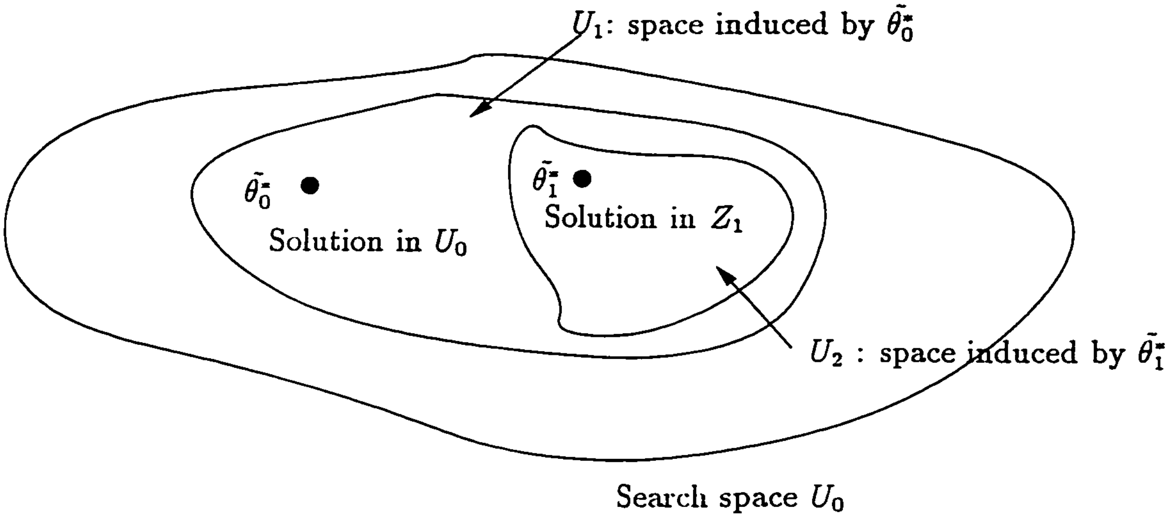


Figure 4.7: Parameter space interpretation of recursive discovery

lengths of the next solution should be shorter and at most equal to the lengths of E_r. Referring back to Figure 4.6, we appreciate that these 2 conditions must be satisfied in order for the second rectangle to lie within the first.

In summary, recursive discovery allows the individual optimizations to be crude and suboptimal since solutions can be recursively refined.

4.9 Summary

In this chapter, I have laid the theoretical groundwork for pattern discovery at the event level. Events, their characteristics, patterns and pattern discovery have been formally defined. Event-based discovery is formulated as an optimization problem with statistical objectives. The proposed solution methodology combines a genetic algorithm with sequential and recursive discovery processes. Figure 4.8 recaps the components of the framework. With these foundational concepts in place, we can now proceed to a number of important specializations and extensions of the basic framework.

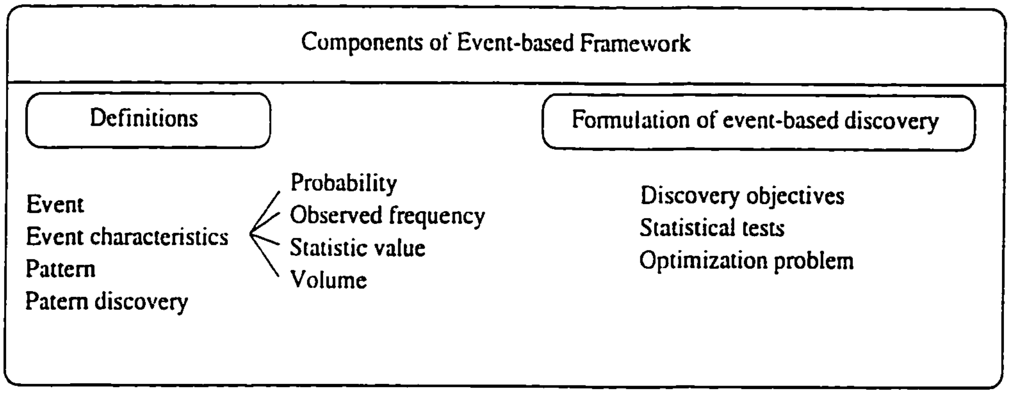


Figure 4.8: Summary of theoretical framework

Chapter 5

Pattern Discovery: Specialization, Extensions and Interpretation

In this chapter, the basic ideas developed for event level pattern discovery are specialized and extended. For data of 2 dimensions, I present a recursive, criterion-driven partitioning algorithm which offers substantial computational savings over the optimization approach, without seriously compromising the ability to discover significant events. Subsequently, I will present extensions of pattern discovery to five areas of pattern analysis: density estimation, smoothing, classification, tracking of dynamic patterns and high-dimensional discovery. These applications will serve to illustrate the versatility of the event level framework. The chapter will close with a discussion on the interpretation of high-dimensional events. Figure 5.1 gives a pictorial overview of the topics.

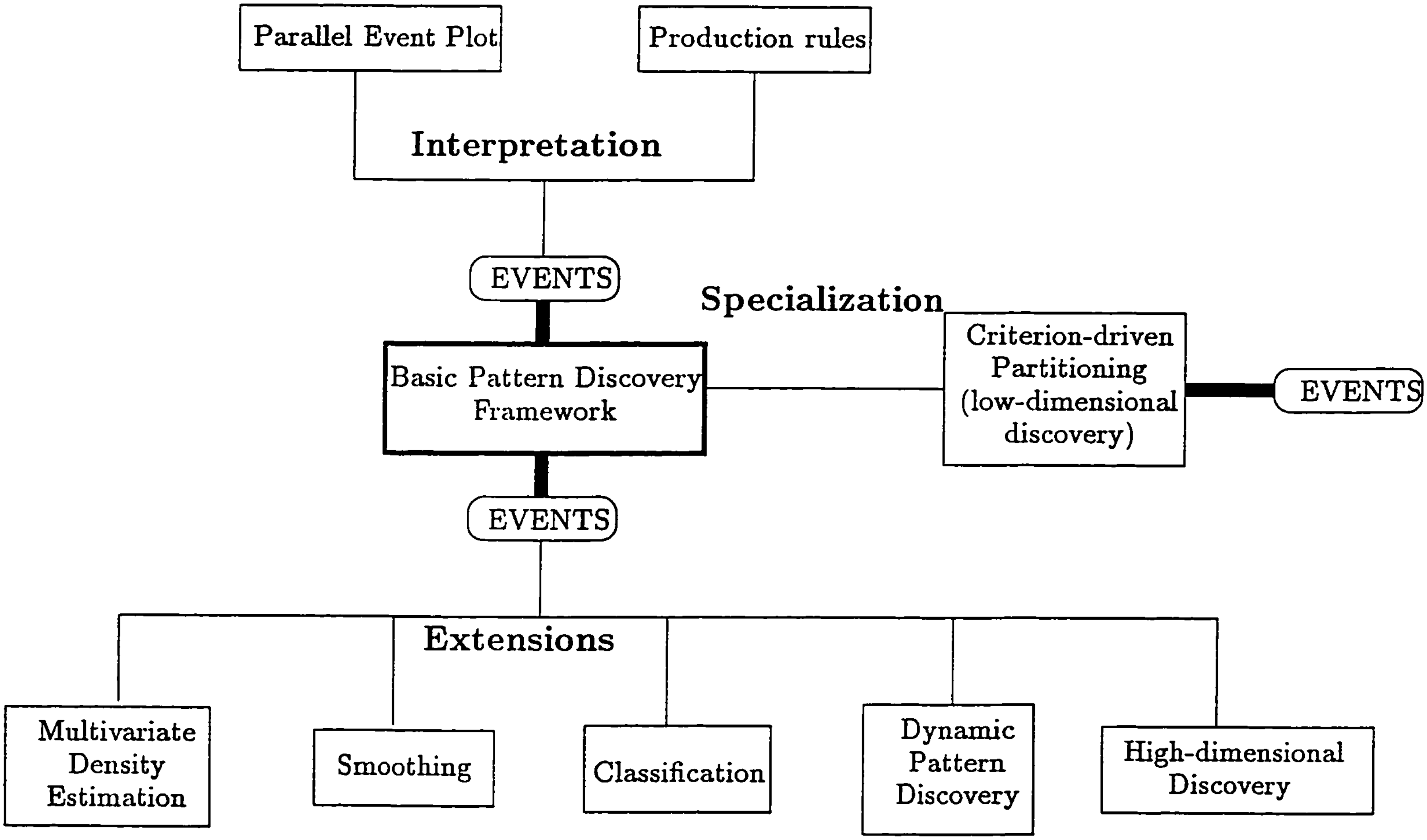

Parallel Event Plot
Production rules
Interpretation
EVENTS
Specialization
Basic Pattern Discovery Framework
Criterion-driven Partitioning (low-dimensional discovery)
EVENTS
EVENTS
Extensions
Multivariate Density Estimation
Smoothing
Classification
Dynamic Pattern Discovery
High-dimensional Discovery

Figure 5.1: Overview of pattern discovery specialization, extensions and interpretation

5.1 A general low-dimensional approximation

When dealing with low-dimensional data, events can be quickly uncovered by dividing the bounded subspace into smaller contiguous regions [126, 24]. Uninteresting regions, as determined by the discovery criterion, are discarded. Subspaces with information are further subdivided into contiguous regions. The process of repeatedly dividing and evaluating subspaces can be continued until certain termination conditions are met. Throughout the process, informative subspaces are stored as significant events. With this overview in mind, a low-dimensional discovery alternative can now be developed. I will first review an information theoretic partitioning scheme which is employed in the low-dimensional discovery alternative.

5.1.1 Marginal maximum entropy partitioning

According to the definitions of events and patterns (Section 4.1), Ω should be partitioned into countably many non-overlapping rectangular regions. Clearly, such a partition is not unique. However, since we eventually want to describe the data's organization probabilistically, we desire a technique which maximally preserves its probability distribution. One such technique is known as marginal maximum entropy partitioning (MMEP) [126, 79]. This method segments the subspace of interest while *approximately* maximizing the overall entropy [107], H, of the partition. If the subspace is partitioned into J events, then the entropy is expressed as,

$$H = -\sum_{j=1}^{J} P(E_j) \log P(E_j) \tag{5.1}$$

where $P(E_j)$ is the probability associated with event E_j. The approximate nature of the method is due to the fact that partitioning is not performed in the full dimensionality, but rather *marginally* along each dimension. Lascurain has shown that this is a reasonable approximation [79] for a number of distributions.

It is well-known that the entropy of a partition (Equation (5.1)) is maximized when all probabilities are equalized [107, 72]. Event probabilities are estimated by (4.4), which is repeated here for reference,

$$\hat{P}_j = n_j/N. \tag{5.2}$$

As usual, N is the total number of data points under consideration and n_j is the number of data points within E_j. Consequently, the equalization of probabilities in (5.1) translates into an equalization of event frequencies. Therefore, marginal entropy maximization involves the segmentation of individual axes into 1-dimensional intervals such that each interval contains an approximately equal number of samples. Lascurain's procedure [79] is paraphrased below.

Marginal Maximum Entropy Partitioning Procedure (MMEP)

Let S represent the subspace to be partitioned. The partition size will be denoted as Q. As usual, d is the dimension of the data. Partitioning will produce $Q \times Q \times \ldots \times Q = Q^d$ events. The operation $\lfloor \cdot \rfloor$ rounds down to the nearest integer.

1. Enumerate the number of sample points, N, in S. Set $i = 1$. Choose a partition size Q.

2. For axis i,

(a) Identify the minimum and maximum value of the i^{th} coordinate. Label them as a_1 and a_{Q+1}, respectively.

(b) Choose $Q-1$ points, $a_2, \ldots, a_Q$ along the axis, between a_1 and a_{Q+1}, so that there are Q intervals, each containing $\lfloor N/Q \rfloor \pm 1$ points.

(c) From each a_i, extend a $(d-1)$ dimensional plane perpendicular to the axis.

(d) If $i < d$, increment i and return to Step 2. Otherwise, goto Step 3.

3. The intersection points of the $(Q+1)^d$ planes define Q^d events in $\Re^d$.

For ease of visualization, the procedure is illustrated in Figure 5.2 with a 2-dimensional example ($d = 2$), using a partition size of $Q = 3$, for a total of $Q^d = 3^2$ events. The data consists of $N = 45$ points. The dashed lines indicate the locations of the partition points on each axis. The vertical and horizontal partitions are shown separately on the left side. They are combined to produce the picture on the right-hand side. Each interval contains $N/Q = 15$ points and the observed frequencies of each event are approximately uniform.

5.1.2 Boundary refinement

The MMEP method is fairly rigid in that the boundaries of the constructed events must lie on the partitioning planes. For discovery purposes, this is inadequate as data on curved surfaces will not be well-represented. The left panel of Figure 5.3 exemplifies this dilemma with 4 events, $\{E_1, \ldots, E_4\}$. A simple enhancement is to adjust the event boundaries to coincide with the maximum and minimum coordi-

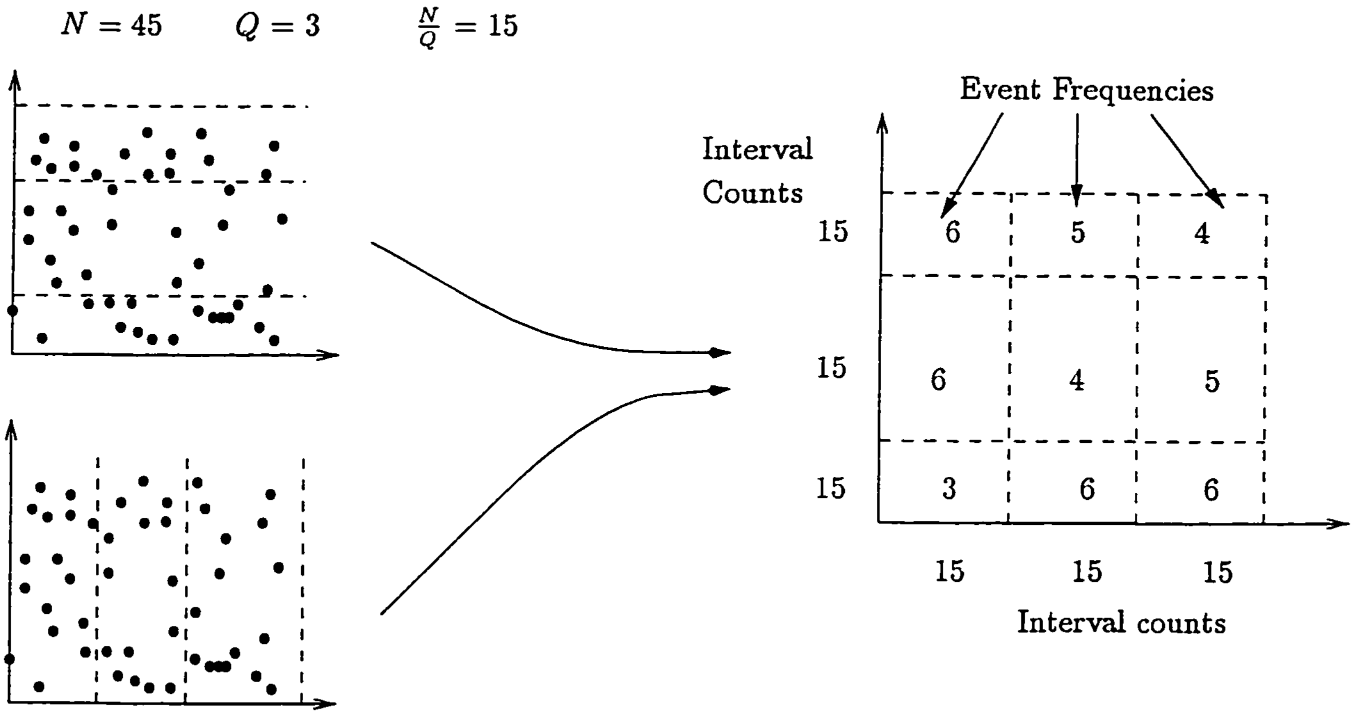


Figure 5.2: Example of MMEP with 2-dimensional data

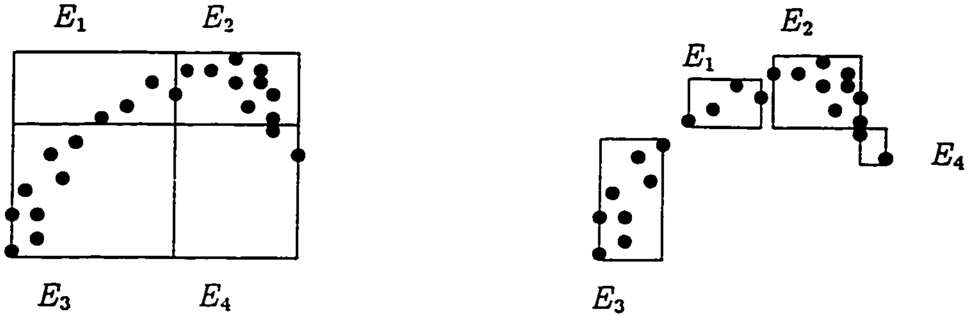


Figure 5.3: Boundary refinement

nates of the contained data. The adjacent pictorial demonstrates this adjustment. In this way, the location of events can be completely general. Close adherence to the actual data is ensured, despite the restrictive nature of the partitioning scheme.

5.1.3 Criterion-driven recursive partitioning

Despite the enhancement due to boundary refinement, it is evident that a single partition of the space yields events which only coarsely capture the data's organization. Although a very fine partition may solve the problem, it is computationally infeasible. To refine the events, we need to extend the above partitioning procedure in a recursive manner.

In the present context, recursive partitioning [126, 24, 110, 54, 47, 64, 28] means that we again apply the MMEP procedure to each constructed event. Clearly, it is not meaningful to continue this process indefinitely. Termination conditions are required (See Section 5.1.5). Moreover, recursive partitioning of every constructed event may not be necessary, especially where there is actually little data or in-

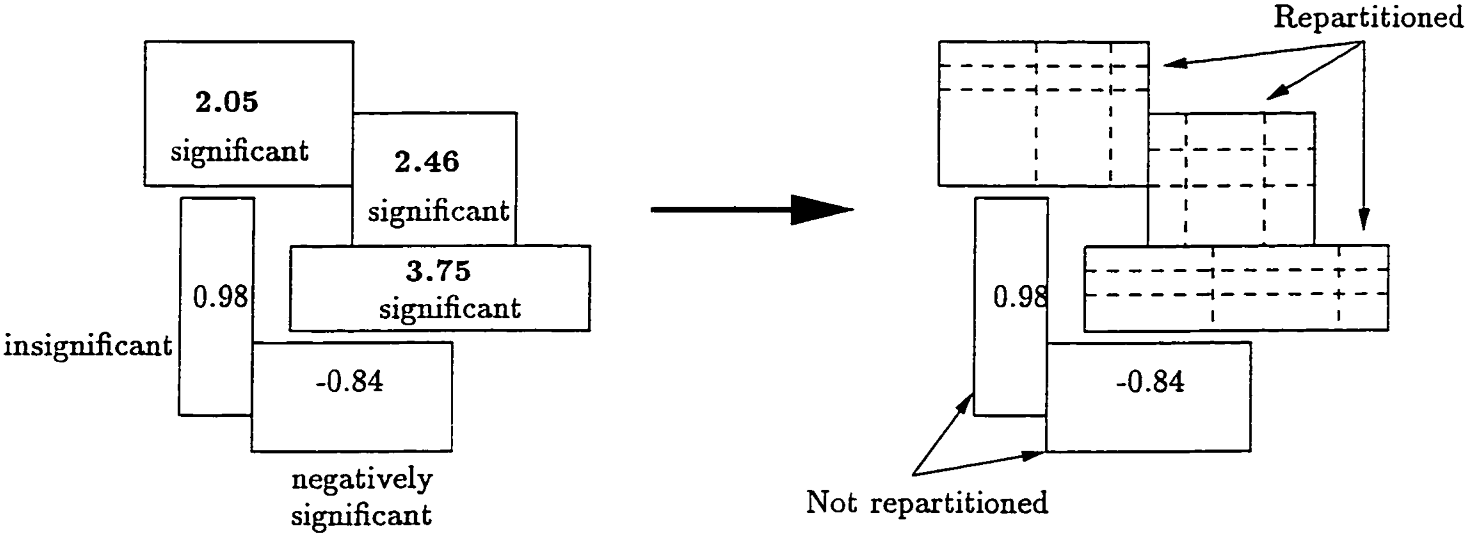


Figure 5.4: Recursive partitioning

formation. Some criterion is needed to guide the process of *selective* recursive partitioning [126, 24].

Criterion-driven recursive partitioning of the sample space has been applied to the problem of pattern classification [64, 47, 87, 55], where partitioning is typically driven by some measure of class discrimination. In like manner, discovery objectives can guide the partitioning process. Chiu et. al. [126, 24] have entertained this idea specifically with a criterion similar to the concentration objective. Here, I generalize the approach so that any of the aforementioned discovery criteria can direct the path of recursive partitioning. Specifically, only events which are significant according to the discovery criterion become candidates for further partitioning. Figure 5.4 portrays an example of two levels of partitioning. The initial partition is shown on the left, along with the objective function values of each event. On the right is the result of recursive partitioning. For this example, I have assumed that the test statistic is normally distributed, so that at a 5% significance level, only events with objective values larger than or equal to $z_{0.975} = 1.96$ are repartitioned.

An interesting interpretation exists for recursive partitioning when each segmentation is driven by an equalization of frequencies (e.g. MMEP). In these circumstances, each level of recursion can be interpreted as a refinement in the estimate of the maximum entropy of the initial partition. More detail on this novel interpretation is provided in the Appendix, Section A.3.

5.1.4 Adaptive partition size

In the original work on MMEP [79] and early recursive partitioning-based discovery schemes [126, 24], there is no mention of how to determine an appropriate partition size Q.

At each level of partitioning, it makes sense to choose a partition size which yields the most information about the data's structure. Since significant events (both positive and negative) capture the information of interest, I choose the partition size Q to maximize the ratio of significant events to all candidate events. The number of significant events alone cannot justify the choice of Q. While the actual number of significant events may be large, the tally of insignificant events may be many times larger, thereby incorrectly suggesting that the current partition is very informative.

Formally, suppose the sample space is to be partitioned into Q^d events. The number of significant events is a function of the partition size Q. Thus, let $M(Q)$ represent the number of positively and negatively significant events in this partition.

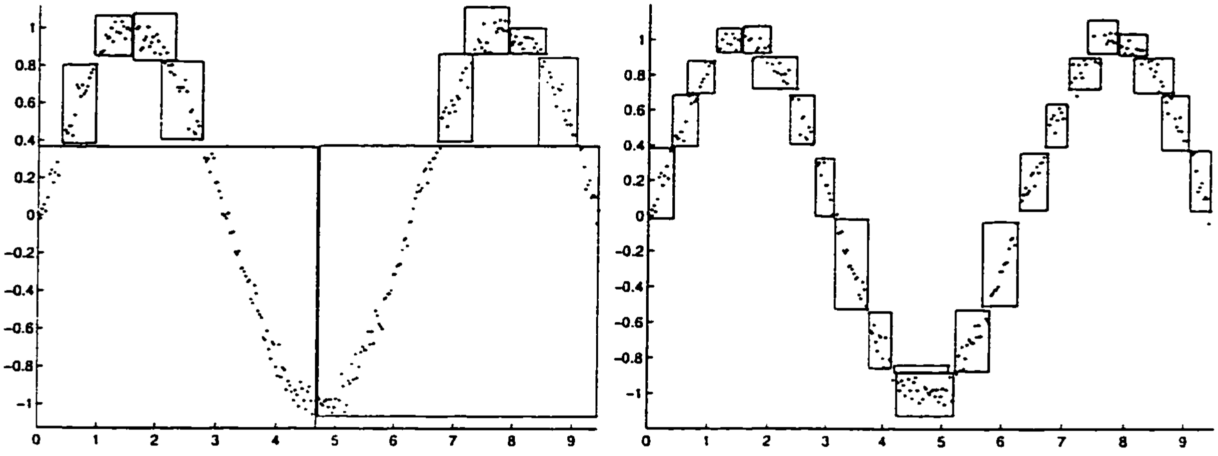


Figure 5.5: Adaptive partitioning

Choose Q to maximize.

$$\frac{M(Q)}{Q^d}. \tag{5.3}$$

As a simple example, suppose that we were trying to detect the structure of the data in Figure 5.5, using the concentration objective. The left panel is the result of fixed 2 × 2 partitioning. Much of the structure in the data is lost. The right panel exemplifies the substantial improvement in discovery due to adaptive partitioning at each level of recursion.

5.1.5 Termination conditions

As alluded to in the explanation of recursive partitioning, criteria need to be established to prevent infinite recursion. These conditions are now outlined. More than one condition may be satisfied simultaneously.

(i) $\Lambda > 20\%$

Here, Λ is the proportion of events with expected frequency below 5.

This first condition states that partitioning should cease at a given level if more than one-fifth of the events have expected frequencies less than 5. This constraint, mentioned in [2], maintains the validity of statistical testing.

(ii) $v_j < \frac{V_{TOT}}{n_{1+}}(2\zeta_{min} - n_{1j})$

This condition applies only for the concentration objective and prevents an event E_j from being partitioned when the expected event frequency $\hat{m}_j$ is below the minimum value ζ_{min}, required for reliable statistical testing. The derivation of this condition is provided in the Appendix, Section A.4.

(iii) $g(E_j) \leq -\theta_c^\alpha$

When an event is negatively significant, it should not be partitioned. In fact, subspaces with little or no data are removed from further consideration.

(iv) $|g(E_j)| < \theta_c^\alpha$

This last condition halts recursive partitioning when an event is insignificant. Since the event's frequency does not deviate from that predicted by the discovery hypothesis, there is no need for additional investigation of the subspace.

5.1.6 Event merging

Although recursive partitioning permits the detection of local organization in the data, two unacceptable complications arise. The first issue is computational. By a straightforward calculation, we see that with increasing levels of recursion, the number of events grows exponentially. Apart from being an exhaustive drain on resources, an overwhelming number of events is unmanageable for tasks such as classification or prediction. Secondly, overlapping events would arise if we were to store every event generated from multiple levels of recursion in a given subspace. From a theoretical perspective, overlapping events are prohibited.

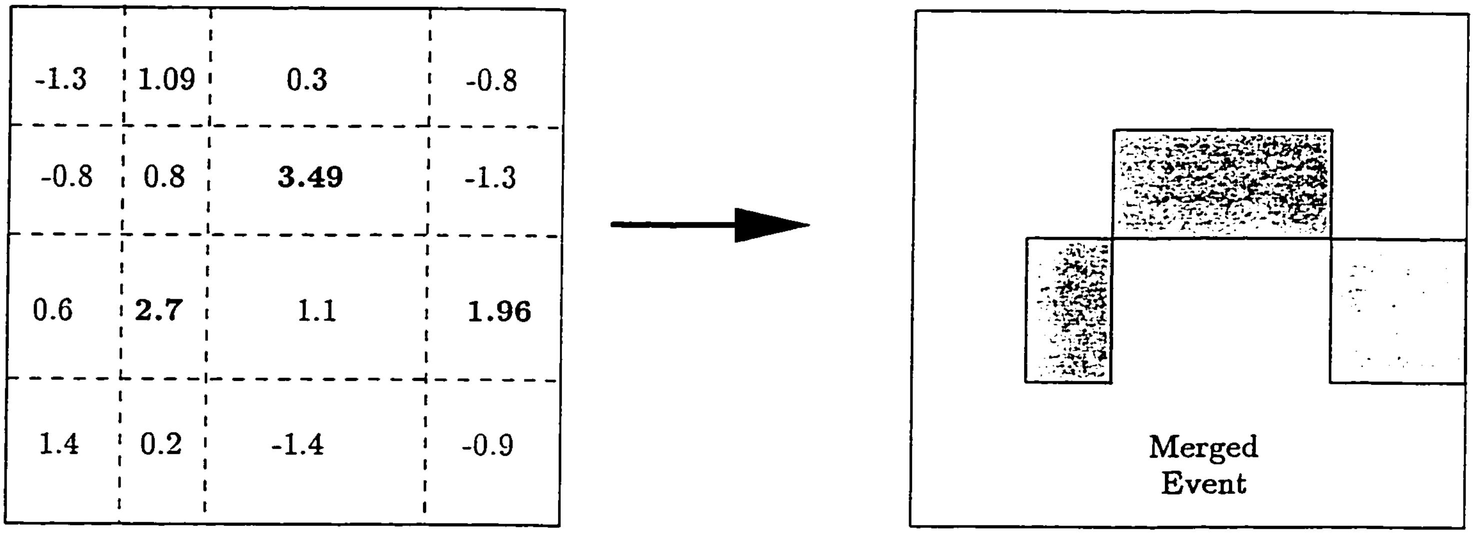
Fortunately, these problems can be effectively alleviated by *merging* insignificant events within the current subspace S under consideration. Suppose the MMEP procedure is applied to a subspace S. using a partition size Q, to produce a set of events $\{E_j\}$, $j = 1 \ldots Q^d$. An index set, κ, can be formed to identify all the insignificant events.

$$\kappa = \{j \;\mid\; |g(E_j)| < \theta_c^\alpha\} \qquad 1 \leq j \leq Q^d \tag{5.4}$$

Once identified, these insignificant events, $\{E_j\}$, $j \in \kappa$, can be merged into a compound event, $E_c = \bigcup_{j \in \kappa} E_j$. The following is a synopsis of the procedure. Assume that the space S has been partitioned by the MMEP procedure.

Event Merging Procedure

1. Identify insignificant events, $\{E_j\} \in S$, $j \in \kappa$. The index set, κ, is defined by Equation (5.4). If there are no insignificant events, then stop.



-1.3	1.09	0.3	-0.8
-0.8	0.8	**3.49**	-1.3
0.6	**2.7**	1.1	**1.96**
1.4	0.2	-1.4	-0.9

Figure 5.6: Event merging

2. Compound event construction.

   (a) Compute the compound event volume, $v_c = \sum_{j \in \kappa} v_j$

   (b) Compute the compound event frequency, $n_c = \sum_{j \in \kappa} n_j$.

3. Remove partition boundaries between insignificant events.

Figure 5.6 exemplifies the effect of event merging. The events and their objective values are shown on the left. As in the previous example, I assume a 5% significance level and a normal test statistic. Significant events are identified as those with an objective value $g(E_j) \geq z_{0.975} = 1.96$. On the right, the significant events are retained (shown as shaded rectangles) while the insignificant events ($|g(E_j)| < 1.96$) are merged.

With regards to the computational problem, event merging offers substantial relief. Suppose that we apply MMEP to S, using a partition size of Q. Without event merging, MMEP will produce $Q^d = M + |\kappa|$ events, where M represents the number of positively and negatively significant events and $|\kappa|$ is number of insignificant events. With merging, the tally of events reduces to $M + 1$. As the number of insignificant events typically far outweighs its significant counterpart, i.e., $|\kappa| \gg M$, merging amounts to an order of magnitude reduction in the total number of events stored.

From a theoretical standpoint, there will no longer be overlapping events. The compound events may assume quite arbitrary geometries but are still consistent with the event definitions. The argument is that the compound event is a union of a finite number of events and is therefore itself an event, by virtue of the closure property of the Borel field, $B(\Re^d)$.

5.1.7 Pattern discovery algorithm

We summarize the above discussions by presenting a low-dimensional pattern discovery algorithm based on criterion-driven recursive partitioning. The algorithm draws upon partitioning ideas from [64, 47, 79] and the hierarchical methodology of [126, 24].

Pattern Discovery Algorithm: Criterion-driven Recursive Partitioning

Suppose we want to seek patterns in a subspace $S \subset \Omega$.

1. Compute the partition size Q for S, according to the criterion of Equation (5.3).

2. Partition S into Q^d events $\{E_j\}$, $j = 1 \ldots Q^d$, using the MMEP approach of Section 5.1.1. Set $j = 1$.

3. For event E_j,

   (a) Refine its boundaries by the method described in Section 5.1.2.

   (b) Compute the value of the test statistic $g(E_j)$ corresponding to the current discovery objective.

4. Store, recurse or continue.

   (a) If termination condition (iv) in Section 5.1.5 is met, mark the event as insignificant. If termination condition (i) or (ii) is satisfied, store this event as significant or mark it as insignificant, depending on the value of its statistic. If condition (iii) is true, simply remove this event from further consideration. Proceed to Step 4c.

   (b) **Recursion.** If no termination conditions are met, set $S = E_j$ and go to Step 1.

   (c) Increment j and proceed to examine the next event by returning to Step 3. If all events have already been examined, i.e. $j = Q^d$, then goto Step 5.

5. Apply the Event Merging procedure of Section 5.1.6 to S. If a compound event is constructed, store the compound event. If this is a nested recursion, return to Step 4c. If this is the top-level of partitioning, stop.

In the next chapter, experiments will demonstrate the application of this low-dimensional discovery algorithm (Section 6.1.5). A number of extensions to the basic pattern discovery framework will now be presented.

5.2 Multivariate (discrete) density estimation

The estimation of the probability density function (pdf) is a central problem in multivariate data analysis, as evidenced by the large body of literature from a diversity of disciplines. See for example the works of Silverman [108],Tapia and Thompson [115] and Scott [104]. The density function gives a probabilistic description of the data's organization. Such a description is useful for data interpretation, regression, classification and prediction. Density estimation techniques can be parametric or nonparametric. Average shifted histograms [104], kernel density estimators [93], and probabilistic neural networks [86] are just a few of the many existing methodologies. Due to the valuable information captured by the pdf and its broad applicability. density estimation tools are indispensable in data analysis.

In the present context, when discovery is performed with the *concentration objective*, the resulting events can be used directly to furnish a flexible, nonparametric, discrete probability density estimate.

5.2.1 Probabilistic description

To simplify notation, the event indicator function is defined.

Definition 9 Event Indicator Function

The indicator function for the event, E_j, is defined as,

$$I_j(\mathbf{x}) = \begin{cases} 1 & if\, \mathbf{x} \in E_j \\ 0 & otherwise \end{cases} \tag{5.5}$$

Here, $\mathbf{x} \in \Re^d$ is a point in the d-dimensional sample space.

Employing Definitions 3 and 4 for event volume and observed frequency, we can assign to an event, E_j, the following probability density estimate,

$$\hat{p}_j = \frac{n_j}{N \cdot v_j} \tag{5.6}$$

where, as usual, N is the total number of sample points under consideration. This definition is along the lines of the general nonparametric density estimate of Duda et. al. [41]. Note that this probability density is just the probability estimate $\hat{P}_j$ of Equation (4.4), divided by the event volume v_j. We see immediately that the normalization condition

$$\sum_j \hat{P}_j = \sum_j \hat{p}_j \cdot v_j = 1 \tag{5.7}$$

is satisfied. To obtain a discrete probability density function, $\hat{p}$, valid for the entire sample space, recall that the events, $\{E_j\}$, do not overlap and therefore we may write compactly

$$\hat{p}(\mathbf{x}) = \sum_j I_j(\mathbf{x})\hat{p}_j \tag{5.8}$$

where again $\mathbf{x} \in \Re^d$ and $I_j(\mathbf{x})$ is the indicator function previously defined. Note that for each $\mathbf{x}$, only one term in the summation will have $I_j(\mathbf{x}) \neq 0$ as the data point can only fall into one event.

In the Appendix Section A.5, it is argued that the pdf furnished by criterion-driven recursive partitioning is asymptotically consistent.

5.2.2 Theoretical validation

In estimating probability densities, $\hat{p}$, from discovered events, there is a theoretical wrinkle that needs to be addressed. Since the density $\hat{p}$ and the probability $\hat{P}$ are related by (5.6), it is sufficient to validate the estimation of probabilities $\hat{P}$ from events.

From measure theory, probabilities should only be assigned to a Borel field of events, otherwise, the axioms of probability will be violated. Suppose we wish to estimate probabilities in a bounded subspace, $S \subset \Omega = \Re^d$. For now, assume that upon completion of discovery, we have a set of events, $\{E_j, j = 1, \ldots J\}$, which completely covers S. Clearly, this set of events does not constitute a Borel field for $\Re^d$, since Borel sets in S^C are not in $\{E_j, j = 1, \ldots, J\}$. Fortunately, this is not a problem if we simply consider $S^C \subset \Omega$ to be arbitrarily partitioned into events. Thus, the Borel field $B(\Re^d)$ will consist of discovered events in S and arbitrary events in S^C. An arbitrary partition of S^C is admissible because we only assign non-zero probabilities to events in S. See Figure 5.7.

Now consider the case when the discovered events only form a partial covering of S. The idea then is to consider the uncovered region of S as an event to which we assign a uniform background probability. To clarify, suppose discovery yields a set of events, $\{E_j, j = 1, \ldots, J\}$ in S. However, we now have,

$$\sum_{j=1}^{J} v_j < v_S \tag{5.9}$$

where v_j and v_S are the volumes of the event E_j and the subspace S, respectively.

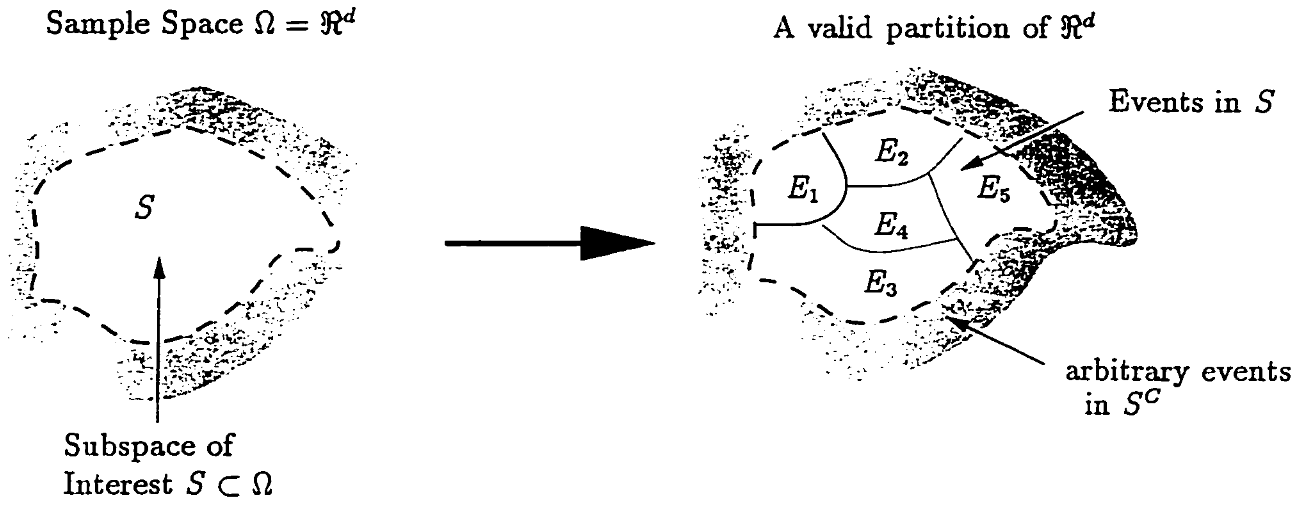

Sample Space $\Omega = \Re^d$
S
Subspace of Interest $S \subset \Omega$
A valid partition of $\Re^d$
Events in S
E_1
E_2
E_3
E_4
E_5
arbitrary events in S^C

Figure 5.7: A Valid Partition the Sample Space

Define the space not covered by the discovered events as,

$$F = S \setminus (\bigcup_{j=1}^{J} E_j) \tag{5.10}$$

Since F can be represented by at most a countable union of rectangles, it is also an event and is assigned the density value,

$$\hat{p}_F = \frac{N - \sum_{j=1}^{J} n_j}{N(v_s - \sum_{j=1}^{J} v_j)} \tag{5.11}$$

Therefore, as shown in Figure 5.8, F and $\{E_j, j = 1, \ldots, J\}$ completely cover S and the same argument as above validates the Borel field, $B(\Re^d)$. In this way, the discovered events in S always provide a valid partition for the sample space $\Re^d$, to which we can assign probabilities.

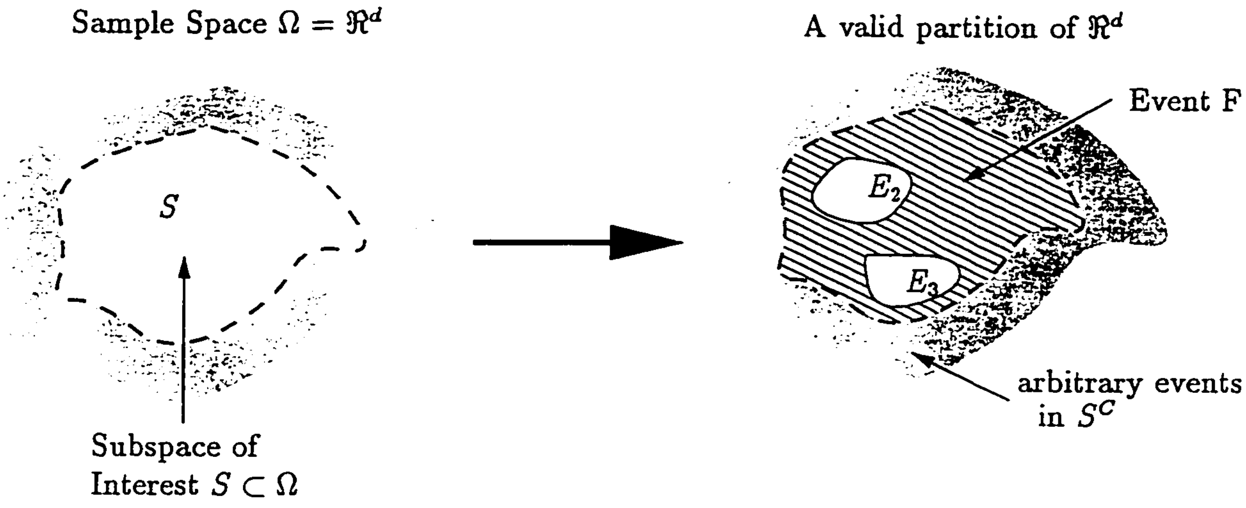

Sample Space $\Omega = \Re^d$
S
Subspace of
Interest $S \subset \Omega$
A valid partition of $\Re^d$
Event F
E_2
E_3
arbitrary events
in S^C

Figure 5.8: A valid partition when S is partially covered

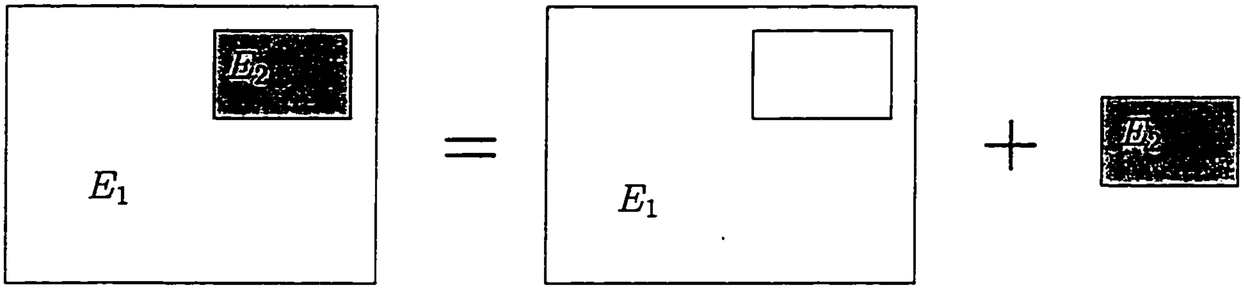


Figure 5.9: Nested events in 2-dimensions

5.2.3 Nested events

There is yet another minor theoretical issue which needs to be resolved. With recursive discovery, it is possible to have spatially nested events. Of course, events cannot overlap and therefore we consider the nested events as 2 disjoint events, as portrayed in Figure 5.9. To ensure that the estimated probabilities sum to unity, we need to subtract the frequency of the smaller, nested event from the larger event. This prevents data points from being doubly enumerated. In the example of

Figure 5.9, suppose the original frequency of E_1 is n_0 and the frequency of E_2 is n_2. After E_2 is discovered, the frequency of E_1 becomes $n_1 = n_0 - n_2$. The corresponding probability density estimates would then be, $\hat{P}_1 = \frac{n_0 - n_2}{N(v_1 - v_2)}$ and $\hat{P} = \frac{n_2}{Nv_2}$, where v_1 and v_2 are the respective volumes. The same adjustment can be extended to any number of nested events and to any number of levels of nesting.

5.2.4 Scale invariance

To explain scale invariance, we need to first define monotonic scaling, a special case of monotone transformations of the coordinate axes [3, p.22-4]. Suppose we have a data set $\mathbf{X} = \{\mathbf{x}_i\}$, $i = 1, \ldots, N$ and each $\mathbf{x}_i \in \Re^d$.

Definition 10 Monotonic Scaling

A data set $\mathbf{X} = \{\mathbf{x}_i\}$, $i = 1, \ldots, N$ is said to undergo a monotonic change of scale to $\mathbf{X}' = \{\mathbf{x}'_i\}$, $i = 1, \ldots. N$. if for the i^{th} data point,

$$\mathbf{x}'_{ik} = \gamma_k \mathbf{x}_{ik}, \quad \gamma_k > 0, \quad k = 1, \ldots, d \tag{5.12}$$

where γ_k are in general d distinct positive constants.

In other words, the k^{th} coordinate of every data point is scaled by γ_k. By scale invariance, we mean that the probability estimates are invariant to a monotonic change of scale. Equivalently, scale invariance implies that the ratio of densities at any 2 points in the sample space is invariant.

The importance of scale invariance was recognized by Devroye [38] and Friedman [47] in their work on nonparametric classification. Numerous partitioning schemes were designed specifically with this property [64, 47, 87, 126, 24]. In discovery, this is an equally indispensable property. Regardless of how the variables are scaled, a discovery system should uncover the same patterns (events) from a data set. Further, the probabilistic description of these patterns should be consistent and independent of scale.

We note that when the low-dimensional discovery algorithm is used to estimate the pdf, the discrete density function *is* scale invariant. This property is attributed to the frequency equalization tendency of the partitioning process. Let us formalize this property.

Suppose we have 2 data sets, $\mathbf{X} = \{\mathbf{x}_i\} \in S$, and $\mathbf{X}' = \{\mathbf{x}'_i\} \in S'$, $i = 1, \ldots, N$. Let $\mathbf{X}'$ be a monotonically scaled version of $\mathbf{X}$, as defined by Equation (5.12). Suppose now we apply MMEP to S and S'. By nature of Equation (5.3), the same partition size, Q, will be selected for both data sets. The result of partitioning will be 2 sets of events $\{E_j\}$ and $\{E'_j\}$, $j = 1, \ldots, Q^d$. We have the following proposition.

Proposition 1 Scale Invariance

Let $\{E_j\}$ and $\{E'_j\}$ be constructed as above. Let the probability of an event E_j be explicitly written as $\hat{P}(E_j)$. Then,

$$\hat{P}(E_j) = \hat{P}(E'_j) \quad \forall j = 1, \ldots, Q^d \tag{5.13}$$

Furthermore, with $i \neq j$,

$$\frac{\hat{p}(E_i)}{\hat{p}(E_j)} = \frac{\hat{p}(E'_i)}{\hat{p}(E'_j)} \tag{5.14}$$

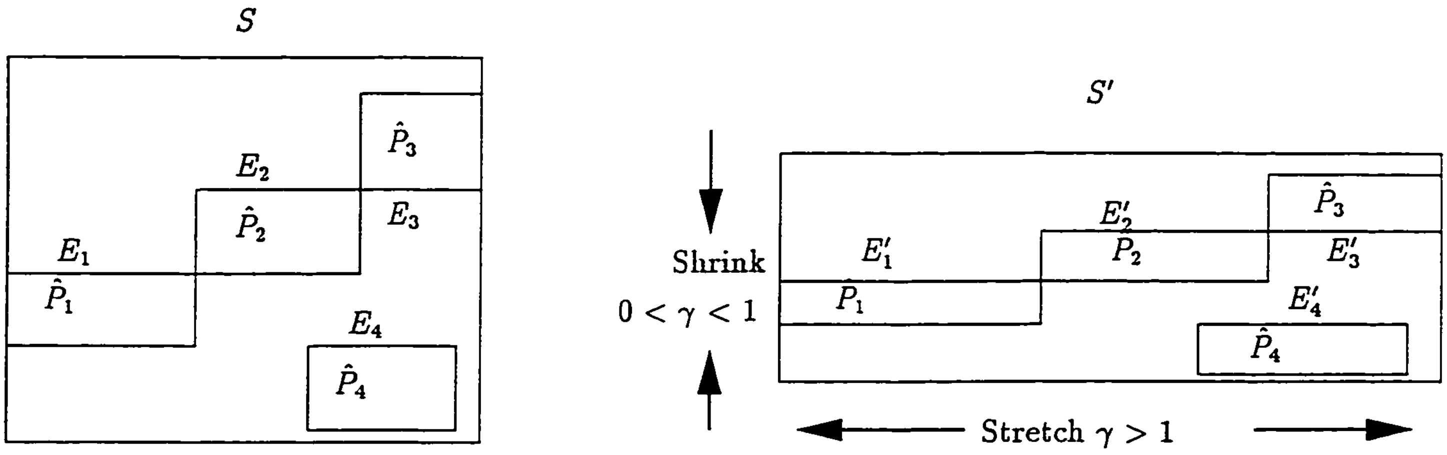


Figure 5.10: Scale invariance - Estimated probabilities $\hat{P}_j$ are unchanged

where $\hat{p}(E_j)$ is the probability density value of event E_j as given by Equation (5.6)

This property is a direct result of MMEP and the chosen density estimate. Verification is provided in the Appendix, Section A.6. A simple example is provided in Figure 5.10, where the scale of one feature is expanded and the other is compressed. The scale factor is denoted as γ. Not only is there a one-to-one correspondence between events detected in S and S', but the probabilities remain unchanged.

Due to the random nature of the GA algorithm, scale invariance cannot be guaranteed for discovery by optimization. Nonetheless, as long as the significant structure of the data is consistently captured, the pdfs can be approximately invariant. Experimental findings will empirically demonstrate the scale invariance property (Section 6.1.4).

5.3 Smoothing

The set of discovered events forms a discrete representation of the data. The representation is discrete because an observation can only fall in one of a finite number of subspaces.

However, oftentimes, a continuous description of the data's organization is preferred. For example, consider the following types of problems.

Analysis of high-dimensional data . Due to the effects of high-dimensional geometry, data in high-dimensions is inevitably sparse [104]. Consequently, a discrete representation built upon a given data set, can only account for a small fraction of the space of interest, even if the space is bounded. Only through smoothing can the vast regions of uncovered space be parsimoniously represented.

Generalization . Whenever we need to generalize from the given data, i.e. interpolate between data points or extrapolate to uncharted territory, some sort of smoothing is required. The problems of prediction, classification and function fitting are typical examples.

Exploratory analysis . When visually inspecting contours and surfaces, a smooth and continuous approximation to the data is more appealing [30].

Evidently, smoothing is an important method of data analysis [84]. It turns out that the discrete representation of events can be easily relaxed to produce a smooth representation of the data's structure.

5.3.1 Formulation

The basic idea is to estimate a kernel for each event. The cohort of these kernels then provides a smooth approximation over the sample space. The Gaussian kernel centered around the mean $\mu \in \Re^d$ with covariance matrix Σ is given by,

$$\psi(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}\Delta(\Sigma)^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu)'\Sigma^{-1}(\mathbf{x}-\mu)\right) \tag{5.15}$$

where here $\Delta(\Sigma)$ is the determinant of Σ and the prime denotes transpose. As usual, d is the dimensionality of the data. The multivariate Gaussian kernel is the most popular choice of kernel since it is continuous everywhere but by proper choice of covariance matrix, its support can be made effectively compact. Another useful property of the kernel $\psi(\mathbf{x})$ is that it is a density function, i.e.,

$$\int_{-\infty}^{\infty} \psi(\mathbf{x})d\mathbf{x} = 1 \tag{5.16}$$

To fit a kernel $\psi_E(\mathbf{x})$ to the event E, with observed frequency n_E, we simply compute the mean and covariance matrix for the data points contained in E.

$$\mu = \frac{1}{n_E}\sum_{i=1}^{n_E}\mathbf{x}_i \tag{5.17}$$

$$\Sigma = \sum_{i=1}^{n_E}(\mathbf{x}_i-\mu)(\mathbf{x}_i-\mu)' \tag{5.18}$$

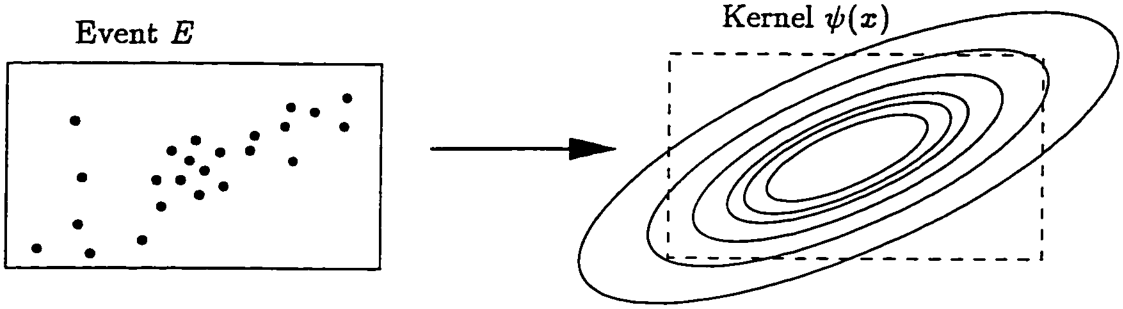


Figure 5.11: Smoothing a 2-dimensional event

Conditions of applicability

The covariance matrix is always symmetric and positive semidefinite [41]. In practice, the use of the kernel is restricted to cases when Σ is positive definite, so that the determinant is strictly positive. The kernel $\psi(\mathbf{x})$ is degenerate when the data within an event falls in a linear subspace, such that the determinant, Δ_Σ, is close to 0. This situation arises when two of the variables are strongly linearly dependent or when one variable has negligible variance.

Figure 5.11 depicts the smoothing of a 2-dimensional event using contours of the kernel density function. The next section will illuminate the application of the smoothed events.

5.3.2 Continuous density estimation

The smoothed events can be strategically combined to yield a continuous pdf estimate that satisfies probability axioms. Suppose discovery yields events $\{E_j, j = 1, \ldots, J\}$. The estimated discrete density is $\hat{p}_j$. Each event is fitted with a kernel, $\psi_j(\mathbf{x})$ as explained above.

The continuous pdf is estimated by,

$$\hat{f}(\mathbf{x}) = \sum_{j=1}^{J} W_j \psi_j(\mathbf{x}) \tag{5.19}$$

where the kernel weight. W_j, is defined as,

$$W_j = \frac{\hat{p}_j}{\sum_{j=1}^{J} \hat{p}_j} \tag{5.20}$$

Immediately, we see that the normalization condition for densities is satisfied,

$$\begin{aligned} \int_{-\infty}^{\infty} \hat{f}(\mathbf{x}) d\mathbf{x} &= \sum_{j=1}^{J} W_j \int_{-\infty}^{\infty} \psi_j(\mathbf{x}) d\mathbf{x} && (5.21) \\ &= \sum_{j=1}^{J} W_j && (5.22) \\ &= 1 && (5.23) \end{aligned}$$

Note that I have elected to use the normalized discrete densities rather than the probabilities as kernel weights. The reasoning is that the significant events may not cover all the data. so that when probabilities are estimated by (4.4), we may have $\sum_{j=1}^{J} \hat{P}_j < 1$.

5.3.3 Nested events

As with discrete probability estimates, nested events arising from recursive discovery should be handled with care. Consider the situation of Figure 5.12. Recall that we consider E_1 and E_2 to be disjoint. Let s_1 and s_2 denote the set of data points exclusively in E_1 and E_2, respectively. In other words, $s_1 \bigcap s_2 = \emptyset$.

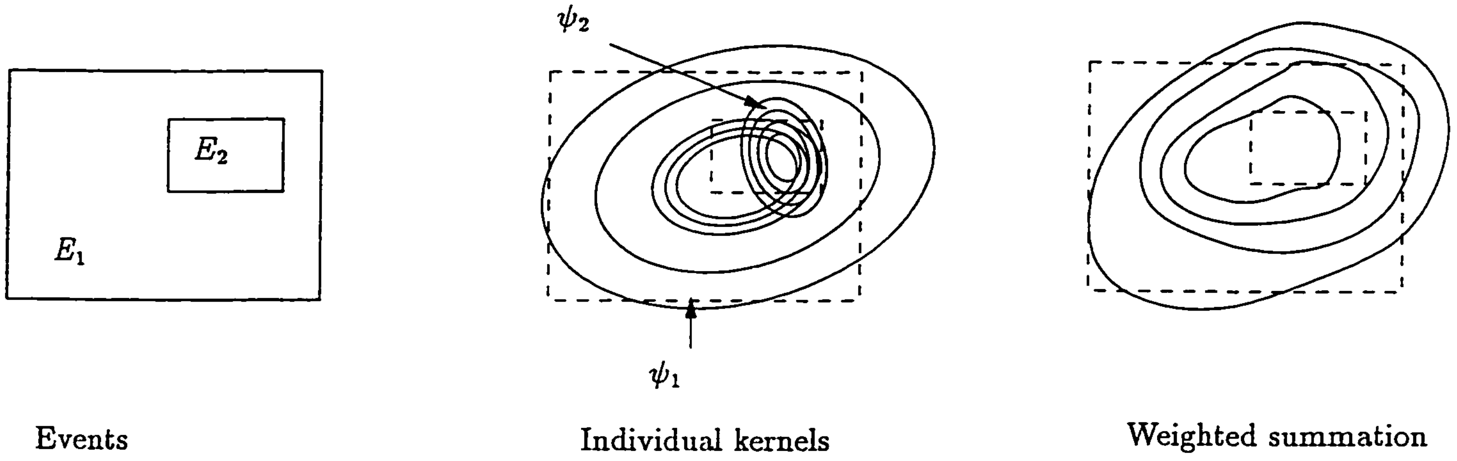


Figure 5.12: Handling nested events in continuous pdf estimation

To ensure consistency among different levels of resolution, the mean μ_1 and covariance Σ_1 for event E_1 should be estimated with the points $s_1 \bigcup s_2$. However, the mean μ_2 and covariance, Σ_2 should only be estimated with s_2. Note that the kernels ψ_1 and ψ_2 will inevitably overlap. Nonetheless, the convex sum of kernels guarantees that the probability axioms are not violated.

5.3.4 Generalized kernel estimation

Despite some attractive analytic properties, kernel estimation has traditionally been plagued with a number of practical difficulties. Historically, to obtain a smooth pdf estimate, Parzen [93] advocated the placement of a kernel density function on every data point. Theoretically, it can be shown that this procedure will produce an asymptotically consistent estimate of the underlying true density function [41, p.89].

Storage and evaluation of N kernels, where N is the size of the data set, proves to be computationally expensive. Efforts to reduce the computational burden have aimed at reducing the number of kernels. However, the questions of model order, i.e. the number of kernels to delegate to the space of interest and where to locate them, are still difficult to answer. Radial basis function research has recently generated a variety of methods to tackle problems. For determining model order, methods such as cross-validation [112], stacked generalization [124] and minimum complexity training have been applied. These are generally computationally expensive approaches, requiring many repetitions of training and retraining. The locations of basis function centers are typically determined by a different set of methods, for example. orthogonal least squares, k-means clustering or learning vector quantization [74].

It is recognized that the quality of a density estimate is primarily dependent on the choice of bandwidth rather on the choice of kernel [104]. If the bandwidth is too large, little detail is captured by the over-smoothed representation. On the other hand, too small a bandwidth will produce a jagged representation with poor generalization ability [85]. Unfortunately, choosing the bandwidth requires yet another set of tools [18, p.186]. Examples include over-smoothing rules, cross-validation rules, bootstrap methods, adaptive kernels [104] and the root-mean-square method [121]. Despite this development, comparative studies have shown that no one bandwidth selector is uniformly preferred [22]. Most methods still seek a global bandwidth applicable for every kernel. The global bandwidth has been generalized to different bandwidths for different variables, for example with probabilistic neural networks [86]. Nonetheless, this is still global in the sense that every kernel still has the same covariance matrix. The main drawback of basis function optimization approaches is that the resulting kernels cannot be completely general in shape and location.

The combination of event-based pattern discovery and kernel-based smoothing yields a generalized kernel method which simultaneously addresses the aforementioned issues.

Generalized Kernel Method

1. Perform discovery on the data in the subspace S, using the concentration objective.
2. For event j, $j = 1. \ldots, J$,
   (a) Compute the location μ_j of the kernel according to the contained data.
   (b) Compute the bandwidth Σ_j of the kernel based on the contained data.
   (c) Place a kernel $\psi_j(\mathbf{x})$ at μ_j with bandwidth Σ_j.

This is considered a general kernel method for the following reasons.

Model Order . The discovery process automatically determines the number of kernels to commission in modeling the sample space. In fact, the number of kernels is simply the number of events.

Kernel location . The location of the events directly determines the location of the kernels.

Bandwidth . The bandwidth of the kernel is determined by the subset of data contained in the event. The full covariance matrix is used so that the orientation of the principal axes is completely general and need not be orthogonal to the coordinate axes. Using the full covariance matrix also mitigates the problems arising from the curse of dimensionality [18, p.184]. Not only can each variable support its own *local* bandwidth, but every kernel can also have its own covariance matrix.

Apart from the positive definite requirement on the covariance matrix, we see that the kernel method founded upon the discovery of events is completely general. Where past methods required 2 or 3 different techniques to answer the questions of model order, kernel location and bandwidth, the present methodology exploits events for a complete solution. It is worthwhile to mention that the number of kernels employed is generally much less than N. The only drawback is that the representation of the covariance matrix is costly, requiring d^2 parameters per kernel.

Lastly, when the above kernel method is used for density estimation, it can be argued that the estimate is asymptotically consistent. This property is inherited from the asymptotic consistency of the event-based estimate (See Appendix Section A.5).

5.4 Classification

Already we have seen that discovered events can be directly employed in density estimation and smoothing. The versatility of the events is further demonstrated in their application to multicategory classification. I will first discuss a classifier with discrete events and subsequently entertain the use of smoothed events.

5.4.1 Bayesian classifier

The discovered events naturally separate the sample space into different regions. Taking advantage of this spatial demarcation, and using the estimated probability densities, a Bayesian classifier can be constructed.

Classifier construction basically involves the discovery of a set of events for each class. This task can be carried out in a supervised or unsupervised manner. Both scenarios will be outlined in a subsequent section. Once we have obtained events to represent each class, the corresponding probability densities are estimated. These densities play the role of the class conditional densities, $\hat{f}(\mathbf{x}|C_k)$, in traditional classification. The resulting classifier for a K-category problem is shown in Figure 5.13. In this figure, the label of the k^{th} class is written as C_k. The prior probabilities, $P(C_k)$, capture any *a priori* information about the class distributions. The set, $\{E\}_k$ denotes the events used to represent class k. The *a posteriori* probabilities $P(C_k|\mathbf{x})$ are computed by Bayes rule,

$$P(C_k|\mathbf{x}) = \frac{\hat{f}(\mathbf{x}|C_i)P(C_i)}{\hat{f}(\mathbf{x})} \tag{5.24}$$

where the density $\hat{f}(\mathbf{x}) = \sum_{k=1}^{K} \hat{f}(\mathbf{x}|C_k)P(C_k)$. The basic decision rule captured in Figure 5.13 is simply the Bayes decision rule,

$$\text{Classify } \mathbf{x} \text{ as belonging to class } C^* = C_i \text{ if} \tag{5.25}$$
$$P(\mathbf{x}|C_i) = \max_{k \neq i} P(\mathbf{x}|C_k)$$

Note that the unconditional density $\hat{f}(\mathbf{x})$ is not included since for a given point $\mathbf{x}$

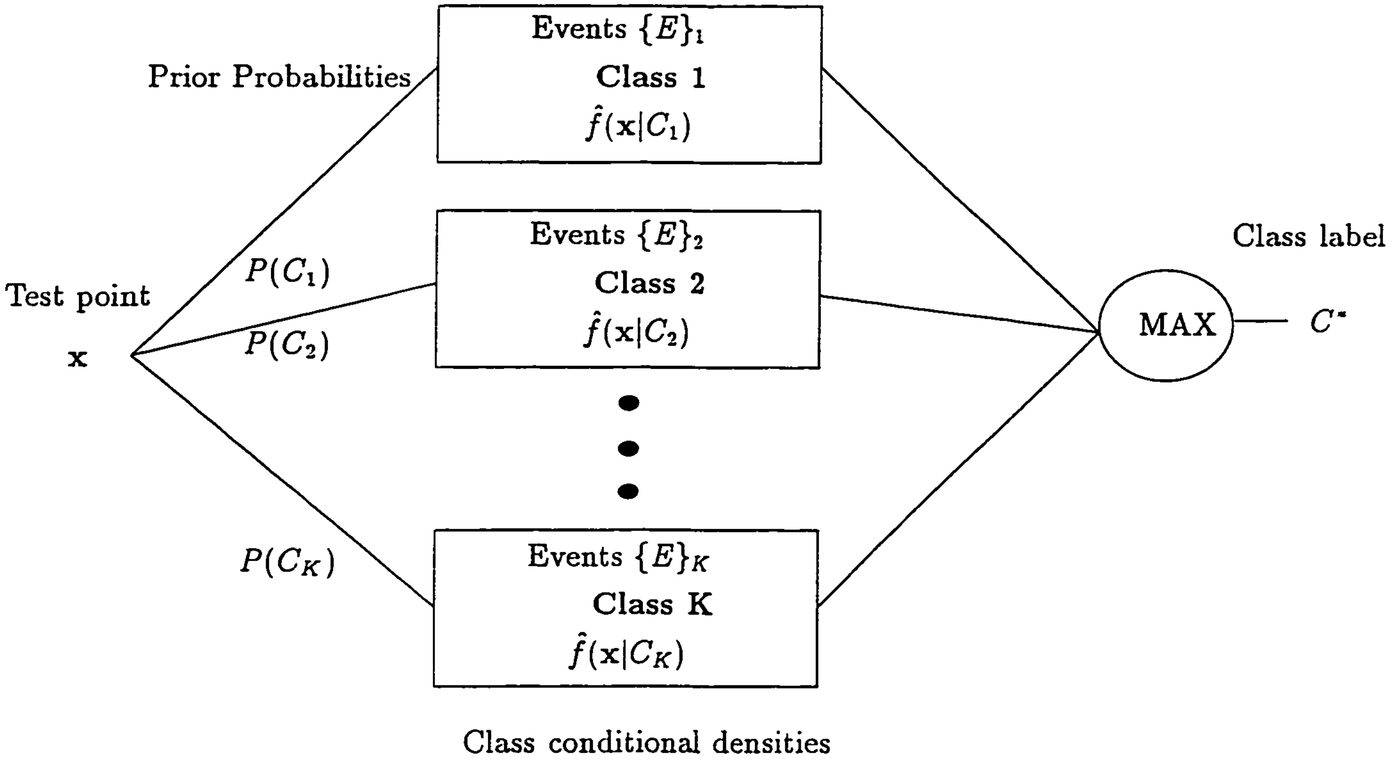


Figure 5.13: Classifier based on a set of events for each class

it is the same for each class.

Advantage of event-based classifier

The principal advantage of building a classifier directly from events is that the representation of the different classes is highly local. Consequently, class content can be disjoint. Decision boundaries can be discontinuous. Classes may be linearly inseparable and even completely nested within each other. The idea of localization also extends to the choice of features in local subspaces. This will be discussed more fully in Section 5.6. The main implication is that each class can be represented by different subsets of features.

5.4.2 Supervised approach

As alluded to in the previous section, classifier construction may proceed in 2 ways. The supervised approach involves performing discovery on each of the class-labeled subsets of data. The procedure for building a K-category classifier is outlined below.

Supervised classifier construction

1. Separate the data into K groups according to the class labels.

2. For the k^{th} group, $k = 1, \ldots, K$,

   (a) Do pattern discovery according to the discovery objective $J(\tilde{\theta})$.

   (b) Store the discovered events $\{E\}_k$.

(c) Estimate the corresponding set of pdf values $\{\hat{p}\}_k$ for the discovered events.

3. Represent class k by the set of events $\{E\}_k$ and the densities $\{\hat{p}\}_k$.

5.4.3 Unsupervised approach

When class labels are unavailable, the approach is to first use concentration discovery to detect natural groupings of the data. The discovered groups of events will then constitute the representations of the different classes.

Clustering research has developed numerous similarity measures for gauging the "likeness" of samples and countless criterion functions for evaluating a clustering partition [41, ch.6], [71, 127]. Recent innovations include the Kohonen self-organizing feature map [75], a versatile clustering algorithm founded upon nearest neighbour principles. Fuzzy c-means generalizes the traditional k-means approach by using partial cluster memberships [16]. As with pattern discovery, clustering algorithms typically rely on a criterion function to direct learning. In addition, clustering procedures generally depend heavily on a well-defined, scale-sensitive similarity measure.

With discovered events, groupings can also be identified, but without the need of a distance or similarity measure. Clusters are understood to be regions of space where the discrete pdf assumes a local maximum. To check for local pdf maxima, we can *restrict* each event's covering and compute the corresponding change in the pdf estimate. An increased pdf value is evidence that a local maximum occurs somewhere within the event. The reasoning is that if an event captures a local pdf maximum, then a multilateral restriction of the event will also contain that maximum. In fact, by arguments of asymptotic consistency [41, p.89], the reduced event volume should provide an even more accurate estimate of the maximum. Hence, if the event embodies a point of maximal density, the pdf estimate must increase. The following 2 definitions summarize the above discussion.

Definition 11 *Restriction of an event*

Let E denote a rectangular event defined by the set of vertices $\{a_i, b_i\}$, $i = 1, \ldots, d$. The restriction of E is defined by bilaterally reducing each interval (a_i, b_i) by a fraction, $0 < \gamma < 0.5$ of the original length, $L_i = b_i - a_i$. Therefore, if $E = (a_1, b_1] \times \ldots \times (a_d, b_d]$ (cf Definition 4.2), then the restriction E_r is defined as,

$$E_r = (a_1 + \gamma L_1, b_1 - \gamma L_1] \times \ldots \times (a_d + \gamma L_d, b_d - \gamma L_d] \qquad (5.26)$$

Figure 5.14 depicts a restricted event.

Definition 12 *Cluster*

An event E is identified as a cluster if its probability density estimate, $\hat{p}$, is less than the probability density estimate, $\hat{p}_r$ of its restriction, E_r. In symbols, event E is a cluster if,

$$\Delta\hat{p} = \hat{p}_r - \hat{p} > 0 \qquad (5.27)$$

Note that if E_r does not contain any points, $\hat{p}_r = 0$ and the event is not a cluster. With the ability to seek out natural groupings in the data, a classifier can now be assembled.

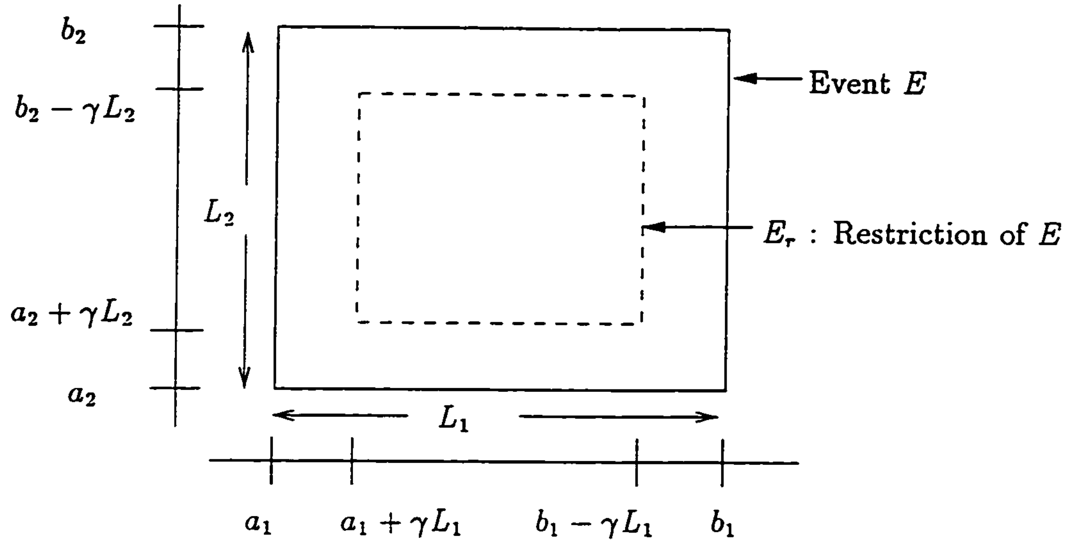


Figure 5.14: Restriction of an event

Unsupervised classifier construction

1. Perform pattern discovery on the whole unlabeled data set to obtain a set of events $\{E\}$.
2. Estimate the set of corresponding pdf values $\{\hat{p}\}$. Initialize the number of classes, $k = 1$.
3. For each event,
    (a) Restrict the event boundaries.
    (b) Compute the pdf estimate for the restricted event, $\hat{p}_r$.
    (c) Compute $\Delta\hat{p} = \hat{p} - \hat{p}_r$.
    (d) If $\Delta\hat{p} > 0$ then E identifies a cluster. Store this event as representative of class k. Increment k.

The presented approach assumes that the discovery process will identify the regions of highest density. Fortunately, the density filtering tendency of the concentration objective gives credence to this assumption.

In either the supervised or unsupervised approach to classifier construction, the same classifier set-up as in Figure 5.13 is applicable. Note that both supervised and unsupervised training is accomplished by the same, common process of pattern discovery.

5.4.4 Classifier with continuous pdfs

The class conditional density functions can be smoothed using the kernel method. The resulting classifier is identical to the one portrayed in Figure 5.13 except the pdfs are replaced with their continuous counterparts. In general, when the data is high-dimensional ($d > 5$) and the sample space is sparsely populated, continuous pdfs are preferred for classification. In contrast, discrete pdfs are favored when the classes are disjoint or when the class boundaries are discontinuous.

5.5 Time-dependent discovery

The pattern discovery framework can tackle two types of time-dependent discoveries: detection of causal relationships and tracking of pattern changes. The first only entails a mere application of already developed techniques while the second requires a novel event updating algorithm.

5.5.1 Causal relationships

Patterns in economic indicators, precipitation recordings, agricultural prices, river flows and disease outbreaks exhibit some dependence on time [65]. Of particular interest are the causal relationships between time-dependent variates. Again the event discovery framework can be immediately applied. Following the lead of Chan et al. [125], we simply consider time to be another variable in the analysis. If t denotes the time parameter, then the variables in time-dependent discovery are, $\{X_1, \ldots, X_d, t\}$. By assuming time to be an additional variable, a number of discovery queries for time-dependent data can be readily addressed.

1. *Causality*

   At the event level. causal influences are manifested as significant dependencies of certain ranges of a variable, X_i, on specific intervals of another variable, X_j, $j \neq i$. These influences can be uncovered by discovery with the dependency objective. For example, we might be interested in answering the following query. "Are there certain times where the values of X_1 cause X_2 to behave in a certain way?"

2. *Joint occurrences*

   Time slices of significant joint occurrences can be detected using the concentration objective. A typical question might be, "What values of X_1 and X_2 are likely to occur together and over what time intervals are these joint occurrences most probable?"

3. *Univariate time-dependence*

   The extent of the dependence of any isolated variable on time can be investigated via a simple bivariate dependency or linearity discovery. The variable of interest is treated as the dependent variable, while time acts as the independent variable. Pattern discovery can reveal local temporal dependencies which may only persist over specific time intervals.

It is worthwhile to mention that functional form discovery [129] with time as a independent variable, will yield a function $f(x,t)$ to describe the dynamic relationship of the variables. Alternatively, nonparametric curve fitting, such as loess fitting [31] may be utilized to describe the dynamic relationship. In either case, prediction (interpolation) or forecasting (extrapolation) can be accomplished.

5.5.2 Tracking pattern evolutions

Often, in real-world data, we encounter patterns which are not static, but evolve over the course of time. To track and describe such evolution, we need to continually adapt our representation of the data's structure. Although the pattern discovery algorithm described so far can only deal with static data, a simple extension will expand its applicability to evolving patterns.

Overview

The basic idea is illustrated in Figure 5.15. Instead of a fixed subspace, we are now concerned with discovery in a dynamic subspace, $S(t) \subset \Omega$, $\forall t$. An initial discovery is performed at time t_0, based on the observations in $S(t_0)$. The detected events

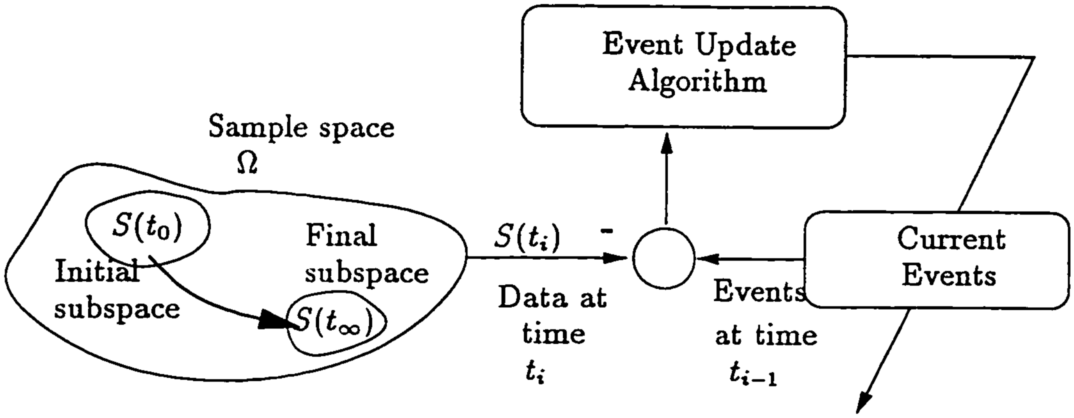


Figure 5.15: Schematic view : Tracking evolving patterns

become the "current events", i.e. the events currently retained in memory. If at a later time, significant changes to the data are detected, the current events are updated by the event update algorithm. The following are assumed about the data and observation process.

Assumptions

1. The changes in the pattern configurations are slow enough that we may analyze a block of observations collected over an interval $T \gg \tau$, where τ is the average time between observations.

2. Each set of observations are equally important.

3. The same number of observations is collected prior to each update. This assumption can be mildly violated and is only intended to prevent updates which are biased by very large or very small sample sizes.

Table 5.1: Contingency Table for Detecting Significant Event Shifts

				Background	
	Current Events			Event	
Time	E_1	...	E_J	E_{J+1}	Totals
t_1	$n_1(t_1)$	...	$n_J(t_1)$	$n_{J+1}(t_1)$	$n_+(t_1)$
t_2	$n_1(t_2)$	...	$n_J(t_2)$	$n_{J+1}(t_2)$	$n_+(t_2)$
Totals	s_1	...	s_J	s_{J+1}	n_{++}

Formulation

To detect changes in a set of events, a contingency table similar to that used for static discovery, can be constructed. In Table 5.1, $n_j(t_i)$ represents the observed frequency of event j at time t_i. Note that an extra column, "Background Event" has been added to the table. This column tabulates the number of observations which fall outside the events at a given time. The background event is defined as,

$$E_{J+1} = \Omega - \bigcup_{j=1}^{J} E_j \tag{5.28}$$

where $\{E_j\}$ are the current events. Although E_{J+1} is a valid event, its boundaries are not well-defined. Since the event volumes do not enter into the analysis, this is not a practical difficulty.

Notation for the row totals is slightly different from before, but intuitively, $n_+(t_i) = \sum_j^{J+1} n_j(t_i)$. Similarly the column totals are denoted by $s_j = \sum_i n_j(t_i)$. The null hypothesis is that the distribution of counts at t_1 and t_2 are identical. Therefore, detecting changes in the patterns (significant events) amounts to isolating local departures from this null hypothesis. As in static discovery, a residual,

$\hat{r}_j(t_1, t_2)$ can be defined to measure the change in event E_j from t_1 to t_2.

$$\hat{r}_j(t_1, t_2) = \frac{n_j(t_1) - n_j(t_2)}{\sqrt{\hat{c}_j}} \tag{5.29}$$

The denominator $\hat{c}_j$ is given by,

$$\hat{c}_j = \hat{m}_j \left(1 - \frac{s_j}{n_{++}}\right) \left(1 - \frac{n_+(t_1)}{n_{++}}\right) \tag{5.30}$$

The derivation is identical to that of the static case. Further, as before, $\hat{m}_j$ is computed as,

$$\hat{m}_j = \frac{1}{2} \left[n_j(t_1) + n_j(t_2)\right] \tag{5.31}$$

At the heart of the detection scheme is the concept of a significant event shift.

Definition 13 ***Significant Event Shift***

A significant event shift is an observable change in the data's organization which cannot be explained by random fluctuations. In symbols, an event E_j has undergone a significant event shift if,

$$\hat{r}_j(t_1, t_2) \geq z_{1-\alpha/2} \tag{5.32}$$

Recall that $z_{1-\alpha/2}$ is the critical value of the normal test statistic at a significance level α. This definition provides a problem-independent test for temporal changes in the events. To complete the update algorithm, we need to specify the appropriate responses to the detected changes.

Responses

1. $\hat{r}_j(t_1, t_2) \geq z_{1-\alpha/2}$, $j = 1, \ldots, J$

A significant event shift has occurred. An unexpectedly small number of observations fall within event E_j at the later time t_2. The support for this event is now questionable and hence E_j should be eliminated. The data points that fall within E_j at t_2 are now added to the current background event, E_{J+1}.

2. $|\hat{r}_j| < z_{1-\alpha/2}, j = 1, \dots, J$

   The number of observations falling within E_j has not changed substantially from t_1 to t_2. Simply adjust the boundaries of the event to accommodate the new data (Section 5.1.2 Boundary Refinement). The event is kept in the pool of current events.

3. $\hat{r}_j < -z_{1-\alpha/2}, j = 1, \dots, J$

   An unexpectedly large number of observations have occurred within E_j. This only further strengthens the validity of the event. Again, adjust the boundaries if necessary, and keep the event as a current event.

4. $\hat{r}_{J+1} < -z_{1-\alpha/2}$

   An unusually large number of observations have fallen into the background, i.e. outside of the current set of events. Once all the data has been collected from the eliminated events, pattern discovery should be performed only on data in the background event.

The algorithm below summarizes the procedure for detecting and tracking time-varying patterns by way of pattern discovery.

Event Update Algorithm

Suppose we are given an initial sample of N points from a subspace of $\Omega = \Re^d$.

Let $J(t)$ denote the number of current events at time t.

1. Perform discovery with the N points. Store the set of discovered events $\{E_j\}$, $j = 1, \ldots, J(t_0)$. Set $t = t_0$.

2. Let $N_C = J(t)$ be the number of current events.

3. Collect approximately N data points over the time interval $[t, t+T]$.

    (a) For each event E_j in the pool of current events, $j = 1, \ldots, J(t)$

    - i. Compute the residual $\hat{r}_j(t, t+T)$ for E_j.
    - ii. If E_j has undergone a significant event shift, i.e. $\hat{r}_j(t, t+T) \geq z_{1-\alpha/2}$, store the contained points in the background event E_{J+1}. Eliminate event E_j and reduce the number of current events $N_C = N_C - 1$.
    - iii. Otherwise, if E_j did not experience a significant event shift, simply adjust the boundaries of event E_j and keep the event as a current event.

    (b) If a significant number of points fall in the background, i.e. $\hat{r}_{J+1}(t, t+T) < -z_{1-\alpha/2}$, perform pattern discovery on the points contained within the background event. Add the k newly discovered events into the pool of current events. Increment the number of current events, $N_C = N_C + k$.

    (c) Set $t = t + T$. Let $J(t) = N_C$. Return to Step 3.

The above tracking algorithm has several advantages over existing methods in dynamic pattern classification. Firstly, the updating of events is data-driven and hence objective. The representation of the changing patterns is nonparametric and therefore general. In contrast, other flexible model-free estimators such as the multilayer perceptron are cumbersome to retrain [61]. Most importantly, in the spirit of event-based discovery, updates occur locally, only where significant changes have been detected. Not only does local updating save computation time but it facilitates the tracking of completely arbitrary changes. As well, the updating strategy may accommodate any discovery objective. Finally, unlike complex self-adaptive schemes such as Zhu's dynamic classifier [132], the static events are easily updated. The simple updating strategy underlines the flexibility of the event-based representation.

5.6 High-dimensional pattern discovery

The main difficulty with the analysis of high-dimensional data is rooted in Bellman's curse of dimensionality [13]. In statistics, this curse describes the inherent sparsity of data in multi-dimensional space. This sparsity negatively impacts on the discovery of structure in high dimensional data. For example, with methods which seek a smooth estimate of the sample space, smoothing parameters must be very large to include sufficient data. In turn, in the absence of astronomical sample sizes, this rampant smoothing compromises the ability to estimate local behaviour of the data [104]. Kernel estimators and artificial neural networks based on continuous activation functions both suffer from this predicament. In fact, for generic kernels, theoretical studies have demonstrated that to attain an equivalent estimation accuracy for increasing dimensionality, the sample size must grow at least exponentially [104]. The requisite sample sizes are seldom available in practical settings. The curse of dimensionality often renders it impossible to work in the full dimension. In fact, techniques to fully explore surfaces beyond 5 dimensions are limited [68].

Projection of high-dimensional data onto lower dimensional subspaces is the prevalent remedy to the maladies of the curse. However, several consequences of high-dimensional geometry tend to confound our interpretation of low dimensional projections. These include the migration of the content of a hypercube to its corners, the concentration of a hypersphere at its surface [122] and the near orthogonality of diagonals in hyperspace. Furthermore, two different kinds of multivariate structure may have very similar projections [104]. In other words, it is easy to misinterpret the structure of the data by examining low-dimensional projections alone.

As a final extension to the pattern discovery framework, I now discuss an event-based approach for discovery in high-dimensional settings. Before presenting the algorithm, the next 3 subsections attempt to justify the use of low-dimensional projections with each discovery objective.

5.6.1 Concentration: Justification for projections

To justify the use of low-dimensional projections for discovery with the concentration objective, we need to argue that the regions of high concentration in d dimensions are not lost by projection. Consider the joint pdf, $f(\mathbf{x}) = f(x_1, \ldots, x_d)$ for d-dimensional data. Here, X_i denotes the variable corresponding to the i^{th} dimension and x_i is its realization. Suppose that the joint density has a local maximum at $\mathbf{x}^* = \{x_1^*, \ldots, x_d^*\}$. Then we know that the gradient vanishes at $\mathbf{x}^*$, i.e.,

$$\nabla f(\mathbf{x}) \mid_{\mathbf{x}=\mathbf{x}^*} = 0 \tag{5.33}$$

implying that all partial derivatives are also zero at $\mathbf{x}^*$,

$$\frac{\partial f(\mathbf{x})}{\partial x_i} \mid_{\mathbf{x}=\mathbf{x}^*} = 0 \tag{5.34}$$

The marginal pdf corresponding to the orthogonal projection onto the X_i and X_j axes is evaluated as,

$$f_{X_i X_j}(x_i, x_j) = \int_{-\infty}^{\infty} f(\mathbf{x})\, d\{x_k\} \tag{5.35}$$

The notation $\int d\{x_k\}$ denotes integration over the set of variables $\{X_k\}$, $k \neq i, j$. We consider the bivariate marginal density since we will concentrate on 2-dimensional projections. Observe that if the gradient of the marginal density,

$$\begin{aligned} \nabla f_{X_i X_j}(x_i, x_j) &= \nabla \int_{-\infty}^{\infty} f(\mathbf{x})\, d\{x_k\} \\ &= \int_{-\infty}^{\infty} \nabla f(\mathbf{x})\, d\{x_k\} \\ &= 0 \end{aligned} \tag{5.36}$$

at $\mathbf{x} = \mathbf{x}^*$, then we are assured that the locations of peak concentration are preserved in the low-dimensional projections[1]. Local maxima in the marginal pdfs can then be used to reconstruct the corresponding local maxima in the joint pdf.

[1] Differentiation under the integral sign does not hold in general. A special case is considered in Proposition 2

Specifically, the events discovered in the low-dimensional projections can be merged to produce significant events in higher dimensions, if they exist. The difficulty in establishing the validity of (5.36) lies in the fact that we do not know $f(\mathbf{x})$. With a leap of faith, let us assume that $f(\mathbf{x})$ can be approximated by a mixture of Gaussians,

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{N} a_i \psi_i(\mathbf{x}) \tag{5.37}$$

where the familiar Gaussian kernel $\psi_i(\mathbf{x})$ is given by (5.15) and a_i is a scalar. This form of $\hat{f}(\mathbf{x})$ is analytically tractable and plausible. Indeed, we know that such kernel estimates are consistent in the mean as $N \to \infty$ [41, 104, 111]. With this assumption, we have the following proposition.

Proposition 2 *If $\hat{f}(\mathbf{x})$ is as given above and $\hat{f}_{X_i,X_j}$ is obtained by integrating $\hat{f}(\mathbf{x})$ over $\{x_k\}$, $k \neq j, i$, then for $x_i, x_j \in \Re$,*

$$\nabla_{x_i x_j} \hat{f}_{X_i X_j}(x_i, x_j) = \int_{-\infty}^{\infty} \nabla_{x_i x_j} \hat{f}(\mathbf{x})\, d\{x_k\} \tag{5.38}$$

Proof: See Appendix Section A.7.

Apostol [6] sets forth conditions under which we may *differentiate under the integral sign.* The strongest of these conditions is to show that the partial derivative of $\hat{f}(\mathbf{x})$ with respect to x_i and x_j is bounded by a non-negative, Lebesgue integrable function of $\{x_k\}$. I postpone more rigorous arguments until the Appendix.

5.6.2 Dependence: Justification for projections

To justify low-dimensional dependence discovery, again consider the joint pdf $f(\mathbf{x})$. Suppose that there exists a dependency among the variables $X_1, \ldots, X_d$. We want to verify that by projection, the dependency between any pair of variables is recoverable. Without loss of generality, consider the projection onto X_1 and X_2. Using conditional densities, we can obtain the corresponding marginal pdf f_{X_1,X_2} from $f(\mathbf{x})$ as follows.

$$\begin{aligned} f_{X_1,X_2}(x_1, x_2) &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f(x_1, \ldots, x_d)\, dx_3 \ldots dx_d && (5.39) \\ &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f(x_2, \ldots, x_d \mid x_1) f(x_1) dx_3 \ldots dx_d && (5.40) \\ &= f(x_2 \mid x_1) f(x_1) && (5.41) \end{aligned}$$

Note that since X_2 and X_1 are dependent, $f(x_2 \mid x_1)f(x_1) \neq f(x_2)f(x_1)$. Therefore, the dependency is preserved in the marginal pdf and we can have confidence in discovery in 2-dimensional projections.

5.6.3 Linearity: Justification for projections

In linearity discovery, we are interested in detecting the presence of a linear relationship between a dependent variable and a set of independent variables. A basic assumption of linear regression is that the independent variables are sufficiently uncorrelated [114]. This assumption applies to linearity discovery as well. In this light, 2-dimensional projections serve two purposes in the discovery of high-dimensional linear relationships.

Determining multicollinearities Consider the $\binom{d-1}{2}$ unique 2-dimensional projections involving only *pairs of IVs*. By performing discovery in these projections, we can detect multicollinearities or highly correlated variables. In fact, we can uncover specific ranges of the IVs where multicollinearity is particularly prevalent. Subsequently, potential higher dimensional linear relationships are investigated only between the DV and the subset of uncorrelated IVs. Hence, the 2-dimensional projections assist in the high-dimensional linearity discovery by eliminating multicollinearities.

Inferring the d-dimensional relationship If there are $d - 1$ IVs, then there will be $d - 1$ unique 2-dimensional projections involving the *DV and one IV*. Assuming that the $d - 1$ IVs are sufficiently uncorrelated, we can infer the relationship between the DV and the entire set of IVs by considering these projections individually. In other words, if we consider $Y = X_d$ to be the dependent variable, then the linear relationship,

$$Y = a_1 X_1 + \ldots + a_{d-1} X_{d-1} + C \qquad a_i \in \Re,\ C \in \Re \tag{5.42}$$

can be approximately reconstructed from the $d - 1$ estimated relationships,

$$\hat{Y}_i = \hat{a}_i X_i + \hat{C}_i \quad i = 1, \ldots, d-1 \tag{5.43}$$

where $\hat{a}_i$ and $\hat{C}_i$ are estimates of the coefficients, obtained for example by least-squares regression. The notation $\hat{Y}_i$ signifies the DV estimated in the projection with the IV X_i, where all the other variables X_j, $j \neq i$ are set to 0. Finally, because the IVs are assumed to be uncorrelated, the linear relationship in d dimensions is approximated by,

$$\hat{Y} = \sum_{i=1}^{d-1} \hat{a}_i X_i + \frac{1}{d-1} \sum_{i=1}^{d-1} \hat{C}_i \tag{5.44}$$

Thus, by examining 2-dimensional projections, higher dimensional relationships between the DV and IVs can be inferred.

With these justifications in hand, it is now fitting to present the discovery algorithm for high-dimensional data.

5.6.4 High-dimensional event synthesis

To motivate the need for statistical testing and event synthesis, recall that with the concentration and dependency objectives, discovery in low-dimensional projections alone may be misleading. Structure detected in low-dimensions may be an artifact of projection such that in higher dimensions, no structure actually exists. Secondly, the structure of the data may be occluded by projection. The synthesis and testing of higher dimensional events will provide a resolution to these issues.

For the ensuing discussions, we will need the definitions of an induced event and a dimension set.

Definition 14 *Induced Event*

Given a data set X, an event E_I induced by X, is the smallest hyper-rectangle which covers all points in X.

Definition 15 *Dimension Set*

Given a d dimensional data set, enumerate the dimensions as $\{1, 2, \ldots, d\}$. *The dimension set* D *of an event* E *in* $\Re^m$, $m < d$, *is the subset of* m *numbers corresponding to the dimensions in which* E *is defined.*

Consider events E_j and E_l characterized by different dimension sets D_j and D_l. Let X_j represent the set of points in $\Re^d$ which, when projected on the set of dimensions D_j, are contained in E_j. Define X_l in a similar fashion. To synthesize an event from E_j and E_l, we first determine the set,

$$X_I = X_j \bigcap X_l. \tag{5.45}$$

Note that X_I is a set of points in $\Re^d$. If this set is nonempty, then we can induce a rectangular event E_I with dimension set $D_I = D_j \bigcup D_l$. The event E_I contains the projection of the points X_I onto the dimensions in D_I. We will refer to the event E_I as the event synthesized from E_j and E_l. Let us consider the example of Figure 5.16 where for simplicity, $d = 3$. The 2-dimensional events E_j and E_l have dimension sets $D_j = \{1.3\}$ and $D_l = \{2, 3\}$, respectively. The points in X_j are indicated by the '+' symbols while those in X_l are labelled by open circles. Their projections onto D_j and D_l are the dots within E_j and E_l. The intersection set X_I consists of points that are common to X_j and X_l and are shown as falling inside the syntheszied event E_I. The dimension set of E_I is $D_I = \{1, 2, 3\}$. The following algorithm provides a synopsis of the synthesis procedure.

Event synthesis algorithm

As above, X_j and X_l denote subsets of points whose projections onto D_j and D_l lie respectively in E_j and E_l. Let A and B represent 2 sets of projections while N_A

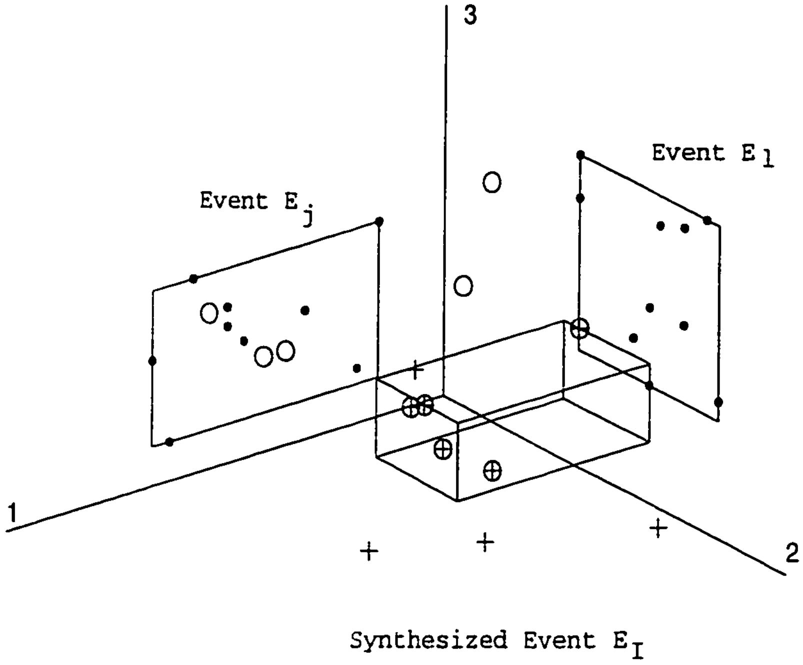


Figure 5.16: Event synthesis

and N_B are the number of unique projections in these sets. Define $NE(i)$ to be the number of events in the i^{th} projection and let E_{new} represent the newly synthesized events.

```
1. For projection $i$ in set $A$, $i = 1, \ldots, N_A$
2.    For event $j$ in projection $i$, $j = 1, \ldots, NE(i)$
3.       For projection $k$ in set $B$, $k = 1, \ldots, N_B$
4.          If projection $i$ and projection $k$ have already been examined, skip to
   the next combination of projections, i.e. return to Step 3.
5.          For event $l$ in projection $k$, $l = 1, \ldots, NE(k)$
```

6. If $X_I = X_j \bigcap X_l \neq \emptyset$, store the common points X_I in their full dimensionality in a Temporary_Bin. Return to Step 5.

7. If Temporary_Bin is not empty, use the data in Temporary_Bin to induce an event E_I with dimension set $D_I = D_j \bigcup D_l$.

8. If E_I is statistically significant according to the discovery objective, append it to the set of new events, E_{new}. Return to Step 2.

Event synthesis is the crucial operation in the high-dimensional discovery agorithm.

The high-dimensional discovery procedure begins by projecting the data orthogonally onto pairwise coordinate axes. Discovery is then performed in these projections. The initial synthesis involves only the 2-dimensional events and produces a set of new events of 3 or 4 dimensions. The next synthesis combines these new events with the original 2-dimensional events to yield events of 3,4 or 5 dimensions. This process of interfusing and testing events is repeated until no more events are found. In this way, significant events of up to d dimensions are synthesized from the collection of 2-dimensional events. The following algorithm realizes these ideas.

High-dimensional discovery algorithm

Let A and B signify 2 different sets of projections. Each set may contain projections of different dimensionalities. Let $\mathbf{E}_A$ and $\mathbf{E}_B$ represent events from these sets of projections. As usual, d denotes the dimensionality of the data.

1. Perform discovery in $\binom{d}{2}$ orthogonal 2-dimensional projections. Denote the discovered set of events as $\mathbf{E}_B$.

2. Let $\mathbf{E}_A = \mathbf{E}_B$

3. Let d_A =maximum dimension of events in $\mathbf{E}_A$.

4. While $d_A < d$ and $\mathbf{E}_A$ is not empty,

    (a) Apply the event synthesis algorithm to $\mathbf{E}_A$ and $\mathbf{E}_B$.

    (b) Store the newly synthesized events in $\mathbf{E}_{new}$.

    (c) Set $\mathbf{E}_A = \mathbf{E}_{new}$. Update d_A to be the maximum dimension of events in $\mathbf{E}_A$. Return to Step 4

I will close the section by mentioning some unique characteristics of the above method. Rather than examining isolated projections as in principal components or projection pursuit, information from different projections are cross-referenced with the hope of mitigating erroneous conclusions about high-dimensional structure. Secondly, unlike global projection techniques, the present strategy allows separate regions of the sample space (events) to be characterized by different subsets of dimensions. This flexibility permits the uncovering of general structure in data.

5.7 Interpretation of events

Insight into the structure of data can only be gained via interpretable patterns. The need for a clearer understanding of discovery results is evidenced in recent work on rule extraction from neural networks [8, 5, 106]. To close this chapter, I will present 2 different modes of interpreting the discovered events, production rules and parallel event plots.

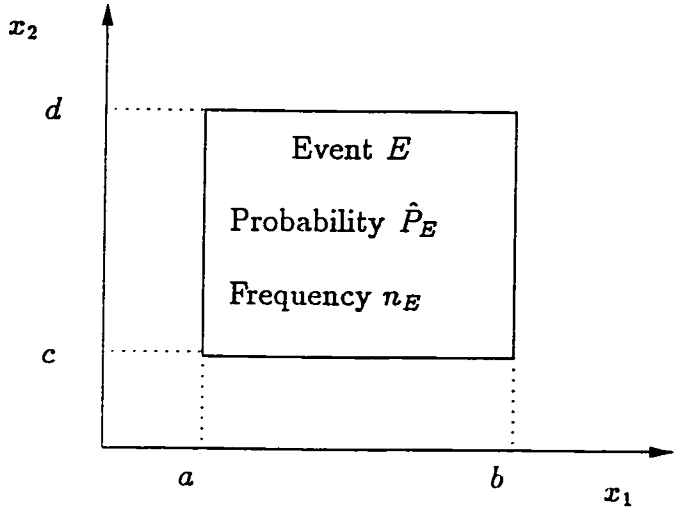


Figure 5.17: Interpreting an event as a production rule

5.7.1 Production rules

Since events delineate a region of the sample space via hyperplanes, production rules can be easily extracted. Consider the 2-dimensional event of Figure 5.17. The event can be interpreted in 2 ways, depending on the application context. Suppose the event is used to represent class k. The rule is written as,

`If` $a \leq x_1 \leq b$ `and` $c \leq x_2 \leq d$ `then the object belongs to class` k`,`
`with probability` p_E `and the support of` n_E `observations.`

In the case where there is no class reference, the discovery result is then interpreted as an association of events.

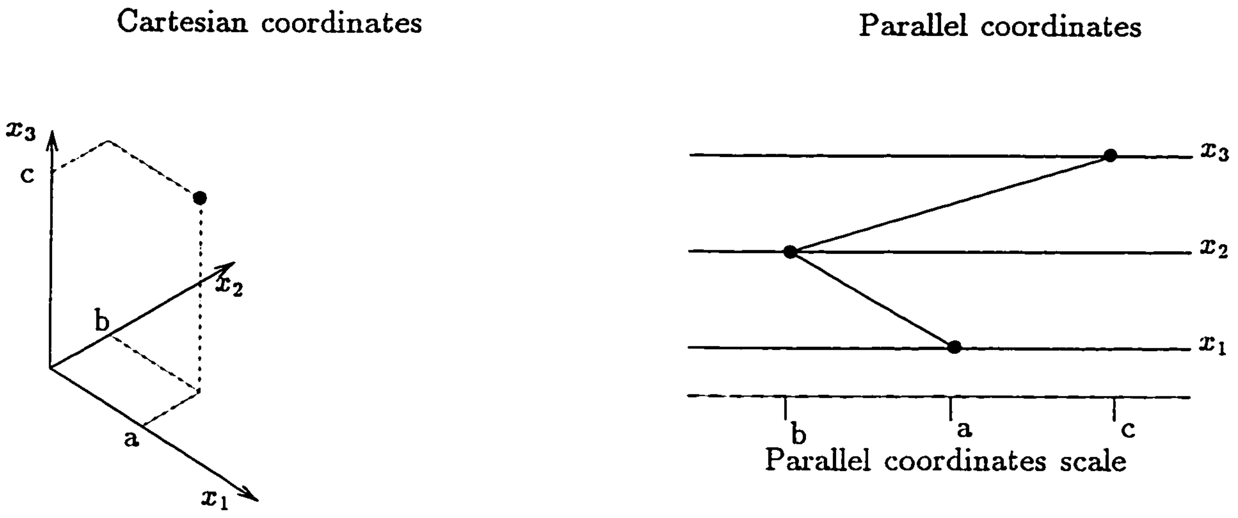
`If` $a \leq x_1 \leq b$ `then` $c \leq x_2 \leq d$ `with probability` $\hat{P}_E$ `and`
`the support of` n_E `observations.`

More sophisticated rules can be assembled by considering the union and intersection of events. This method of rule extraction from discovered events is similar in spirit to Narazaki's explanatory mechanism [89], designed to generate rules from trained neural networks.

The discovered events can also be interpreted graphically via a parallel event plot. I will first give a very terse overview of the parallel axes plot [70, 122], which is the basis for the event visualization aid. Incidentally, the parallel axes plot has been recently incorporated as an interpretive tool for database mining [80].

5.7.2 Parallel axes plot of data

When plotting d-dimensional data in Cartesian coordinates, the d-axes are orthogonal to each other. In contrast, parallel coordinates opt for d axes drawn parallel and equally spaced. To plot a multivariate point in $\Re^d$, each coordinate value is plotted on the corresponding parallel axis. The resulting d points are connected by a piecewise linear curve. Figure 5.18 shows how a point in $\Re^3$ is represented in parallel coordinates. The advantage of parallel coordinates is that all variables are treated symmetrically. A powerful duality between points and lines can be deduced in Euclidean and parallel coordinates [122]. However, in practice, parallel coordinates suffer from a number of limitations. When the data is dense, a parallel axes plot is rendered uninterpretable. The predicament is known as splotching [104], since the large number of intersecting and overlapping lines occludes structure. The process of thinning [104] attempts to address splotching by plotting random subsets of the data. Unfortunately, there is no guideline as to what subsets should be viewed and interpretations are no longer reproducible. In addition, other than Kendall's correlation coefficient [70], the parallel axes plot does not provide any quantitative



Cartesian coordinates

Parallel coordinates

Figure 5.18: Parallel axis plot of a point in $\Re^3$

information to describe or verify the visual analysis. These problems will in part be alleviated with the parallel event plot.

5.7.3 Parallel event plots

The uniqueness of the parallel axes event plot lies in its local rather than global focus. At most, only local subsets of data are displayed on a single plot. The 2 variations of the parallel axes event plot are introduced below, along with the information which can be directly extracted.

Plots of events only

When plotting a discovered event on parallel coordinates, only the vertices of the hyper-rectangle are displayed. Figure 5.19 exemplifies the parallel plot of an event in $\Re^3$. From a plot of events, we can easily detect certain types of multivariate structure.

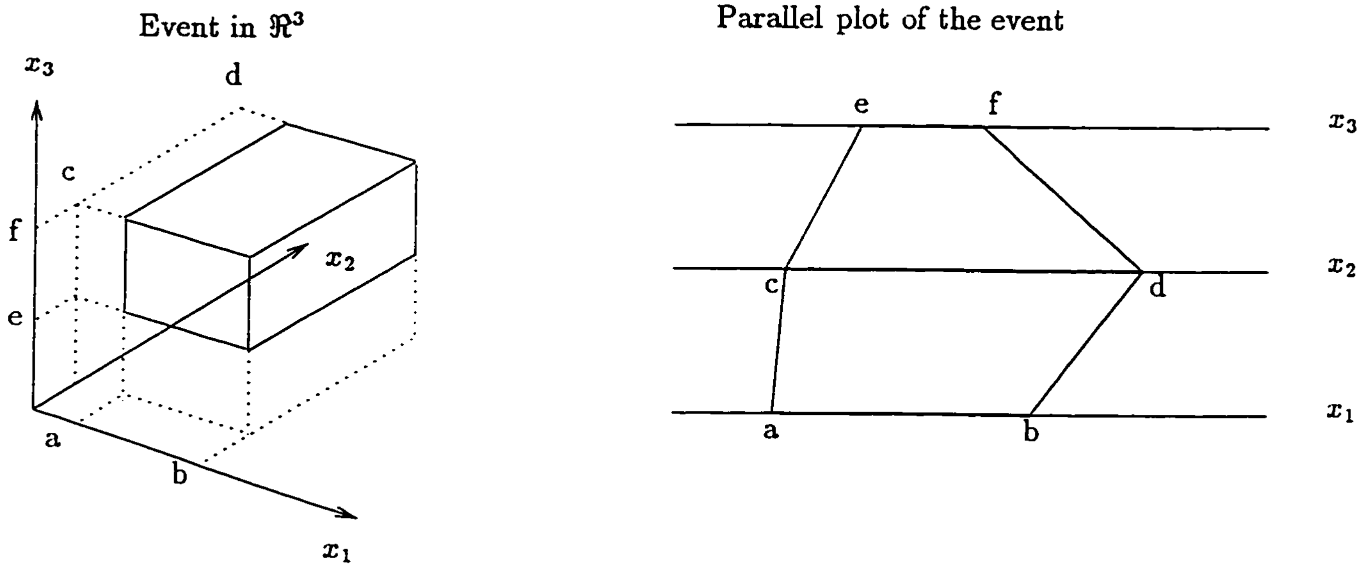


Figure 5.19: Parallel plot of an event in $\Re^3$

1. Clustering and multimodality can be easily identified in either the full dimensionality of the data or in a subset of dimensions. The probability density estimates can serve to quantitatively verify the identification of modes and can be communicated by the intensity of the shading. Figure 5.20 exemplifies a 2-dimensional cluster in data of 5 dimensions.

2. Class separability is another type of structure which can be readily spotted from an event plot. The event plot can reveal separability in the full dimensionality of the data or in only a subset of dimensions. Figure 5.21 shows events for a 2 category classification problem. We see that the classes are completely separable in the second dimension.

3. General associations among specific ranges of the variables are disclosed via event plots. For example, the 2 events in Figure 5.22 indicate that very small

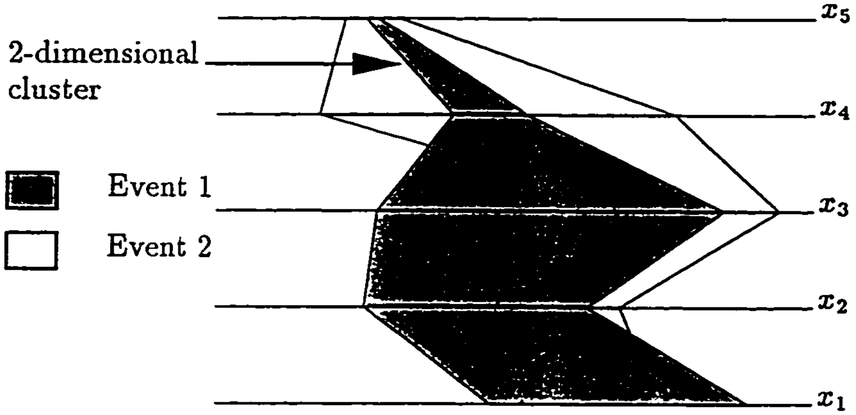


Figure 5.20: Detection of clusters from event plots

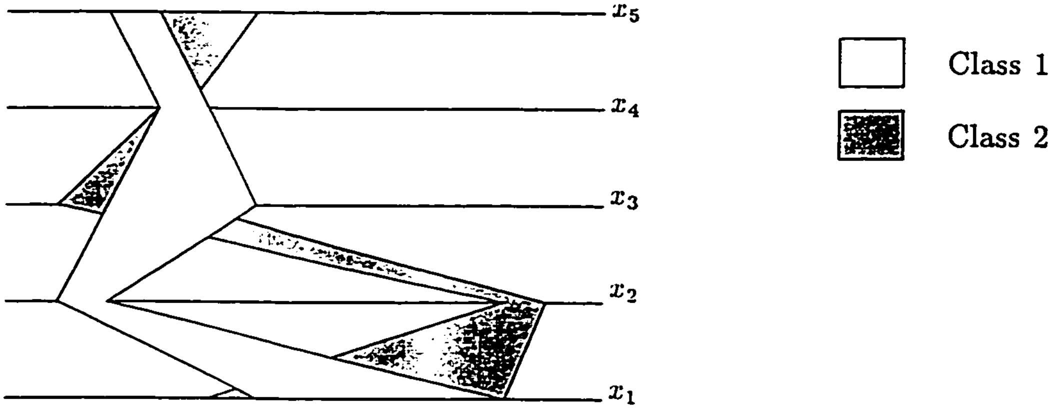


Figure 5.21: Detection of class separability from event plots

values of x_1 are likely to occur with very large values of x_5 while large values of x_1 are associated with small values of x_3, x_4 and x_5.

It is clear that by plotting events alone, we can gain insight into the multivariate structure of the data. To take advantage of the properties of standard parallel axes plots, as detailed in [122, 70], a second variation of the parallel event plot is proposed.

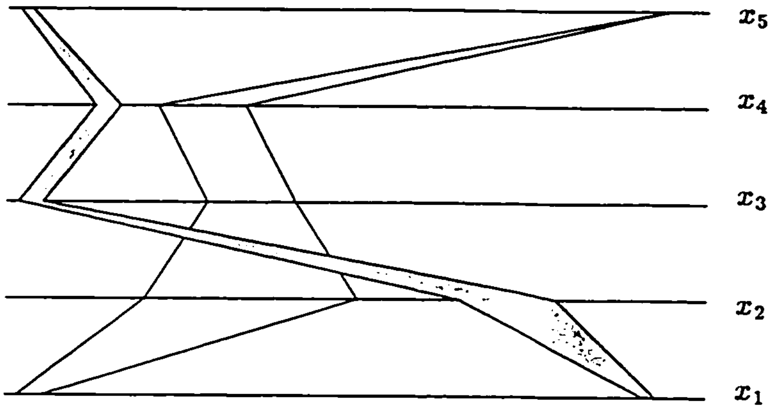


Figure 5.22: Detecting general associations from event plots

Plots of events and data subsets

Further insight into the local properties of the data can be gained by superimposing a subset of data points on an event plot. In the following, the first 3 observations apply when only the subset of data contained in an event is added to the event plot.

1. Linearity between 2 variables is easily detected as parallel lines between a pair of axes. When the parallel line behaviour extends for more than 2 variables, a hyperplane is suggested. Figure 5.23 plots an event and the subset of points that it covers. Note that the parallel lines between the variables x_1 and x_2 and again between x_2 and x_3 indicate that the data is linear in the first 3 dimensions, forming a 3-dimensional hyperplane.

2. Marginal densities can be visually ascertained by examining the subset of data points. By inspecting the x_1 axis in Figure 5.23, we recognize that the data is denser for lower values of x_1 and sparser for higher values.

3. One of the easiest structures to detect via traditional parallel axes plots is negative correlations between variables. This property is indicated by intersecting lines, creating a "cross-over" effect [122]. Referring again to Figure 5.23, notice the prominent cross-over of lines between axes x_3 and x_4. We can conclude that x_3 and x_4 are negatively correlated.

4. When the data is not overly dense, it is possible to plot the entire data set along with the events. Outliers can then be easily pinpointed. Not only can an outlying observation be identified, but we can also determine which particular dimensions exhibit deviant behaviour.

Although the information in the combination plot could be obtained from a standard parallel axes plot, the use of events enhances the interpretability. The plot of the entire data set may contain noise, outliers, or overlapping structures. On the other hand, events inherently delineate a subset of data that is homogeneous in some characteristic, as implied by the discovery objective. Therefore, working with an event's data subset can be much more rewarding from a discovery viewpoint.

Advantages of parallel event plots

The parallel plot of events has two additional advantages over conventional parallel axes plots of data. First, the problem of splotching is overcome by only showing significant events. Unlike random thinning procedures, there is no fear of losing valuable structural information. Secondly, the event plot implicitly carries useful quantitative information about the discovered structure, in terms of numerical event boundaries, statistical significance, probability and the number observations

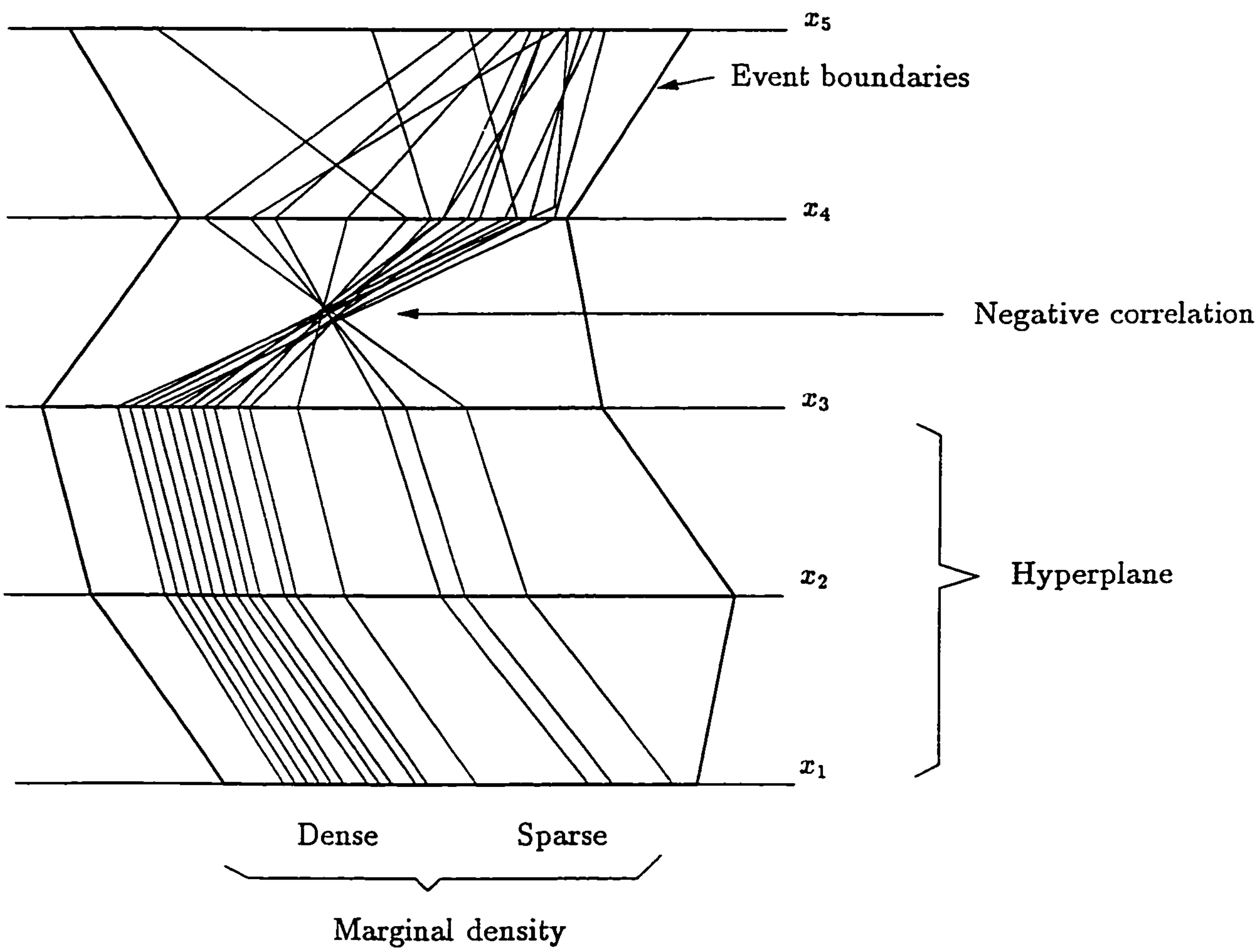


Figure 5.23: Detecting linearity, marginal density and negative correlation in a combined event and data subset plot

supporting each event.

5.7.4 Summary

In this chapter, I have presented a low-dimensional specialization of pattern discovery. The technique relies on recursive partitioning and evaluation of local regions of the sample space. Discovery objectives are applied to guide the partitioning process. The basic pattern discovery framework was extended to provide smooth representations of the data, using a general kernel method. Density estimation and classification were also presented as straightforward applications of discovered events. The significant event shift was defined for the detection of changing patterns while a novel event synthesis algorithm was developed to decipher high-dimensional data. The interpretability of discovered events was illustrated through the use of simple rules and the versatile parallel event plot. All in all, this chapter demonstrates that conceptually, events provide a fundamental and flexible description of data organization. Moreover, events can form the basis of exploratory and inferential tasks. The duty of the coming chapter is to support these conceptual arguments with some empirical results.

Chapter 6

Experiments and Discussion

The previous two chapters built up many theoretical arguments and proposed a number of algorithms for pattern discovery. In this chapter, a subset of these items are empirically demonstrated and tested by way of experiments with simulated and real data. The presentation will be divided into 2 main groups of experiments. The first group will primarily illustrate the basic properties of the pattern discovery methodology, including scale invariance, noise rejection and discovery by different criteria. The second series of experiments applies pattern discovery to several real life data sets, highlighting exploratory analysis, outlier detection and classification. Throughout, the focus is upon the interpretability of the discovery results and the versatility of the basic event framework.

Appendix 2 contains the details on the generation of the artificial data sets and explains where the real-world data sets may be obtained via the Internet.

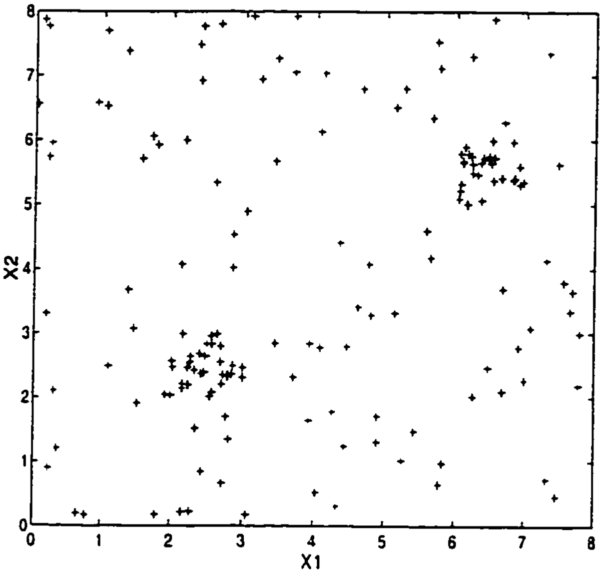


Figure 6.1: Example of bivariate clusters

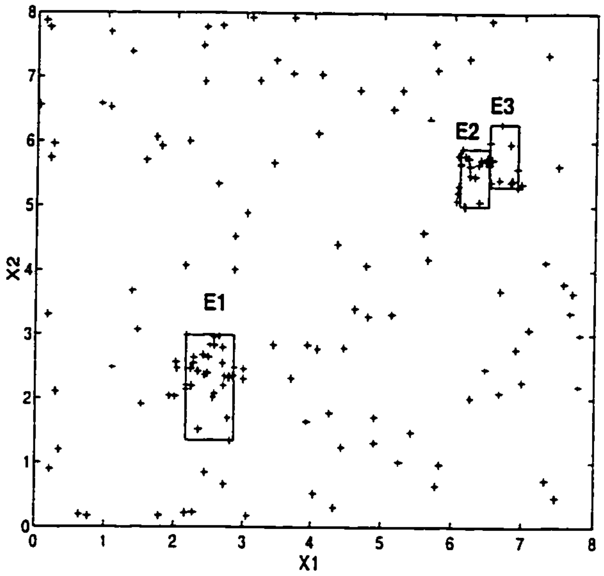


Figure 6.2: Discovered events

6.1 Basic properties

6.1.1 Discovery by different criteria

To begin, I will use some simple 2 and 3-dimensional data sets to exemplify the types of organization detected by the discovery objectives.

Concentration

A simple bivariate data set is shown in Figure 6.1. There are 2 visible clusters among a background of uniformly random points. Figure 6.2 is the result of applying discovery with the concentration objective. Visually, 2 groups have been identified. An examination of the estimated pdf values confirms that the discovered events have higher pdf values than the background event, defined to be the rectangle containing

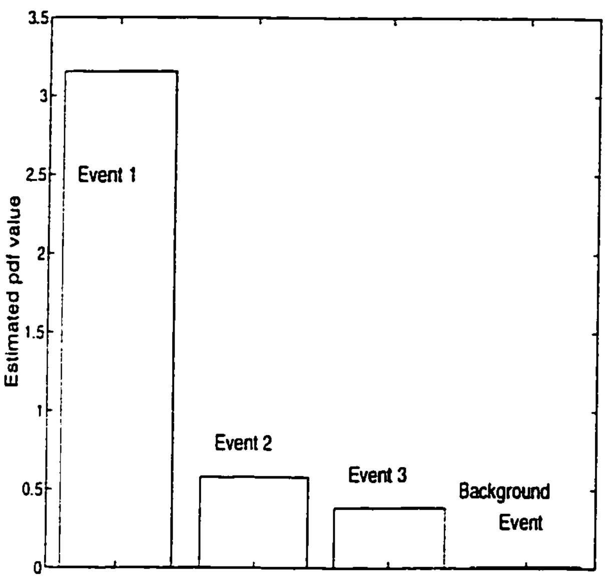


Figure 6.3: Estimated pdf values for the discovered and background events

Table 6.1: Discovered group centres

Group	Estimated Centre	Noise-free Centre
1	(2.52, 2.39)	(2,2)
2	(6.27, 5.56)	(6,6)

the data. Finally, by applying the cluster determination method of Section 5.4.3, two groups are verified and the centres are within 8% of the noise-free values. In general, the concentration objective is suited for the detection of natural grouping tendencies in the data.

Dependency

At the outset, the data in Figure 6.4 appears to be just random noise. However, using the dependency objective, an event is discovered near the center of the plot,

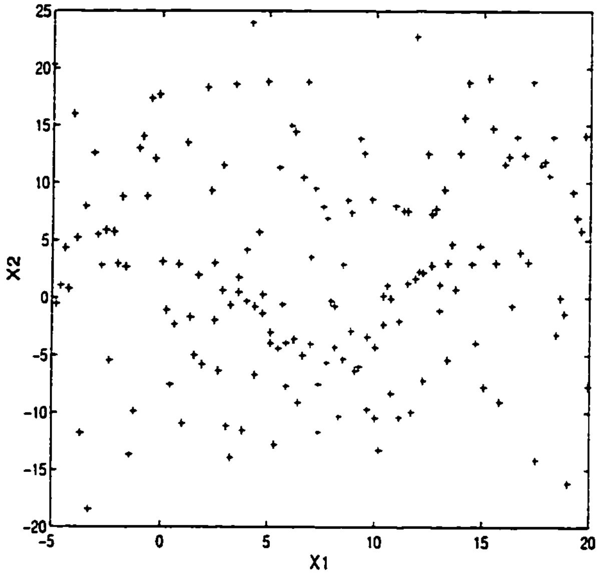


Figure 6.4: Example of subtle dependency

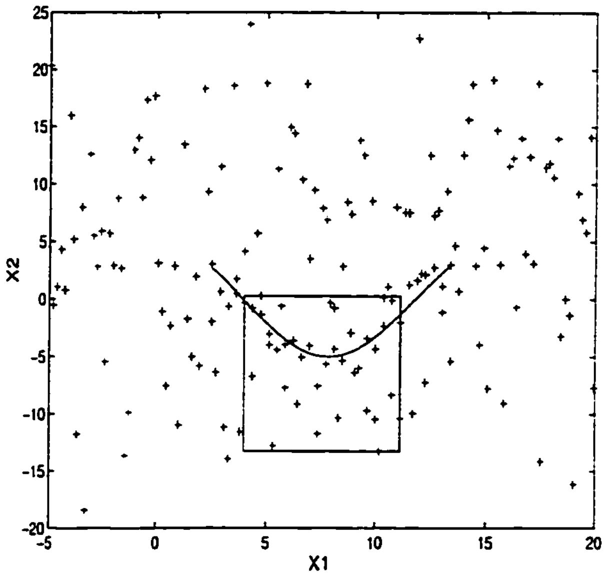
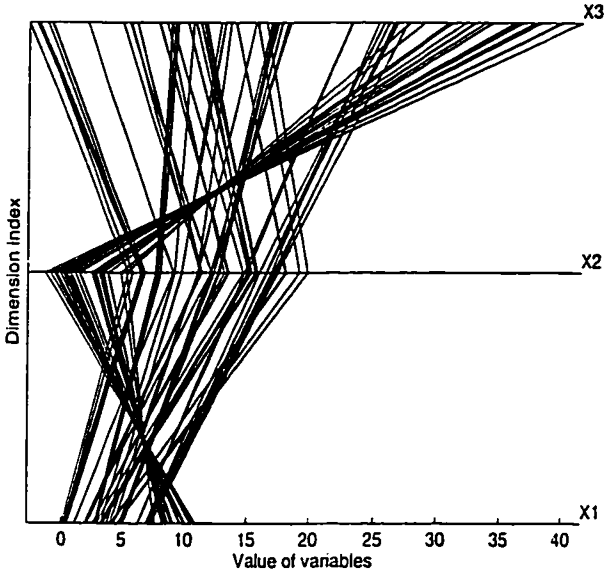


Figure 6.5: Discovered event

as shown in Figure 6.5. In fact, embedded in the apparently random data is a local, nonlinear relationship between the variables. The known governing relationship is superimposed. Note that unlike the concentration objective, the discovered event does not enclose a region of necessarily high data concentration. Rather, the event captures a region of space where X_1 and X_2 exhibit a strong interdependence.

Linearity

Linear dependence is an important special case of the general dependency among variables. Figure 6.6 shows data points lying on 3 different planes in $\Re^3$. The data points have been perturbed by uniform random noise. Sequential, recursive discovery using the linearity objective unveils that indeed there are 3 unique linear relationships indicated by the 4 events in Figure 6.7. Events 3 and 4 define the same relationship. However, they are not redundant for their intersections between



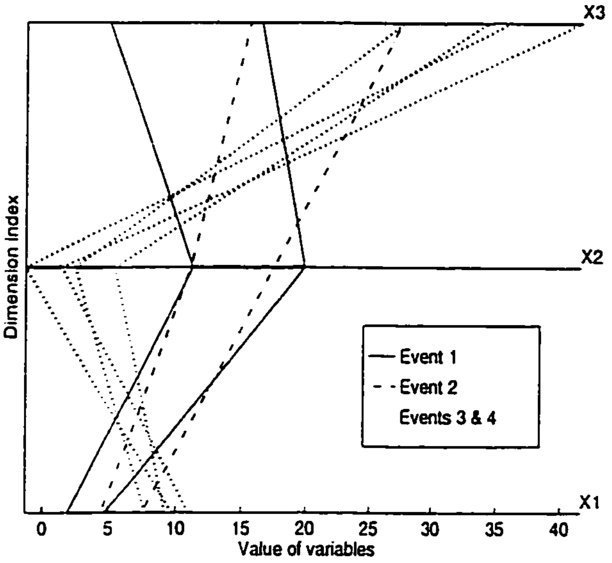


Figure 6.6: Example of 3 planes in $\Re^3$

Figure 6.7: Events from linearity discovery

the axes emphasize that on this plane, the variables are negatively correlated.

Without *a priori* knowledge, it is not obvious how the data in Figure 6.6 should be separated for analysis, especially since the relationships overlap in every dimension. Discovery by the linearity objective paves the way for local analysis. In fact, a standard regression can be performed on each subset of data. The F-statistic value and the multiple r^2 together give a quantitative indication of the strength of the linear dependency. As summarized in Table 6.2 the data within each event exhibits a strong linear dependency, with r^2 close to 1 and the estimated F-value $\hat{F}$ exceeds the critical value F_C^α.

Table 6.2: Strength of linear dependency

Event	r^2	$\hat{F}$	F_C^α
1	0.992	957.6	7.7
2	0.983	291.9	9.4
3	0.998	2946.0	7.9
4	0.973	166.0	10.1

6.1.2 Noise tolerance

Two aspects of noise tolerance are demonstrated here, recovery of information tainted with noise and secondly the outright rejection of data with no organization. To illustrate the first, consider the simple bivariate data shown in Figure 6.8 where data with a parabolic trend is salted with 20 outliers. The solid line is the true curve while the dotted line is the result of a quadratic loess fit with the smoothing parameter set to 1. The presence of outlying points have resulted in a poor fit with a sum-of-squared error of 118.4. Of course, we can use the bisquare method [67] to obtain a robust fit, but with pattern discovery the useful information can be immediately identified without the need of iterative refitting. In Figure 6.9, pattern discovery with the concentration objective has successfully filtered out the outliers in the data. Loess fitting can be performed again on the contained data points, yielding a much improved fit, as the sum-of-squared error plummets to 2.1. This example confirms that organization can be detected in the presence of outliers. In addition, the example shows that pattern discovery can be easily used in conjunction with existing data analysis tools.

The purpose of the next basic experiment is to exemplify the type of data which

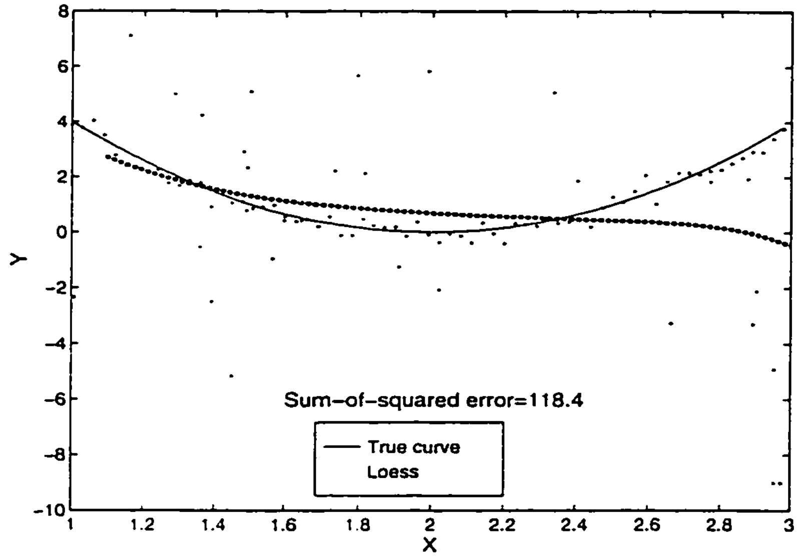


Figure 6.8: Loess fit to raw data

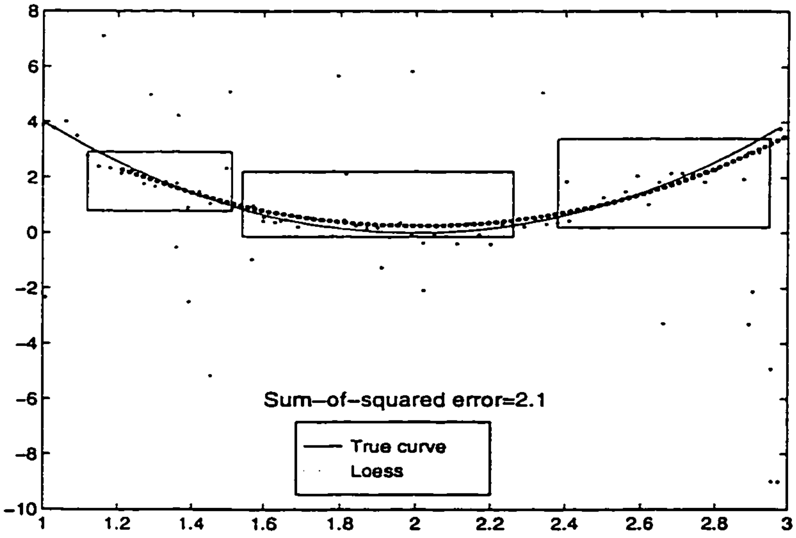


Figure 6.9: Pattern discovery and loess fit

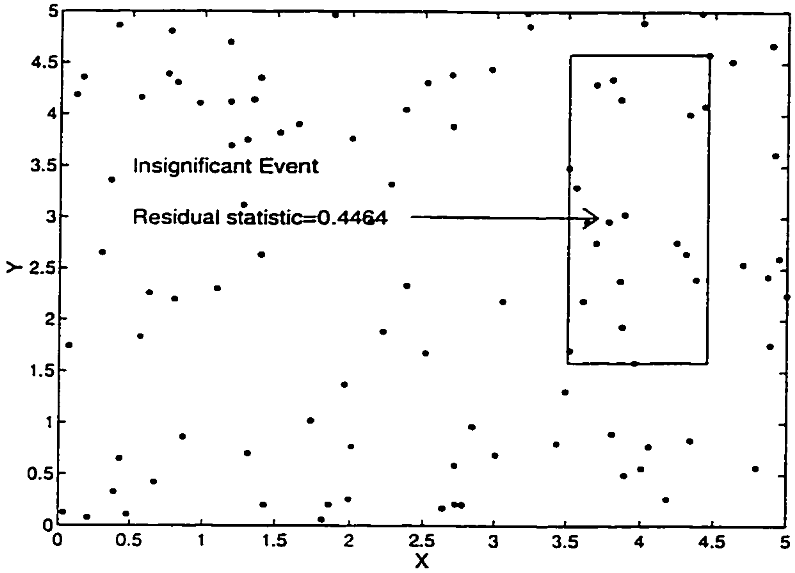


Figure 6.10: Noise rejection: concentration objective

pattern discovery judges as containing no information according to the discovery objective. We desire strong discrimination against data that is counter-objective. Again for ease of visualization, bivariate data is used. For the concentration objective, data that is uniformly distributed throughout the space of interest is deemed insignificant since there are no regions of unusual concentration. Discovery on the data in Figure 6.10 resulted in no significant events. As an example, the event unveiled from one sequence of discovery is shown. The residual statistic indicates that it is indeed insignificant at the 5% significance level, since $0.4464 < z_{1-0.05/2} = 1.96$.

A lack of dependence is portrayed in Figure 6.11. Here the value of Y is centered around 3 regardless of the value of X. As anticipated, dependence discovery turns up no significant events. The event shown is the maximum from one sequence of discovery. Again, the residual statistic indicates that it is insignificant at the 5%

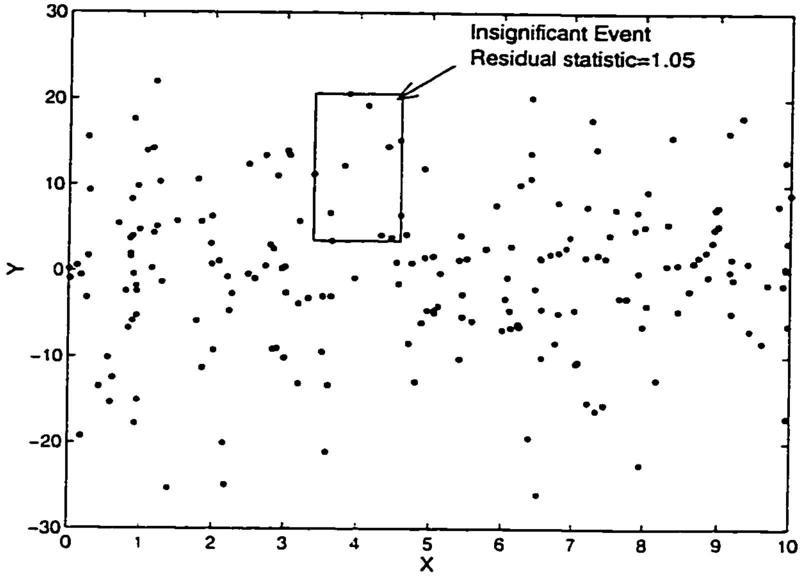


Figure 6.11: Noise rejection: dependence objective

significance level.

With linearity, we expect that data governed by nonlinear relationships will not be detected as significant. However, local subsets of the data may be approximated by a linear relationship. An example is shown in Figure 6.12. Clearly, the data is not linear and the dotted regression line fits very poorly. Hence, the event encompassing the whole data set is insignificant. However, global nonlinearity does not preclude local linearity and discovery locates a subset of data where a linear model is quite appropriate. Although I have only exemplified discovery with 2 variables, the noise rejection properties extend to the multivariate case.

It is worthwhile to mention that a subspace which is judged as lacking organization by one objective, may very well be tagged as informative by a different objective. For example, in a highly concentrated region the variables may be statistically independent. Hence, the definition of organization and noise depend on

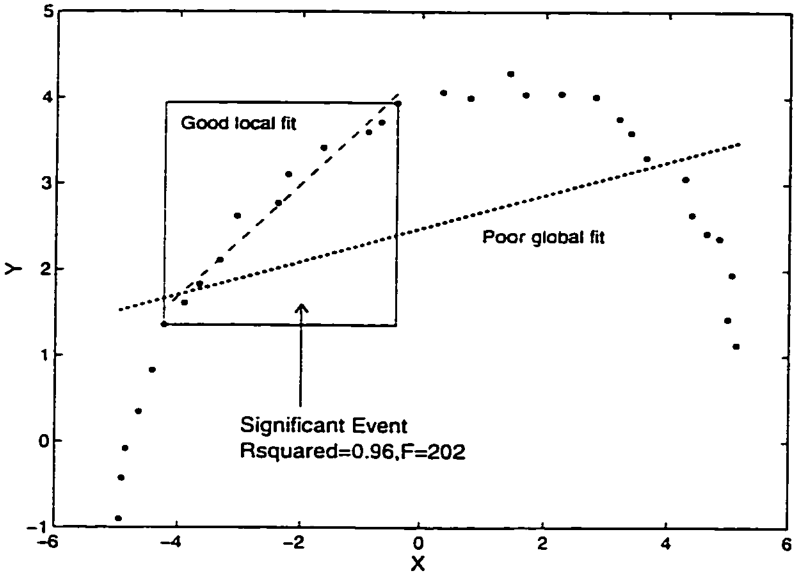


Figure 6.12: Noise rejection:linearity objective

the discovery objective invoked.

6.1.3 Discovery amid non-centralized noise

This section will examine the ability to discover information when the noise distribution is skewed towards a region of the sample space. Figure 6.13 shows the nonlinear relationship between intensity and mean angle taken from a physics problem. The governing equation is

$$y = (1 + \cos(2x)^2)\frac{\exp(-\sin(x))}{\sin(x)^2} + \delta \tag{6.1}$$

where δ is a Gaussian variable with 0 mean and standard deviation 10^{-3}. From the governing relationship, 1000 points were generated with additive Gaussian noise. In addition, the data has been corrupted with 250 uniform background noise points

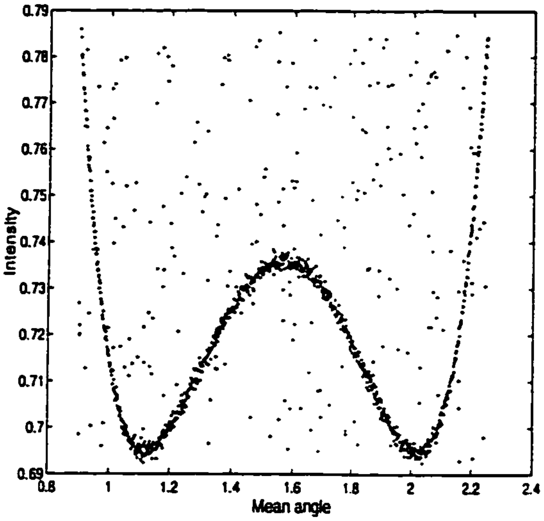


Figure 6.13: Raw data

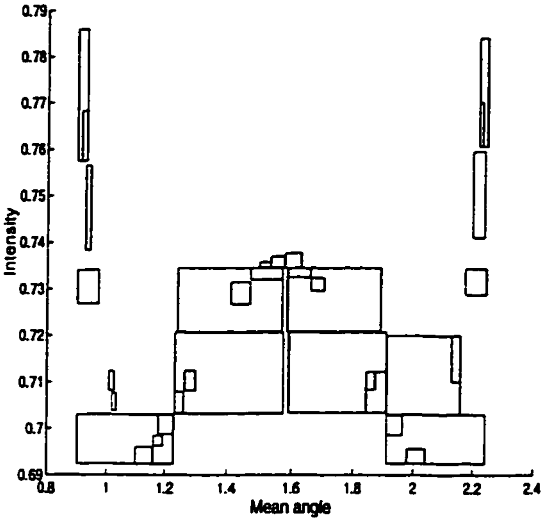


Figure 6.14: Discovered events

(x, y) with $0.89 \leq x \leq 2.24$ and $0.69 \leq y \leq 0.79$. Application of low-dimensional pattern discovery resulted in 34 events (Figure 6.14). Notice that the noise has essentially been filtered out.

We can proceed to make use of the discovered information for predicting intensity values for given mean angles. Based on the data contained in each event, a centroid point can be computed. These centroids can then be spline-fitted. Intensity values can be predicted for given mean angles, simply by linearly interpolating between the spline points. The result on a test set is shown in Figure 6.15. Observe that the predicted values quite reasonably resemble the actual points. The key point is that discovery does not perform averaging of the data. The presence of non-centralized background noise, skewed towards the upper portion of the sample space, would force the mean to occur substantially above the actual curve.

To demonstrate this phenomenon, a 3-layer feedforward neural network us-

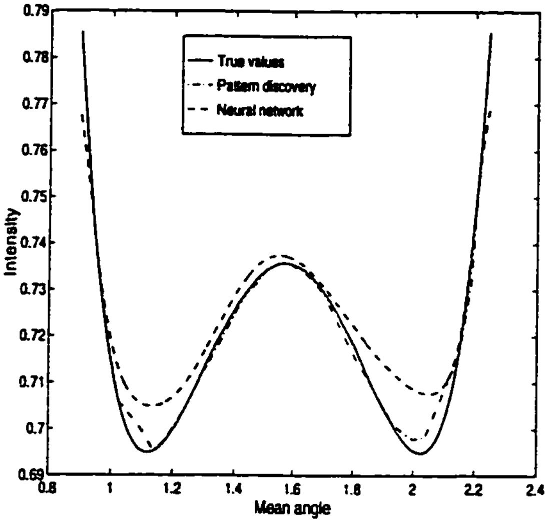


Figure 6.15: Predicted values

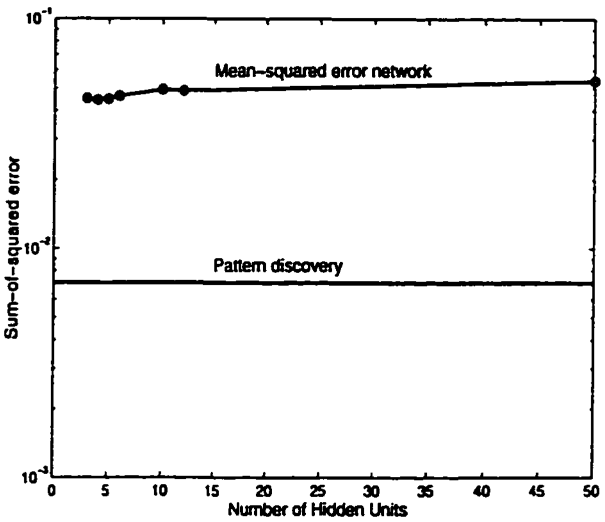


Figure 6.16: Sum-of-squared errors

ing a mean-square error objective function was trained with this data using the Levenberg-Marquardt algorithm. Even when a large number of hidden units is commissioned to learn the data, the sum-of-squared error could not be further reduced (See Figure 6.16). In fact, with larger networks, over-fitting was observed, resulting in larger errors. This is attributed to the skewness in the data.

From this example, we observe the important property of robust discovery in the presence of non-centralized noise.

6.1.4 Scale invariance

Next, I exemplify the important property of scale invariance. Consider the housing data shown in Figure 6.17 with units in US dollars and miles. The scaled housing data is the same data set, plotted in Figure 6.18 but in units of Japanese Yen and kilometers. Each sample point represents the value of a home located at a certain

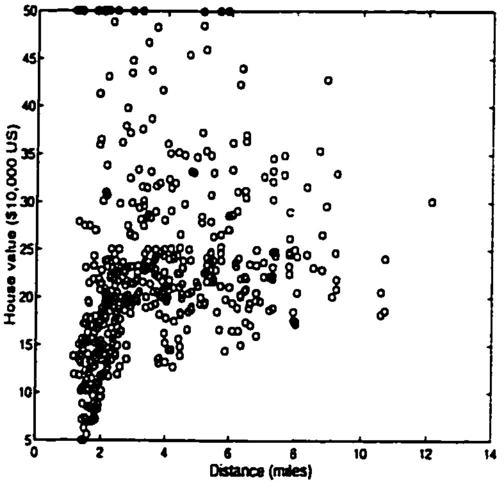


Figure 6.17: Housing data

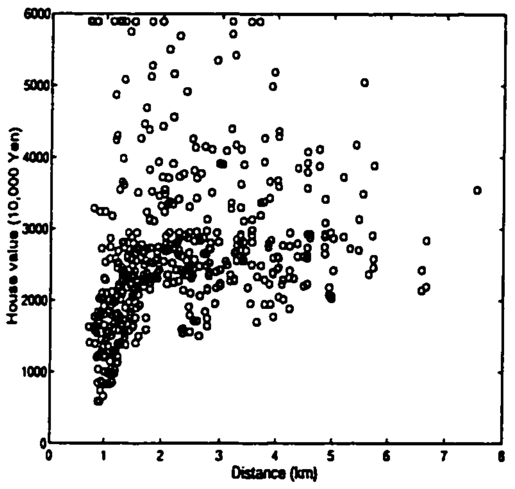


Figure 6.18: Scaled housing data

distance from the city centre. When the low-dimensional pattern discovery algorithm was applied, 13 events were discovered in each of the scaled and unscaled data sets. Figures 6.19 and 6.20 show the discrete pdfs based upon the discovered events. We observe that the shape of the density function is preserved with scaling. This suggests that the relative magnitude of event density values remains constant, a necessary and sufficient condition for scale invariance. To demonstrate the usefulness of scale invariance consider the following hypothetical query.

> Is it more likely to find a $170,000 home that is 3 miles from the city center (Home A) or a $190,000 home that is 5.5 miles from the city center (Home B)?

Ideally, we should arrive at the same answer, regardless of how the distance (kilometers or miles) and home value (dollars or Yen) are specified. In fact, using the pattern discovery by partitioning to answer this query, we find that Home A is 2.2

Table 6.3: Ratio of probability density values before and after data scaling

	Home Specification			
	A (3 miles, \$170,000) B (5.5 miles, \$190,000)		A (1.86 km, 20,044,700 Yen) B (3.42 km, 22,402,900 Yen)	
Method	pdf ratio	Answer	pdf ratio	Answer
Partitioning	5.7	A	5.7	A
Optimization	2.0	A	1.9	A
PNN_1	1.2	A	0.6	B
PNN_2	1.1	A	1.1	A

times more likely regardless of how distance or home value are specified. Using pattern discovery by optimization, a similarly consistent answer is obtained. In Table 6.3, we note that the pdf ratios for partitioning and optimization are different due to deviation in the event locations and sizes. However, the important point is that the conclusion is consistent with both methods. As mentioned earlier, with the discovery by optimization, we can only expect approximate invariance, as suggested by the slightly different pdf ratios before and after scaling.

In contrast, methods based on distance measures are generally scale sensitive. As an example, consider the probabilistic neural network (PNN) that uses the Euclidean distance. The PNN is chosen for this illustration because like event-based discovery, it *directly* yields an axiomatically true pdf. Table 6.3 indicates that a seemingly harmless change of scale has reversed the decision of the PNN (PNN_1). Hopefully, the prudent practitioner would normalize the data or optimize the PNN smoothing parameter to mitigate the effects of scale. The consequence of normalizing both data sets is shown in the last row (PNN_2). The decision is now invariant to scaling and is consistent with that of event-based discovery.

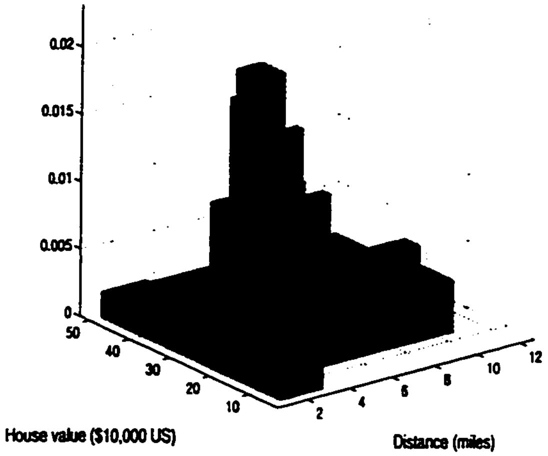


Figure 6.19: Discrete pdf for housing data

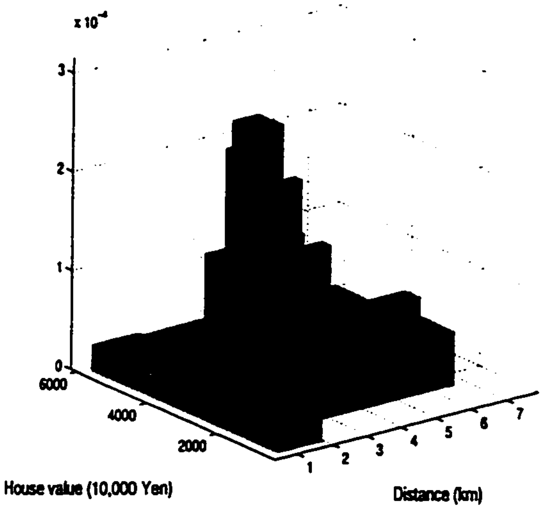


Figure 6.20: Discrete pdf for scaled housing data

The advantage of event-based discovery is that the often ad-hoc standardization of feature variables can be avoided and sensitivity to different magnitudes in the features is minimized.

6.1.5 Partitioning approximation to optimization

The next two experiments show that given a sufficient number of data points, low-dimensional discovery by the concentration and dependency objectives can be adequately approximated by criterion-driven partitioning.

When using the concentration objective, we need to show that the partitioning and optimization approaches identify similar local optima in the density function. Consider the bivariate lipid data shown in Figure 6.21. This data is due to

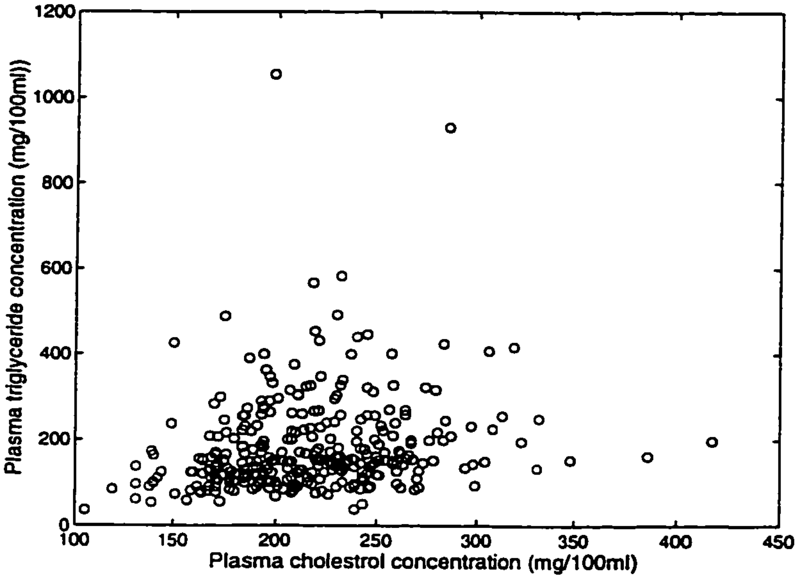


Figure 6.21: Lipid data - diseased subjects

Scott [105] and represents two lipid measurements, triglycerides and cholesterol, taken from 320 male subjects with heart disease. Although the data appears unimodal, it is actually bimodal [104], as discovery will reveal. Figures 6.22 and 6.23 are the detected events from optimization and partitioning discovery respectively. In both cases, the concentration objective was used. At first glance, the events appear quite different. Using the clustering determination criteria discussed previously, both sets of events yield 2 clusters. The values are reported in Table 6.4 and are indicated by circles in the figures. The coordinates of the centers obtained from partitioning are within 7% of their optimization counterparts. The estimated pdf values are a little more dissimilar, but more importantly the discrimination of the local maxima from the surroundings is maintained. This quality of the approximation is clearly captured in the discrete density plots of Figures 6.24 and 6.25. Incidentally, the discovered centers agree closely with those found by a mixture of

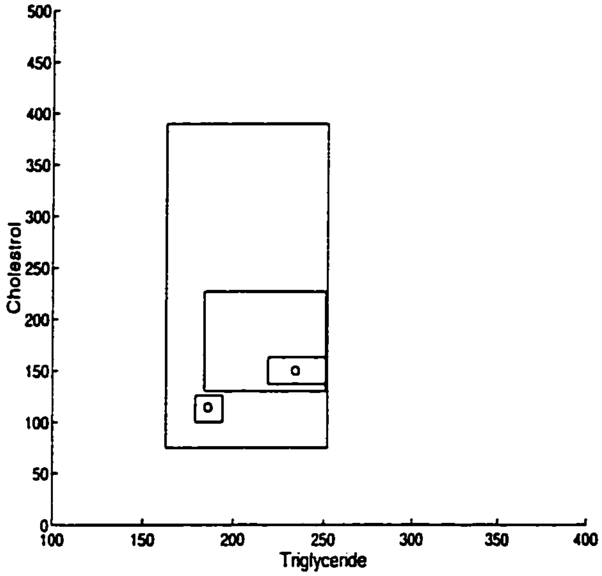


Figure 6.22: Events from optimization

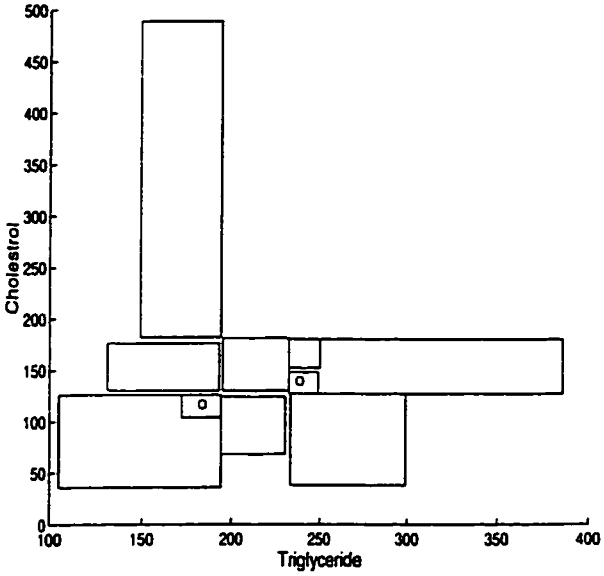


Figure 6.23: Events from partitioning

Table 6.4: Discovered centers by partitioning and optimization

Method	Centers	pdf values
Optimization	(186.1,114.4) (233.8,149.7)	1.35×10^{-4} 1.2×10^{-4}
Partitioning	(183.8,116.6) (238.4, 139.5)	1.1×10^{-4} 1.05×10^{-4}
Mixture of kernels [105]	(185,122) (233,145)	- -
Self-organizing map	(185,118) (223,139)	- -
Fuzzy c-means	(193,118) (235,171)	- -

Note: For the self-organizing map, a 6 × 6 grid was used with a neighborhood of 1. The fuzzy c-means algorithm was invoked with a fuzzy parameter of 3.

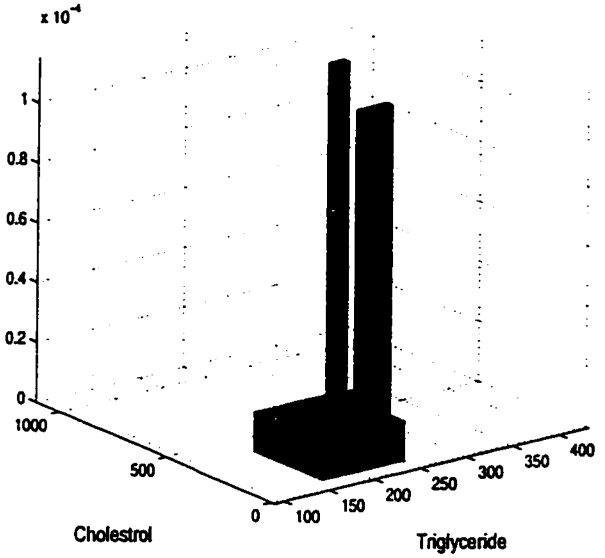


Figure 6.24: pdf from optimization

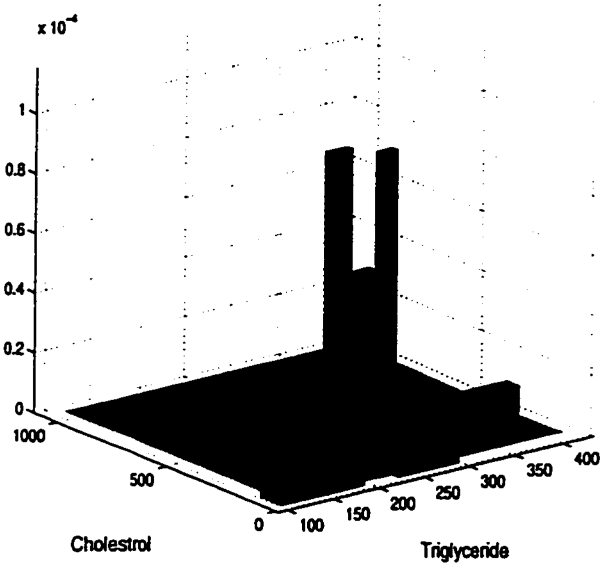


Figure 6.25: pdf from partitioning

Gaussian kernels [105] and independently, by a self-organizing map. Fuzzy c-means clustering [16, 17] locates a centre with a deviant second coordinate, but otherwise is also within the same range. From this example, we see that with adequately dense data, partitioning can approximate optimization with the concentration objective. Further, in this example, results of both discovery approaches are in agreement with findings of other credible methods.

With the dependency objective, we desire that discovery by partitioning and optimization identify similar regions of significant dependence. For this illustration, I consider two variables used in an environmental pollution study [21]. The objective is to determine how ozone level, the standard indicator of smog severity, depends on solar radiation. Discovery by optimization and partitioning both yield 1 significant event at the 5% significance level. To enhance the comparison, the insignificant events are also shown in Figures 6.26 and 6.27. The leftmost event

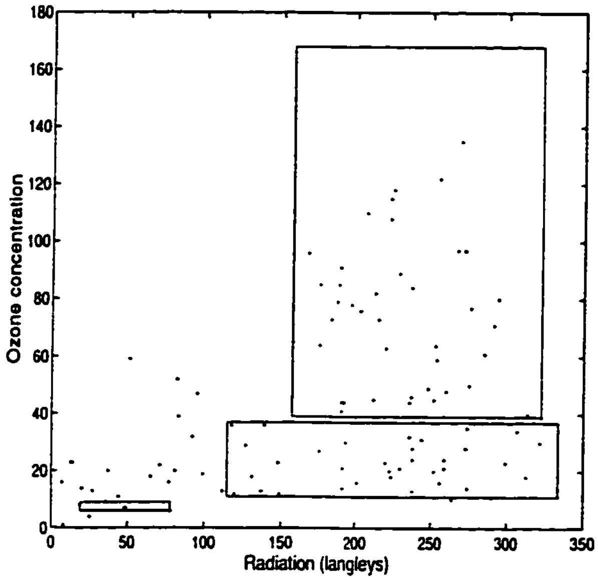


Figure 6.26: Dependency - optimization

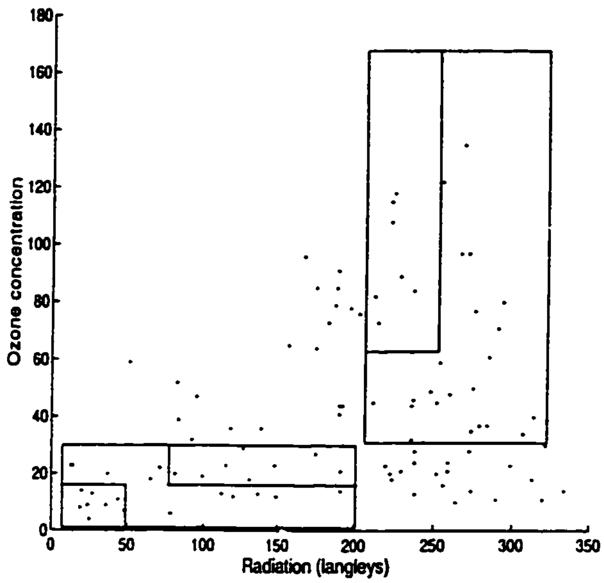


Figure 6.27: Dependency - partitioning

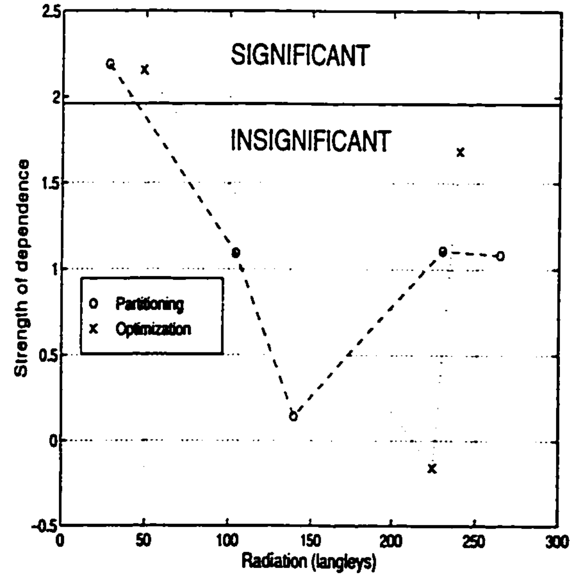


Figure 6.28: Diminishing dependence

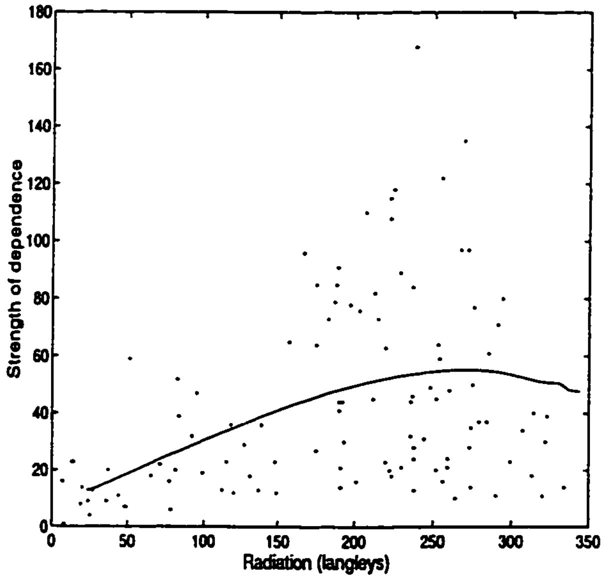


Figure 6.29: Loess fit

is the significant event in both plots. The important characteristic to compare is the detected variation in the ozone-radiation dependence. We can consider the residual statistic as a measure of the strength of the dependency between ozone concentration and radiation levels. Using the radiation level at the centre of the event as the abscissa value, we can graph the statistic value as in Figure 6.28. The graph shows that at low radiation levels, ozone is strongly dependent on radiation while as radiation levels increase, the dependency generally diminishes. Although a slight rise in dependence is observed at the highest radiation levels, the dependence remains statistically insignificant. This fluctuation in the dependency is suggested by the events of both partitioning and optimization discovery. In Figure 6.29, a superimposed loess curve with smoothing parameter equal to 1 and linear local fitting further suggests that the dependency tails off at higher levels of radiation. This variation in dependency is also in line with the coplot analysis of Cleveland [30]. In this example, we see that discovery by partitioning can also approximate discovery by optimization when using the dependency objective.

To close this subsection, I remark that the main advantage of the partitioning approximation is the rapid speed at which significant events can be uncovered in two or three dimensions. The obvious limitation is that partitioning is of exponential complexity, and thus application to data of higher dimensionality is impractical. In fact, the very act of partitioning requires adequately dense data for meaningful statistical testing. Hence, partitioning is generally ill-suited to sparse data sets. Furthermore, with its frequency equalization tendency, partitioning is inclined to over-represent Gaussian-type groupings, resulting in an excess of significant events. Despite this hoard of shortcomings, partitioning can be used in the analysis of 2-dimensional orthogonal projections of the data, where the low dimensionality mitigates the ill effects of its incapacities. The example of Section 6.1.7 will demonstrate that partitioning can thus serve as a starting point for analyzing high-dimensional data.

6.1.6 Classification properties

Although the focus of pattern discovery is not on classification, an event-based classifier does exhibit some interesting properties. In this section, I will compare event-based classification to the stalwart of statistical pattern classification, the nearest neighbour and to the mainstay of model-free estimators, the multilayer feedforward neural network. Consider first the nonconvex and linearly inseparable classes of Haykin [61], reproduced in Figure 6.30. The task is to distinguish between the two interlocking classes on the basis of 100 training points per class. The test set consisting of 500 points is shown in Figure 6.31. To properly estimate the error rate, 10 such training and testing sets were generated. In each trial, the low-dimensional discovery algorithm was applied to the individual classes with the concentration objective. No more than 19 events per class were detected. Figure 6.32 is a contour plot of the discrete density function estimated from the events of one trial. Adherence to the class boundaries is evident. The detected events were used directly for classification. Table 6.5 reports the results averaged over the 10 trials. For comparison, the performance of the nearest neighbour algorithm and a 4-layer neural network with architecture 2-5-4-1 are included. We note that the

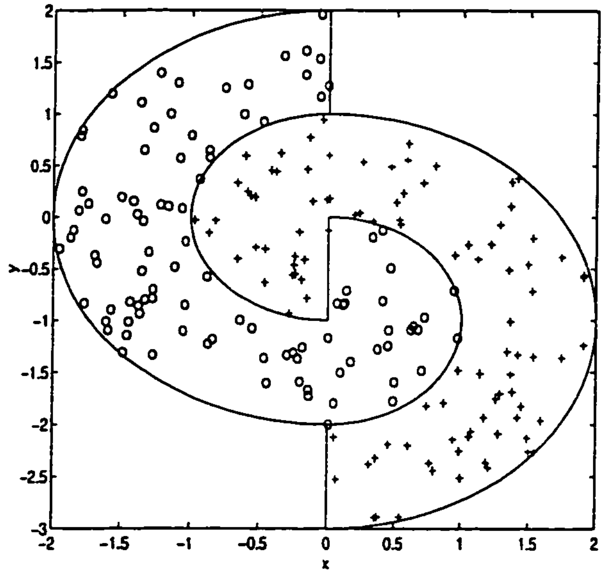


Figure 6.30: Training data and class boundaries

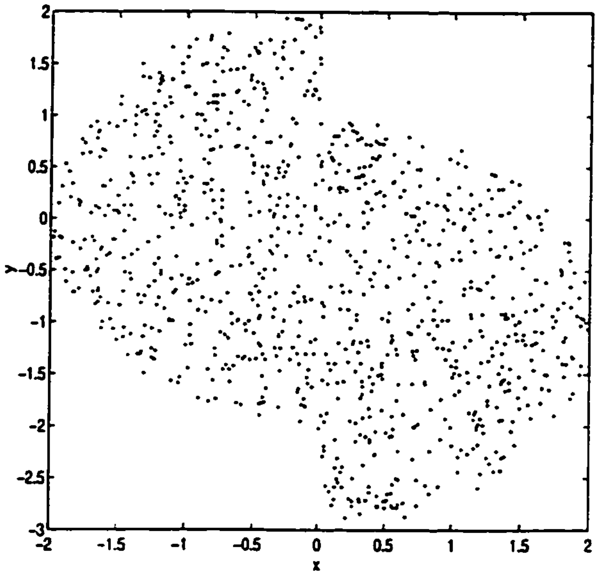


Figure 6.31: Test data

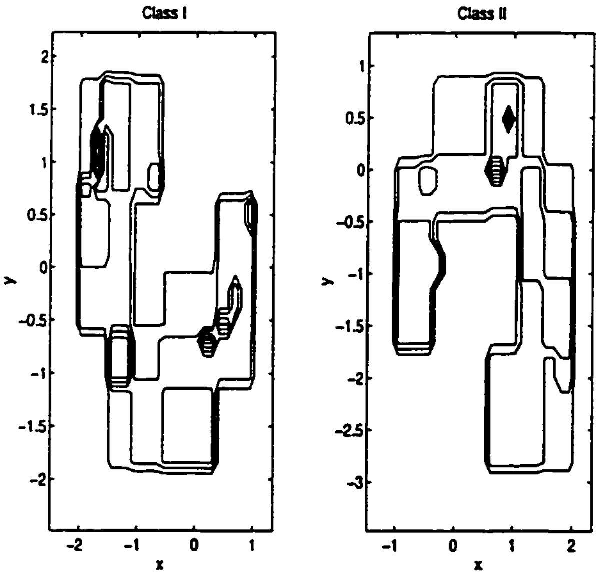


Figure 6.32: Contours of the estimated discrete densities

Table 6.5: Average classification results for Haykin's data

	Event classifier	Nearest neighbour	4-layer neural network
Average Error rate	7.56	5.60	4.24
Average Response Time	3.26 ms	14.1 ms	<1 ms

event-based classifier responds 3 times as fast as the nearest neighbour classifier, but with a slightly higher error rate. The majority of additional errors committed by the event-based classifier are actually points which have been rejected. Disregarding rejected points, events only err on 0.83% more occasions than the nearest neighbour and 2.2% more than the neural network.

The robustness of the classifiers are also of interest. Noise, in terms of mislabelled training samples were added to each class at varying percentages of the "clean" data. As shown in Table 6.6, error rates for both the event and the nearest neighbour classifiers worsen with increasing noise corruption, but the event classifier appears to be more robust. With a different choice of k (# of neighbours) or perhaps using a different metric, the nearest neighbour error rate may improve [88]. However, without knowledge of the underlying noise density, these choices are not obvious. The robustness of the neural network is clearly the most impressive. Figure 6.33 shows that the performance of the pattern discovery classifier is between that of the neural and statistical classifiers.

In assessing classifiers, the rate of convergence to the Bayes error rate is of theoretical interest. For this study, I consider two overlapping bivariate Gaussians with unit variance. The known Bayes rate is 15.87% and can be achieved with a simple

Table 6.6: Average error rates for noise corrupted data

Noise level	Event classifier	Nearest neighbour	Neural network
10 %	11.2	8.7	4.5
25 %	12.6	16.9	5.0
30 %	13.4	18.5	5.12
50 %	17.3	27.2	6.34
Rate of increase (error rate/ % noise)	0.18	0.44	0.041

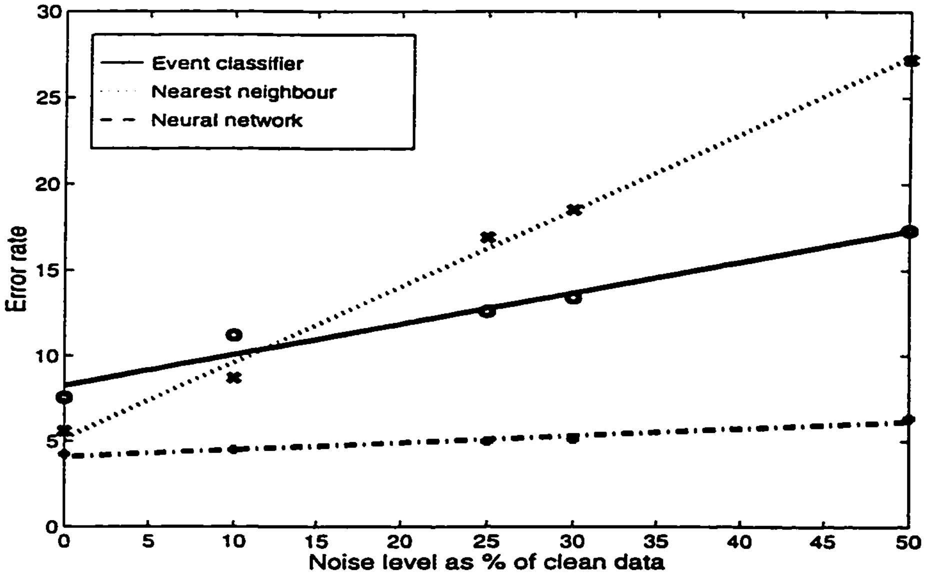


Figure 6.33: Error rates with increasing noise

Table 6.7: Convergence to Bayes rate (15.16%)

Sample size	Event classifier	Nearest neighbour	Neural network
1000	22.5 (17.0)	22.5	16.5
3000	18.58 (16.42)	22.15	15.83
4000	18.11 (16.0)	21.67	15.75
7000	17.80 (15.32)	21.56	15.17

linear discriminant classifier.[1] The error rates for this experiment are reported in Table 6.7. Although the event classifier does not converge with remarkable swiftness, its error rate does improve more quickly than that of the nearest neighbour.

Based on this example, the event classifier converges at a rate of $O(N^{-0.13})$ while the nearest neighbour dawdles along at a rate of $O(N^{-0.023})$, where N denotes the sample size. Although the neural classifier exhibits the best convergence to the asymptotic limit, the performance of the event classifier does not lag far behind. In fact, by employing kernels as discussed in Section 5.4.4, the event classifier comes within 1% of the theoretical limit. The corresponding error rates are shown in brackets in the first column of Table 6.7. Figure 6.34 graphically summarizes the results of Table 6.7.

In this section, the experiments have illustrated some properties of the event-based classifier. In particular, we see that the event classifier performs comparably to other established methods on a linearly inseparable problem. The example also showed that quite arbitrary decision boundaries can be accommodated. In addition,

[1] It is argued in the Appendix Section A.5 that an event classifier constructed from maximum entropy recursive partitioning is Bayes risk consistent.

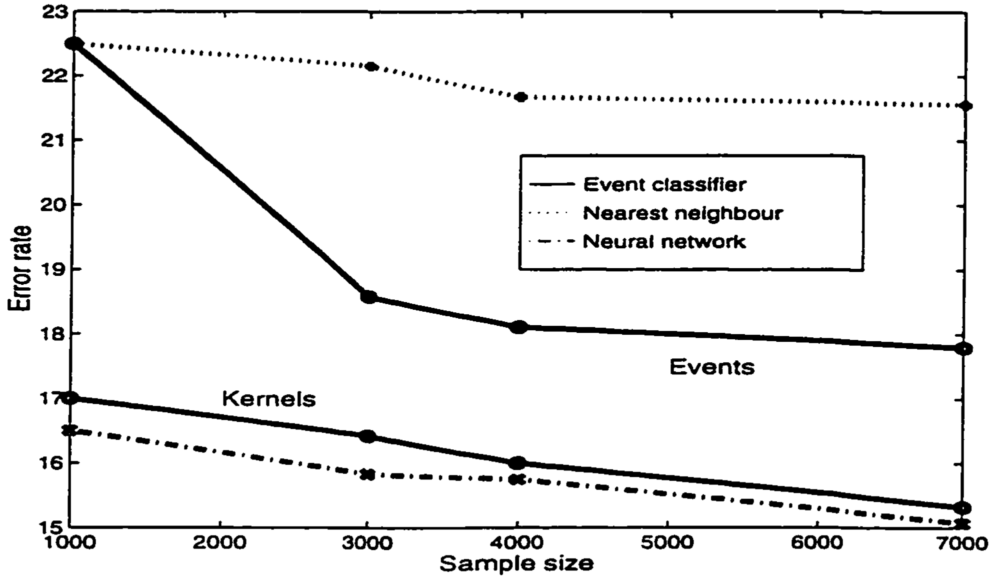


Figure 6.34: Convergence to the Bayes error rate for an artificial data set

the event classifier is seen to be fairly robust to noise corruption and exhibits a higher asymptotic convergence rate than the nearest neighbour classifier.

6.1.7 High-dimensional discovery and synthesis

The final basic property to be illustrated is the high-dimensional algorithm's ability to unearth sub-dimensional groupings in the data. Consider a 20 dimensional artificial data set embedded with one 8-dimensional cluster and one trivariate cluster at the following locations,

$$\begin{aligned}(X_1, X_2, X_5, X_7, X_8, X_9, X_{17}, X_{18}) &= (-2,-2,5,5,-3,-3,-3,4) \quad (6.2)\\ (X_5, X_7, X_9) &= (-5,-5,5)\end{aligned}$$

Table 6.8: Projections with significant events

Projection	# events
X_1X_2	1
X_1X_5	1
X_1X_7	1
X_1X_8	1
X_1X_9	1
X_1X_{17}	1
X_1X_{18}	1
X_2X_5	1
X_2X_7	1
X_2X_8	1
X_2X_9	1
X_2X_{17}	1
X_5X_{18}	1
X_5X_7	1
X_5X_8	1
X_5X_9	1
X_5X_{17}	2
X_5X_{18}	1
X_7X_8	1
X_7X_9	1
X_7X_{17}	2
X_7X_{18}	1
X_8X_9	1
X_8X_{11}	1
X_8X_{17}	1
X_8X_{18}	1
X_9X_{13}	1
X_9X_{17}	1
X_9X_{18}	1
$X_{17}X_{18}$	1

Each cluster consists of 25 points normally distributed around the corresponding center, with identity covariance matrix in the respective dimensions. In addition, 125 points are generated in 20 dimensions according to a uniform distribution within $[-25, 25]$. On applying the high-dimensional discovery algorithm, 33 events were detected in the 2-dimensional projections. According to the variables defining the clusters, there are 28 unique pairwise combinations of the variables and hence we expect that number of projections to yield significant events. However, in Table 6.8, we see that some extraneous relationships have arisen, as 30 out of 190 possible projections sport significant events. Table 6.9 summarizes the number of significant events synthesized at each dimension. The event synthesis algorithm determines that no structure exists beyond 8 dimensions, as no significant events are found for $d > 8$. This is in agreement with the dimensionality of the embedded patterns. The single 8 dimensional event is shown in Figure 6.35. The true cluster center is plotted as an open circle on each axis. Clearly, the 8-dimensional grouping has

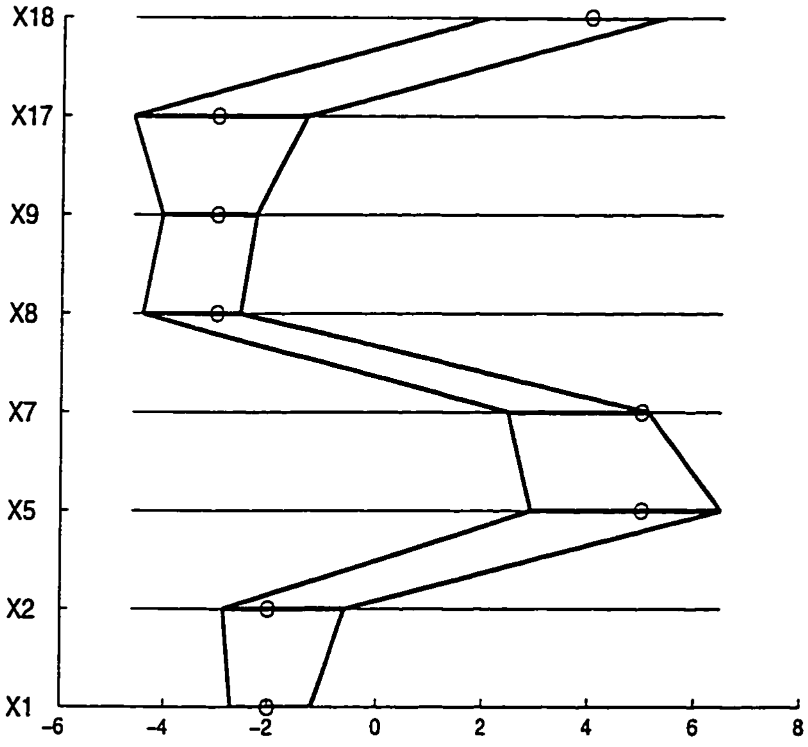


Figure 6.35: The 8-dimensional event revealed by discovery

Table 6.9: Number of events at each dimension d

d	# events
2	33
3	70
4	107
5	93
6	28
7	8
8	1
> 8	0

been detected. To confirm the detection of the trivariate pattern, we can apply the cluster test criterion to the 70 3-dimensional patterns. Doing so reveals 16 3-dimensional clusters. The centres in each dimension are plotted as open circles on the parallel axes of Figure 6.36. The solid line represents the location of the detected 8-dimensional cluster centre. The proximity of the trivariate cluster centres to those of the 8-dimensional cluster suggests that the former are actually projections of the 8-dimensional cluster onto $\Re^3$. The only distinct 3-dimensional cluster, delineated by the dashed line corresponds to the embedded 3-dimensional pattern. The detected groupings are quantitatively summarized in Table 6.10.

As a final note, clusters determined in dimensions 4 to 7 are all found to be projections of the 8-dimensional event. Hence, there are no distinct clusters in those dimensions. Further, the spurious events detected in the 2 dimensional projections do not propagate beyond 5 dimensions. In dimensions of 5 or less, synthesis involving these spurious events do not give rise to clusters and thus no false patterns are detected.

Table 6.10: Quantitative specification of discovered events

Feature low value
high value

	X_1	X_2	X_5	X_7	X_8	X_9	X_{17}	X_{18}	r_j	n_j
8-dimensional	-2.7	-2.8	2.9	2.49	-4.4	-4.1	-4.6	2.0	3.6	12
event	-1.2	-0.6	6.5	5.1	-2.6	-2.3	-1.3	5.4		
3-dimensional	-	-	-5.9	-8.6	-	4.3	-	-	3.5	12
event	-	-	0	-2.6	-	5.7	-	-		

(r_j = z-statistic value, n_j = observed frequency)

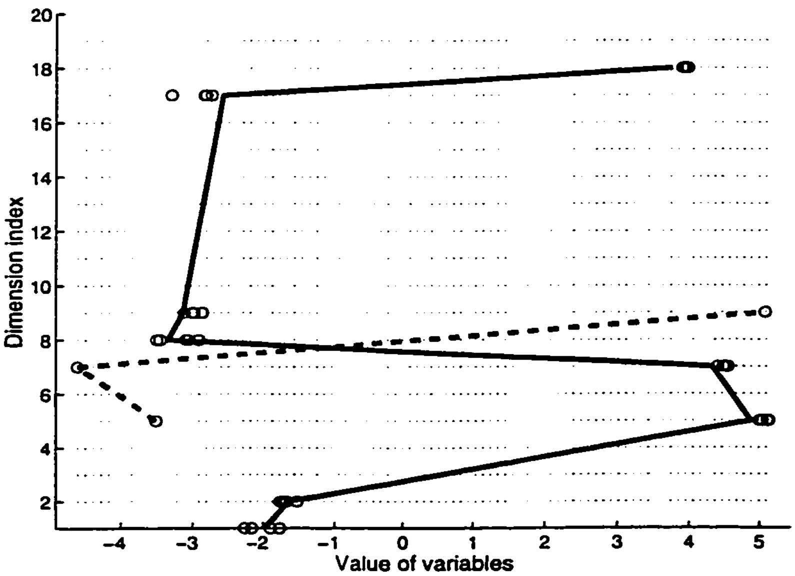


Figure 6.36: Discovery of the trivariate group

6.2 Case studies

Having illustrated a number of basic properties of event level pattern discovery, I now proceed to present the analyses of a few real world data sets.

6.2.1 Exploratory analysis and interpretation : Thyroid disease diagnosis

The first case study illustrates the ability of event level pattern discovery to elucidate the most distinguishing characteristics of a multiclass, multivariate data set. We visit the fairly well-understood area of thyroid gland disease to allow for easy verification of the discovered patterns. The data is taken from Coomans [33] and consists of 3 categories, hypothyroidism (deficiency in thyroid hormones), hyperthyroidism (excess of thyroid hormones) and euthyroidism (normal thyroid function). The features are 5 continuous clinical measurements summarized in Table 6.11. The data for each class is displayed using a parallel axes plot [122] in Figures 6.37 and 6.38. Undoubtedly, it is difficult to extract much information from these plots.

Pattern discovery with the concentration objective was applied to each category. The most significant events are shown in Figure 6.39 and the upper portion is magnified in Figure 6.40. These plots lend to rapid verification with domain theory. Consider the following observations.

1. The features T4 and T3RIA represent the 2 types of hormones produced by the thyroid gland. As expected, the events suggest that the hypothyroid subjects exhibit a reduced level of these hormones while the hyperthyroid

Table 6.11: Thyroid data features

Feature #	Measurement	Units	Symbol
1	T3-resin uptake test	%	RT3U
2	Total serum thyroxin	$\mu g/dL$	T4
3	Total serum triiodothyronine	$\mu g/L$	T3RIA
4	Basal thyroid-stimulating hormone	$\mu IU/mL$	TSH
5	Maximal absolute difference of the TSH value after injection of 200μg of thyrotropin-releasing hormone (TRH) as compared to the basal value.	$\mu IU/mL$	dTSH

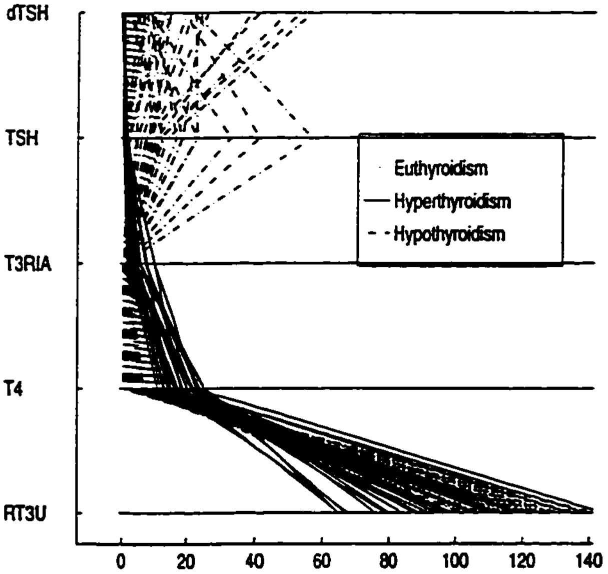


Figure 6.37: Parallel axes plot of thyroid data

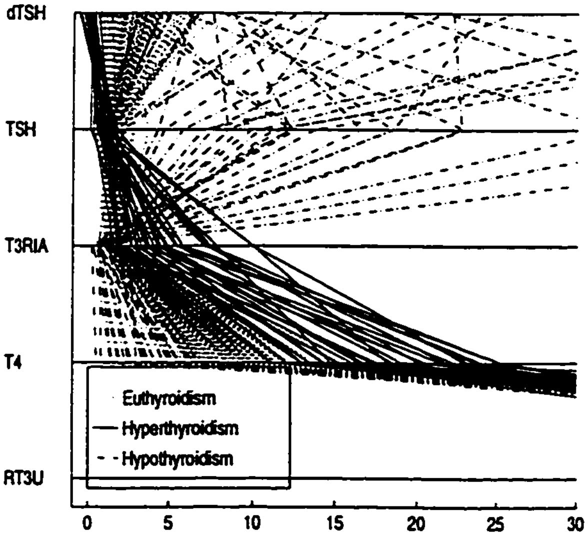


Figure 6.38: Zoom-in on features 2-5

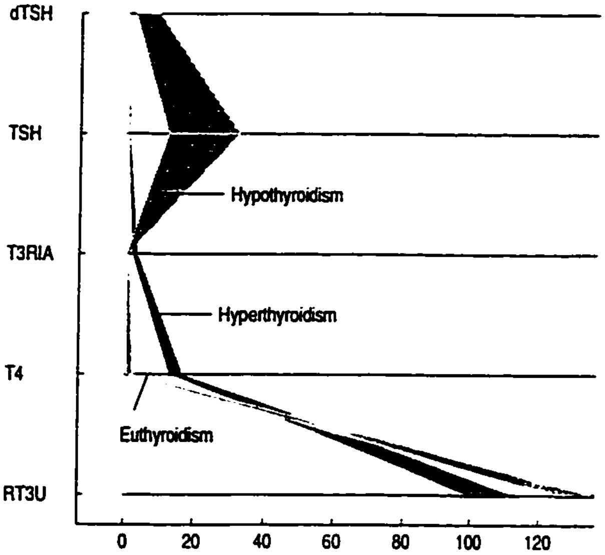


Figure 6.39: Most significant events

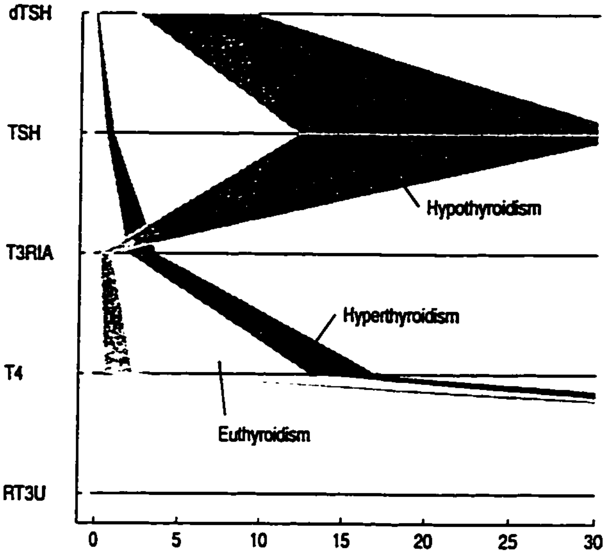


Figure 6.40: Magnified significant events

subjects are characterized by abnormally high hormone concentrations. The euthyroid hormone levels fall between those of the two disease states.

2. The events also suggest that TSH is substantially higher in the hypothyroid case. Again, this is in agreement with physiological principles. In secondary hypothyroidism, the low-functioning thyroid gland produces very few thyroid hormones (T3 and T4). The negative feedback that these hormones normally provide to regulate TSH production is removed and TSH levels soar. The opposite situation occurs in the hyperthyroid case and is also elucidated by the significant events.

3. The observation that

   dTSH (hypothyroid) > dTSH (euthyroid) > dTSH (hyperthyroid)

   can be explained with similar reasoning. Referring back to Table 6.11, we note

Table 6.12: Event statistics for the most significant events

Category	Residual	% Observations in support
Euthyroidism	2.7	5.3
Hyperthyroidism	2.3	14.3
Hypothyroidism	2.6	20.0

Critical value of the statistic is 1.96 at a 5% significance level.

that dTSH (the change in TSH) is a measurement obtained after injection of TRH, a hormone which stimulates the production of TSH. In the hyperthyroid case, there is so much inhibition from the excess thyroid hormones (observation 1), that additional TRH has little effect on TSH production. The opposite is true for hypothyroidism. In the the normal functioning thyroid, injection of TRH increases the production of T3, but the intact negative feedback maintains TSH at normal levels.

Conventional exploratory data analysis such as parallel axes plots, may qualitatively hint at the separable structure in the T4, T3RIA and TSH features. However, apart from quantifying this information, pattern discovery also sheds light on the previously obscured structure of the dTSH feature. The quantitative information communicated by the events are in terms of event boundaries, statistic value and % of observations supporting each event. Table 6.12 summarizes some of these findings.

The first case study has illustrated that by providing interpretable patterns, event-based pattern discovery can clearly elucidate the most distinguishing characteristics of a multiclass, multivariate data set.

Table 6.13: Attributes of chemical-overt diabetes study

Feature #	Measurement	Symbol
1	Relative weight	RW
2	Fasting plasma glucose	FPG
3	Glucose area (glucose intolerance)	GA
4	Insulin area (insulin response to oral glucose)	IA
5	Steady state plasma glucose (insulin resistance)	SSPG

6.2.2 Local dependencies and locally significant features: Chemical and overt diabetes

The second example illustrates the abilities to detect interesting local dependencies and to identify locally important features. The latter differs from traditional feature selection in that individual regions of the sample space may be characterized by different sets of features. Here, I will examine a 5 dimensional diabetes data set, collected to study the relationships among 3 clinical classifications of 145 non-obese adult subjects: (1) overt nonketotic diabetic (33 subjects), (2) chemical subclinical diabetic (36 subjects) and (3) healthy (76 subjects). The attributes are listed in Table 6.13. In the original analysis of Reaven and Miller [99], the last 3 features were determined as the "primary" variables which captured the natural groupings in the data. Subsequently, Symons [113] studied the same 3 features in evaluating various clustering criteria, noting that the heterogeneous and nonellipsoid clusters are difficult to detect. The present analysis will take an unbiased view towards the 5 features and search for dependencies and locally informative features.

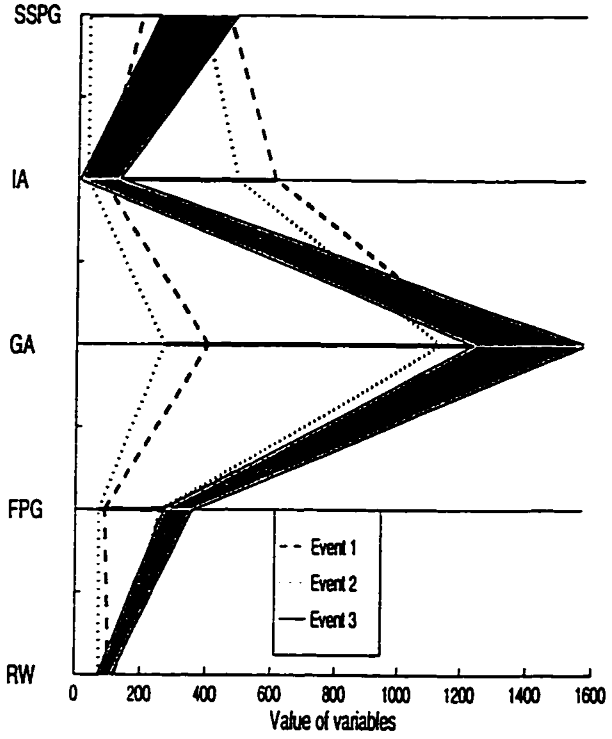


Figure 6.41: Events from dependency discovery: chemical-overt diabetes
Note: In this plot, the values of the RW feature have been scaled by a factor of 100 for visualization purposes.

Local dependencies

Considering the entire data set in the full dimensionality, 5 separate discoveries with the dependency objective consistently revealed 3 types of significant events, as exemplified in Figure 6.41. To gain a better understanding, Table 6.14 enumerates the number of examples from each class falling within individual events. We note that event 3 describes a dependence among the 5 attributes, exclusively for a subset of the overt subjects. This event is shaded in Figure 6.41 and reveals a dramatically unique structure. Referring back to Table 6.14, event 1 indicates that there is also

Table 6.14: Class memberships of dependency events: chemical-overt diabetes

	Number of examples in each class		
	Overt	Chemical	Healthy
Event 1	12	18	1
Event 2	11	16	72
Event 3	10	0	0

a dependency which applies to both the overt and chemical diabetics, i.e. the disease afflicted subjects. Again from Table 6.14, we see that event 2 contains mostly healthy subjects but has a substantial disease component as well. This suggests that there is a dependency most characteristic of the healthy subjects, but also shared by some of the afflicted patients. To explore these local dependencies further, one might employ standard fitting procedures, such as regression or one of the many robust connectionist paradigms. For simplicity, I will only examine the correlation matrix for the subsets of data falling within each event.

Using Hinton diagrams [85], to display the correlation matrices, we obtain the plots of Figures 6.42, 6.43 and 6.44. These correlation matrices suggest that different dependencies are indeed captured by the events. Some interesting observations are noted.

1. In Event 1, the dominant negative correlations all involve the IA variable: IA-FPG,IA-GA and IA-SSPG. These dependencies can be explained by the basic physiological principle that decreases in insulin response (IA) causes the levels of glucose to rise (SSPG,FPG).

2. In contrast, Event 2 almost exclusively consists of positive correlations, with

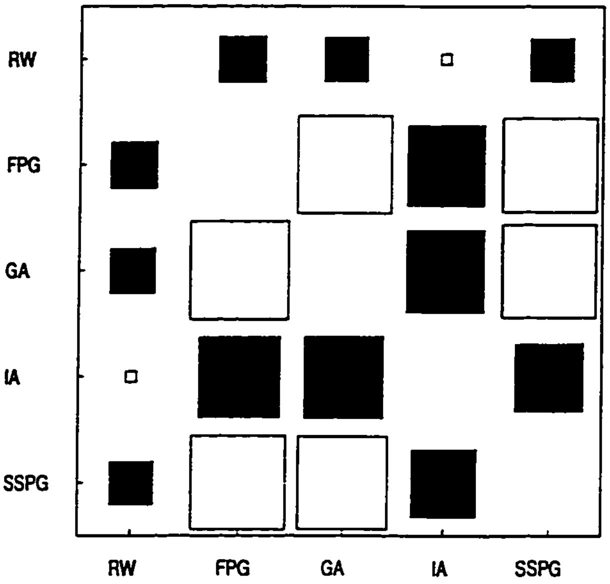


Figure 6.42: Correlation matrix-Event 1

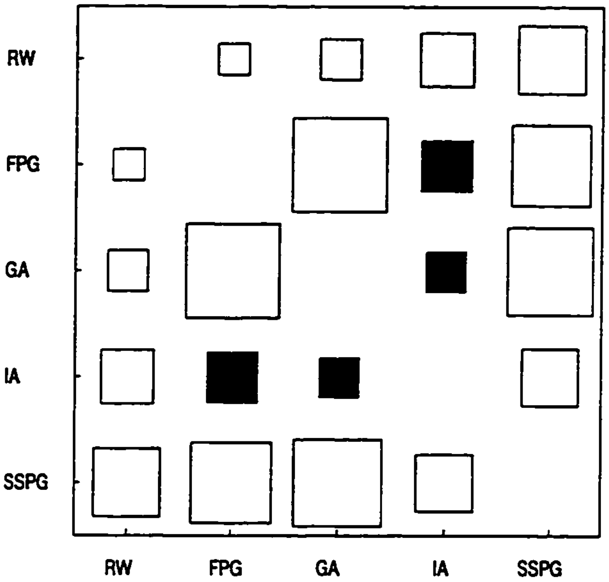


Figure 6.43: Correlation matrix-Event 2

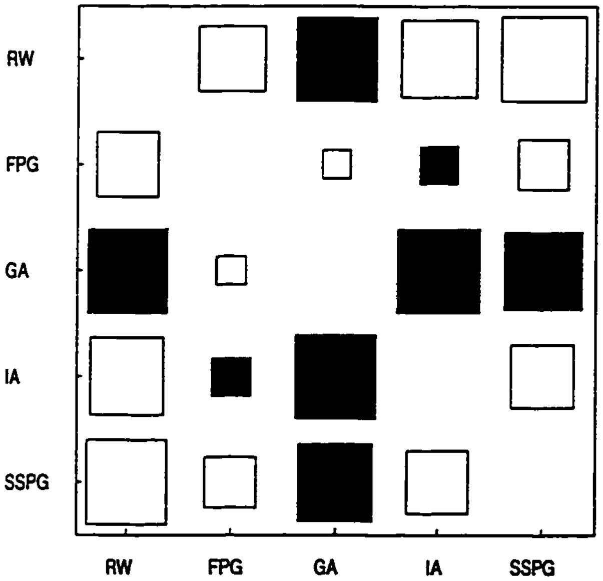


Figure 6.44: Correlation matrix-Event 3

In the above diagrams, a shaded square represents a negative correlation while an empty square represents a positive correlation. The size of the square reflects the magnitude of the correlation. Note that the main diagonal entries are all unity and are omitted for clarity.

the same dominant ones as in Event 1. In addition, all relationships with SSPG and RW are now positive correlations. With healthy subjects, a rise in glucose levels reflects an increase in weight. This also applies for subjects in the advanced stages of the disease. However, in the earliest stages of the disease, elevated glucose levels cause initial weight loss. This physiological phenomenon accounts for the inclusion of predominantly healthy subjects in Event 2, along with a fraction of diabetics, i.e. those in the advanced stages of the disease.

3. Event 3 is the most different. The dominant negative correlations all involve GA, i.e. GA-RW, GA-IA and GA-SSPG. These relationships all agree with theory. Increases in glucose intolerance (GA) can cause weight loss (decrease in RW) and is reflective of damage to the pancreas (decrease in IA) or damage to the insulin cells (decrease SSPG).

In these observations, I have interpreted the dependencies in terms of the usual direction in which the variables change. For example, glucose intolerance (GA) is a negative characteristic and it usually only gets worse (increase) rather than better (decrease). With this in mind, it is particularly interesting to note that only Events 1 and 3 contain dependencies which suggest illness. Incidentally, these two events only contain diabetic subjects, with the exception of one healthy person. By this preliminary analysis, we can already see that the events have successfully captured interesting and significant local dependencies.

This first experiment with the chemical-overt diabetes data illustrates that pattern discovery can immediately pinpoint potentially unique, local relationships in

Table 6.15: Chemical-overt diabetes: Number of events at each dimension d

d	Overt	Chemical	Healthy
2	19	21	31
3	5	2	7
4	5	1	1
5	1	1	1

Note: For this discovery experiment, the significance level was relaxed to $\alpha = 0.15$ to allow for greater coverage by the crude process of partitioning. Event synthesis was conducted with $\alpha = 0.01$.

the data.

Locally significant features

To investigate locally significant combinations of features, we can invoke the event synthesis algorithm, using the concentration objective. The aim is to seek out attribute values which have significant joint occurrences. The results of applying discovery and synthesis to each class are tabulated in the Table 6.15. Computing class memberships for each event and examining the frequency of support, we find that overt and chemical classes are strongly represented by the features FPG and GA, while the healthy subjects cluster most prominently in the GA and IA dimensions. Higher dimensional patterns do not offer a better combination of class homogeneity and statistical significance. To obtain a more refined representation in these selected dimensions, optimization discovery was performed for the chemical and healthy classes. The partitioning result for the overt class was deemed acceptable. The final 4 events are sufficient to categorize the subjects with an error rate of only 8.3%, i.e. 133/145 subjects correctly categorized. The corresponding

Table 6.16: Rules for chemical-overt diabetes data

OVERT - Features 2 and 3	CHEMICAL - Features 2 and 3
If $120 \leq$ Fasting plasma glucose ≤ 203 and $538 \leq$ Glucose area ≤ 972 then the subject is OVERT diabetic.	If $88 \leq$ Fasting plasma glucose ≤ 114 and $413 \leq$ Glucose area ≤ 568 then the subject is CHEMICAL diabetic.
OVERT - Features 2 and 3	HEALTHY - Features 3 and 4
If $213 \leq$ Fasting plasma glucose ≤ 353 and $1001 \leq$ Glucose area ≤ 1578 then the subject is OVERT diabetic.	If $289 \leq$ Glucose area ≤ 426 and $73 \leq$ Insulin area ≤ 292 then the subject is HEALTHY.

classification rules are listed in Table 6.16.

Pattern discovery has revealed that each group of subjects can be adequately characterized by a distinct group of only 2 features. In particular, the SSPG feature identified by Reaven and Miller is not needed in the characterization of any of the natural groupings. Instead, pattern discovery found the FPG factor to be of primary importance in delineating the diseased groups. Incidentally, FPG (fasting plasma glucose) is known to be a key clinical indicator of the diabetic state [66]. The accuracy of the categorization is close to the 5.5% error rate obtained with a 5-10-1 feedforward neural network, using all 5 features. However, pattern discovery offers greater interpretability through simple rules and via statistical testing, avoids the use of the full 5 features.

The second experiment with the chemical-overt diabetes data shows that event-based pattern discovery can detect subsets of features that are significant only in local subspaces, a generalization of traditional feature selection.

Table 6.17: Attributes in PIMA diabetes study

Feature #	Measurement	Symbol
1	Number of pregnancies	NP
2	Plasma glucose concentration (glucose intolerance test)	PG
3	Diastolic blood pressure (mmHg)	BP
4	Triceps skin fold thickness (mm)	SK
5	2-hour serum insulin (μU/mL)	SI
6	Body mass index (kg/m^2)	BMI
7	Pedigree function	PF
8	Age (years)	AGE

6.2.3 Pattern discovery and classification: Diabetes diagnosis in PIMA Indians

In this case study, pattern discovery offers insight into the complications of a difficult classification problem. The data is a collection of 8 clinical measurements (Table 6.17) taken from 392 female subjects[2] in the Pima Indian population [15, 73]. The subjects are categorized into 2 groups, diabetic and non-diabetic. This data set has been used in the evaluation of various advanced classification algorithms [109, 101]. A common result of these analyses is a fairly large error rate, usually around 25%. The aim of the present investigation is to seek an explanation for the prevalently poor classification rate by uncovering patterns inherent in the data set.

To seek significant joint occurrences with the concentration objective, the high-

[2] The original data set has 768 subjects, but 376 have incomplete measurements. As in many previous analyses, we study only the complete examples here.

Table 6.18: PIMA diabetes: events in each dimension d

d	Healthy	Diabetic
2	137	74
3	56	15
4	70	40
5	56	41
6	28	23
7	8	5
8	1	0

dimensional discovery algorithm was applied to each class of subjects (133 healthy, 67 diseased).[3] Table 6.18 contains the results of discovery and synthesis. Focusing on the highest dimensional events, we see that a number of obvious patterns have been identified. Figure 6.45 is a typical example. Based on this plot, the following predominant characteristics are observed.

1. PG (diabetic)$\gg$ PG (healthy)

   It is important to note that the normal range for this feature is PG$<$ 140, whereas 140 $<$ PG $<$ 200 suggests impaired glucose tolerance, but is not considered to be an indication of diabetes mellitus (Type II) [66]. Nonetheless, for the events shown, healthy subjects seem to fall in the range 74 $<$ PG $<$ 105 while for the disease events, 105 $<$ PG $<$ 181. It appears then, that this could serve as an important discriminatory feature.

2. SI (diabetic) $\gg$ SI (healthy)

   This second observation is consistent with the insulin resistance syndrome [63],

[3] As in other analyses of this data set, 192 examples are reserved for testing.

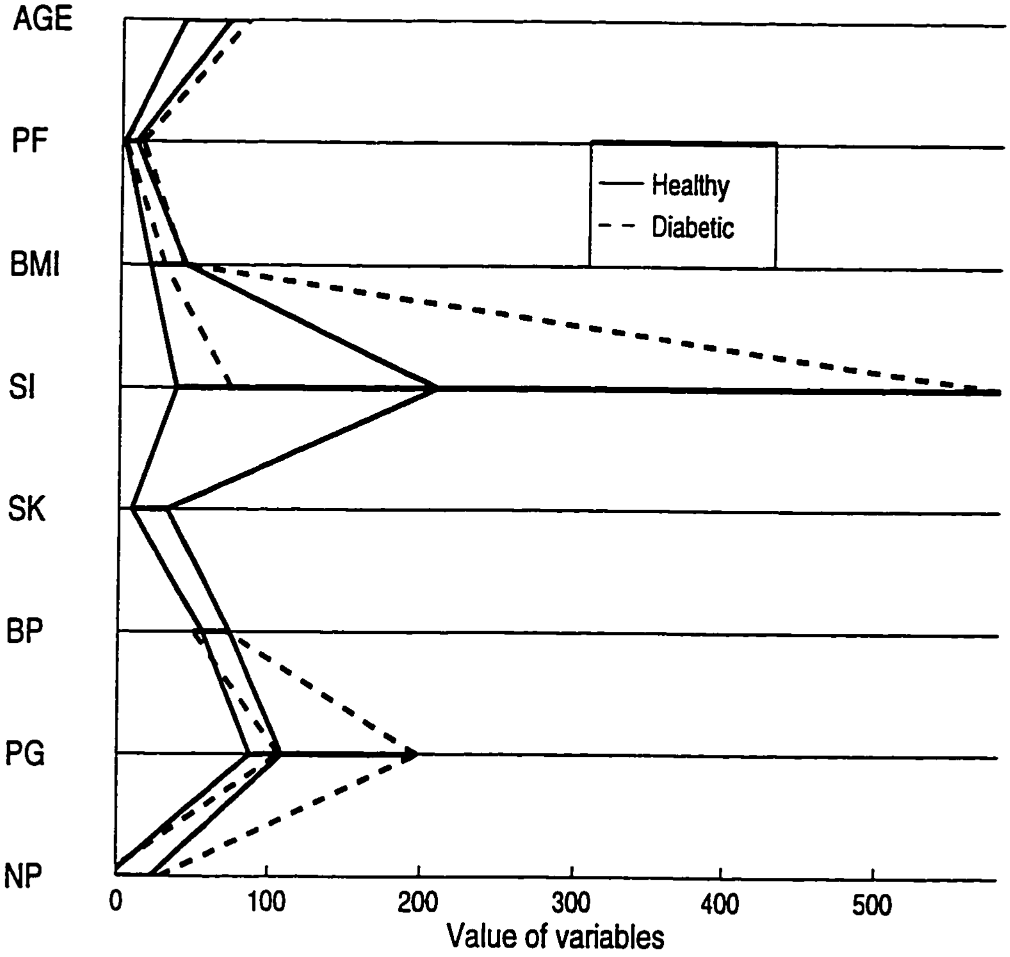
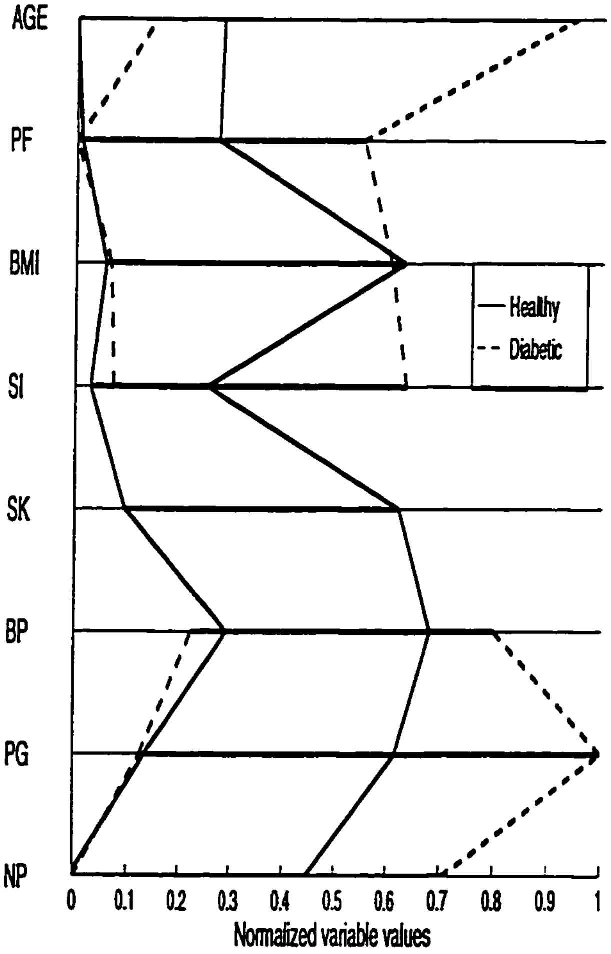


Figure 6.45: Example of obvious patterns in PIMA data
Note: The event boundaries for the NP,PF and AGE attributes have been magnified by factors of 10,10 and 2 respectively, for ease of visualization. The diabetic event is actually only 7 dimensional as the SK attribute did not appear in the pattern.

a metabolic abnormality known to occur in Type II diabetic subjects. Defective insulin cells fail to act on their targets and consequently glucose levels remain elevated, stimulating the production of yet more insulin. In Figure 6.45, although the diabetic range for SI is definitely wider than the corresponding healthy range, it also considerably overlaps the latter.

Although theory suggests that blood pressure and age would also be higher for Type II diabetics [62], this was not clearly evident from the discovered events. In fact, other high-dimensional events seem to suggest that the healthy and diabetic subject are indistinguishably similar.

Constructing a kernel-based classifier using the two events shown in Figure 6.45, an error of 47% is obtained. To seek improvement in classification, we explore the data a little further. Taking the dimensions selected from event synthesis (8 for the healthy group and 7 for the diseased group), optimization discovery is performed on each category of data. After one level of recursion, an interesting pattern emerges. As seen in Figure 6.46, the AGE variable seems to show reasonable separation. Following up on this hypothesis, it is verified that indeed, less than one-third of the diabetic patients are younger than 27 years of age. If we refine the event boundary for the healthy group to reflect this discovery, the error falls to 23%, using just one kernel for each class. This is slightly better than the results of Smith [109] but comparable to that of Ripley [101]. The immediate question is how close is this error to the best achievable error with this data set. Assessment of this error rate, allows us to dissociate limitations of the data from those of the classifier. For a given problem, without knowledge of the underlying class densities, this error rate



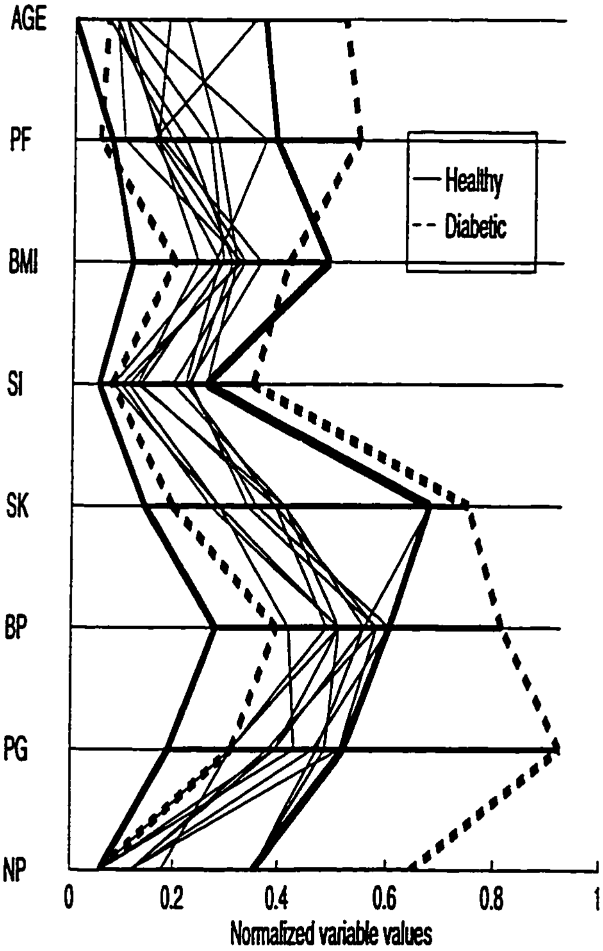


Figure 6.46: Result of recursive discovery

Figure 6.47: Overlap estimation

The variable values have been normalized to enhance visualization.

Table 6.19: Estimates of the achievable error rate

Number of points covered				Estimated
(A)	(B)	(C)	(D)	Error
Healthy	Diabetic	Overlap	Net	Rate
			(A+B-C)	C/D
273	164	114	323	0.353
218	93	50	261	0.192

is difficult to determine. With events, we can obtain a rough approximation.

The unavoidable errors are observations from one class which are completely indistinguishable from those of the other class, based on the provided attributes. We can identify such observations as those which simultaneously fall into significant events for both healthy and diabetic data sets. Performing pattern discovery by optimization with the concentration objective, on the entire data set, in the full dimensionality, we find one significant event for each class. Based on these two events, the estimated error rate is 35.3 %. With one level of recursive discovery, the estimate falls to 19.2 %. Further recursion did not reveal any more significant events. The findings are summarized in Table 6.19. The first estimate is considered pessimistic since without recursion, the concentration objective may not be sufficiently maximized. The second estimate is more indicative of the true overlap and may be slightly conservative since observations outside the highest concentration regions are ignored. Figure 6.47 exemplifies some of the confounding observations on a parallel axes plot. The events and the observations have been normalized to facilitate visualization. Note that the confusing points, indicated by thin lines, fall within the intersection of the two events. As shown in Table 6.20, the healthy and

Table 6.20: Example of confounding observations

NP	PG	BP	SK	SI	BMI	PF	AGE	Diagnosis
2	122	76	27	200	35.9	0.48	26	Healthy
2	129	74	26	205	33.2	0.59	25	Diabetic

diabetic values for most of these observations are practically indistinguishable.

The large number of confounding observations is likely due to the incomplete abstraction of the expert's arsenal of diagnostic tools. In clinical diagnosis, additional information such as symptoms (thirst, polyuria, impairment of visual acuity, unexplained weight loss) along with other lab tests are taken into consideration [66]. Furthermore, some of the healthy subjects might have been afflicted with impaired glucose tolerance [62], a 'risk class' which had not yet been identified at the time of this particular study. This would explain their similarity to the diabetics based on the 8 measurements alone.

In this example, we see that pattern discovery can offer insight into the amount and type of confusion in a data set. By interpreting events, we can pinpoint discriminatory features. Both pieces of information are valuable to classifier design and may guide the collection of additional data.

6.2.4 Time-dependent discovery: EMG control signals

In this example, I will explain the application of time-varying pattern discovery to detect nonstationary changes in the data. The data, shown in Figure 6.48, consists of the low-pass filtered electromyographic (EMG) signals measured from two forearm muscle sites of a below-elbow amputee subject. The data was collected

Table 6.21: EMG recording conditions

Time	Residual limb condition
t_0	At rest
t_1	Fatigue
t_2	Lifting 500 g load
t_3	Lifting 1 kg load
t_4	Lifting 1.5 kg load
t_5	In motion, frontal plane
t_6	In motion, sagittal plane
t_7	Maximum voluntary contraction

over 200 ms time intervals, under different conditions of the residual limb, as summarized in Table 6.21. Each time slice contains the recordings of five consecutive muscle contractions. It is known that the EMG signal is nonstationary and zero mean [92]. However, it has been proposed that the initial 200 ms transient can be considered locally stationary [69].

With pattern discovery, we investigate the stationarity of the signal by detecting significant event shifts. Applying the time-dependent discovery algorithm, over the eight 200ms intervals for wrist flexion, we obtain the plots in Figure 6.49. The concentration objective was invoked and the resulting events have been smoothed by the kernel method and the contours of the pdf at each time are plotted. According to a 5% significance level, significant event shifts are detected at almost every transition. The only exception is the change from t_5 and t_6, during which the residual limb remains in motion with only a change in orientation. From t_0 to t_1, the change from rest to fatigue state is minor. In conditions of fatigue, an overall increase in EMG amplitude is expected [11], so the available data may not have be an accurate characterization of fatigue. The most dramatic changes in the pdf occur during

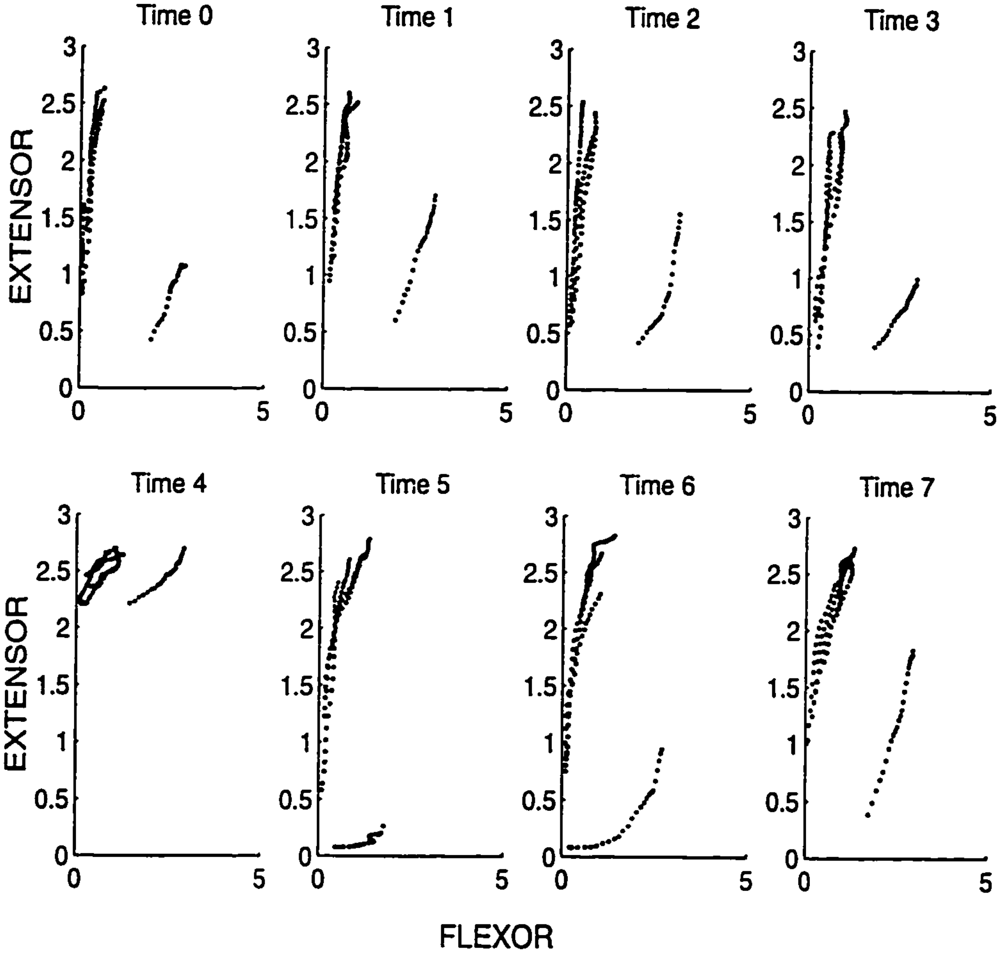


Figure 6.48: Wrist extensions measured under different residual limb conditions

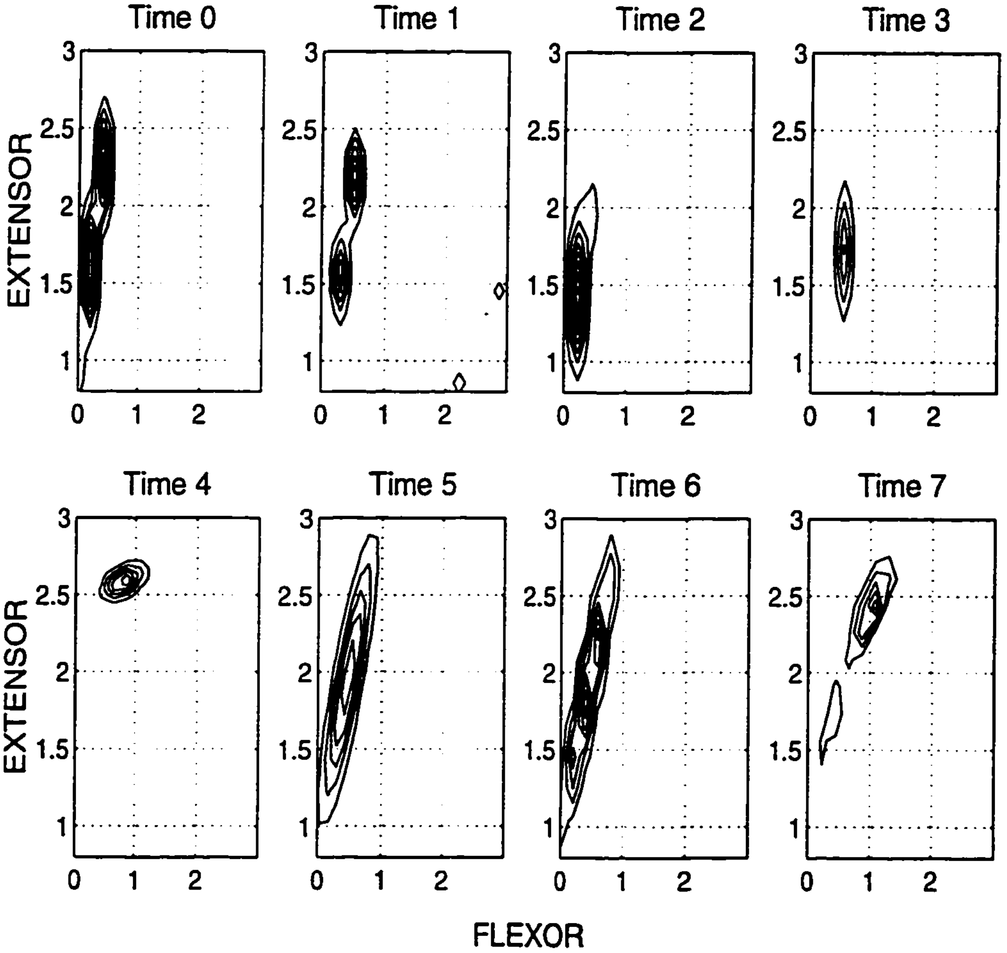


Figure 6.49: Time-evolution of pdf for wrist extension
Note that to emphasize details of the contours, the vertical and horizontal scales cover smaller ranges than in Figure 6.48.

Table 6.22: Example of significant event shift: t_1 to t_2

	Events			
Time	E_1	E_2	E_{OUT}	Totals
t_1	90	23	12	125
t_2	67 $\boxed{4.4}$	16 $\boxed{2.3}$	42 $\boxed{-8.6}$	125
Totals	157	39	54	250

times of loading, i.e. t_2 to t_4. This is expected as during loading, antagonist and synergist muscles actively contribute to the measured signal, altering its properties [11]. Note that the outlying signals in Figure 6.49 have little contribution to the pdf.

In Table 6.22, the significant event shift from t_1 to t_2 is detailed. The E_{OUT} symbol denotes the background event. The table entries are event frequencies for each time, while the boxed numbers in the t_2 row are the test statistic values, measuring the change from t_1 to t_2. Based on a 5% significance level, a significant event shift is detected. The counts in E_1 and E_2 have decreased significantly, while the background total has increased. Subsequently, discovery is applied to the data at t_2. From the pattern discovery analysis, we can conclude that the signal can only be considered stationary during times, t_5 and t_6, i.e. only when the limb is in motion in the sagittal and frontal planes.

In this example, it would be difficult to define threshold probabilities for detecting changes. Further, global retraining at every instant of time would be redundant especially in the intervals $[t_1, t_2]$ and $[t_5, t_6]$. Event-based pattern discovery on the other hand, provides an objective means of gauging the significance of local variations in the pattern. In so doing, periods of stationary behaviour can be identified and unnecessary retraining is avoided.

6.2.5 High-dimensional discovery and prediction: Housing data

The fifth example illustrates the use of pattern discovery to determine ranges over which dependencies are strongest, and hence candidates for model fitting. The data set is a collection of 13 factors as listed in Table 6.23 which are believed to influence housing prices in various towns neighbouring Boston [59]. There are 506 data points in total, with price being the dependent variable.

The number of significant events determined by the event synthesis algorithm, using the dependency objective, are listed in Table 6.24. An extra column enumerates the patterns related to housing price, the dependent variable of interest. No events were discovered beyond 12 dimensions. By studying the 11 and 12 dimensional patterns, it becomes evident that housing price exhibits at least two possible types of dependencies, one for lower priced homes and another for higher priced homes. In Table 6.25, four 11-dimensional patterns are reported. Of the 21 11-dimensional patterns, 11 predicted high housing prices, 8 dealt with low housing prices and 3 covered both ranges. The difference between the high and low price patterns is striking. Consider the following observations.

1. The price of expensive homes are influenced by extremely low crime rates and low pollution levels (nitric oxide). The reverse applies for cheaper homes.

Table 6.23: Attributes for housing price data

Feature #	Measurement	Symbol
1	Per capita crime rate by town	CRIM
2	Proportion of residential land zoned for lots > $25,000 ft^2$	ZN
3	Proportion of non-retail acres/town	INDUS
4	Charles river variable	CHAS
5	Nitric oxides concentration (ppm/10)	NOX
6	Average # rooms/dwelling	RM
7	Proportion of homes built prior to 1940	AGE
8	Weighted distances to 5 employment centres	DIS
9	Index of accessibility to radial highways	RAD
10	Property tax-rate/$10,000	TAX
11	Pupil-teacher ratio	PTRATIO
12	Ethnic Composition Index	ECI
13	% lower status of population	LSTAT
14	Median value of owner-occupied homes in $1000's	MEDV

Table 6.24: Number of events at each dimension d

d	# significant events detected	# significant events related to median price
2	139	11
3	164	35
4	519	161
5	998	410
6	1294	655
7	1195	711
8	797	540
9	377	285
10	120	100
11	23	21
12	2	2
> 12	0	0

2. Expensive homes are generally more spacious as a larger proportion are zoned for large lots (30-90%). In fact, zoning is not even a determining factor for the price of cheaper homes as it never appears in the low price patterns.

3. High-priced homes are noticeably younger than their low-priced counterparts, ranging from 10 to 50 years old, while cheaper homes typically boast an excess of 70 years.

4. The neighbourhoods are also substantially different. Frugal homes are close to highways (radial highway accessibility) and within walking distance to the city core. In contrast, thrifty homes enjoy quiet comfort away from highways and the bustle of city centres. Neighborhoods of low-priced homes are zoned for up to 20% industrial use, while higher priced neighborhoods are zoned a maximum of 6%.

Table 6.25: Some 11-dimensional patterns detected in the housing data

High-priced homes

If $0.01 \le$ CRIME RATE ≤ 0.06
and $31.92 \le$ ZONING ≤ 98.75
and $-0.24 \le$ INDUSTRIAL ≤ 4.65
and $0.39 \le$ NITRIC OXIDES ≤ 0.47
and $7 \le$ NUMBER OF ROOMS ≤ 8
and $9.90 \le$ AGE ≤ 49.30
and $4.02 \le$ DISTANCE ≤ 12.13
and $201.88 \le$ TAX ≤ 363.01
and $12.30 \le$ PUPIL-TEACHER RATIO ≤ 19.89
and $1.98 \le$ PERCENT LOWER STATUS ≤ 8.23
then $22.00 \le$ MEDIAN PRICE ≤ 50.00
with statistic value 4.850869,
and support 23

If $0.01 \le$ CRIME RATE ≤ 0.06
and $31.92 \le$ ZONING ≤ 95.27
and $-0.40 \le$ INDUSTRIAL ≤ 6.27
and $0.39 \le$ NITRIC OXIDES ≤ 0.48
and $6 \le$ NUMBER OF ROOMS ≤ 8
and $9.90 \le$ AGE ≤ 56.40
and $4.08 \le$ DISTANCE ≤ 8.54
and $0.67 \le$ RADIAL HIGHWAYS ≤ 6.33
and $217.76 \le$ TAX ≤ 374.52
and $11.91 \le$ PUPIL-TEACHER RATIO ≤ 18.83
then $22.30 \le$ MEDIAN PRICE ≤ 50.00
with statistic value 5.164617,
and support 26

Low-priced homes

If $5.44 \le$ CRIME RATE ≤ 73.53
and $16.67 \le$ INDUSTRIAL ≤ 19.10
and $0.58 \le$ NITRIC OXIDES ≤ 0.74
and $4 \le$ NUMBER OF ROOMS ≤ 7
and $85.40 \le$ AGE ≤ 100.00
and $1.18 \le$ DISTANCE ≤ 2.36
and $22.59 \le$ RADIAL HIGHWAYS ≤ 25.23
and $648.68 \le$ TAX ≤ 691.72
and $7.68 \le$ ETHNIC COMPOSITION ≤ 372.92
and $15.02 \le$ PERCENT LOWER STATUS ≤ 36.98
then $7.00 \le$ MEDIAN PRICE ≤ 17.20
with statistic value 5.062703,
and support 23

If $4.67 \le$ CRIME RATE ≤ 73.53
and $16.67 \le$ INDUSTRIAL ≤ 19.10
and $0.58 \le$ NITRIC OXIDES ≤ 0.74
and $4 \le$ NUMBER OF ROOMS ≤ 7
and $85.40 \le$ AGE ≤ 100.00
and $1.18 \le$ DISTANCE ≤ 2.58
and $648.68 \le$ TAX ≤ 691.72
and $19.48 \le$ PUPIL-TEACHER RATIO ≤ 20.92
and $7.68 \le$ ETHNIC COMPOSITION ≤ 372.92
and $15.02 \le$ PERCENT LOWER STATUS ≤ 36.98
then $7.00 \le$ MEDIAN PRICE ≤ 17.20
with statistic value 5.266492,
and support 25

Note: The statistic value is the residual statistic of the event. The support is the observed frequency.

5. High-priced homes are associated with lower student to teacher ratios, suggesting a higher quality of education.

6. The tax rate for expensive homes is substantially lower. This may seem counterintuitive at first, but the rate is only part of the equation. The higher value of the expensive homes would result in a larger total levy.

7. Social trends are also characteristically different. The patterns reveal a higher percentage of lower "status" home owners among cheaper homes. Ethnic composition is not even a consideration for the high-priced dwellings but can range to very high purities for more humble abodes.

A commonality between the 2 housing prices is that neither seems to depend on the Charles River variable, which is related to the home's proximity to the river. The same differences between high and low priced dwellings propagate to the two 12-dimensional patterns. We see that pattern discovery immediately reveals a wealth of information from a voluminous high dimensional data set, which would otherwise be difficult to interpret.

The 12 dimensions selected by discovery have sufficient predictive power, as the 12-dimension linear fits are comparable to that obtained with the full dimensionality. Other combinations of 12 dimensions yield poorer fits and further reduction in the number of variables also gives inferior prediction. These findings are reported in Table 6.26. The last two rows show that prediction improves when we use local models defined over the detected events. However, random selection of the same number of models, each accounting for approximately the same number of samples, yields equally good results. It seems that this data set is governed overwhelmingly

Table 6.26: Verifying the selected dimensions

Dimensionality	r^2	Sum-of-squared error
14	0.798	3.14
12 (low price)	0.799	3.14
12 (high price)	0.793	3.22
12 (other combinations)	0.75	3.82
11	0.77	3.57
12 (event-based)	0.88	2.8
14 (random)	0.89	2.6

The regressions were performed with the transformations listed in [14, p.231].

Table 6.27: Average values of 'outlying' high price homes

Crime rate	Zoning	Industrial	Nitric oxides	# rooms	Age	Distance	Tax	Pupil/ Teacher	% lower status	Median Price
3.5	0.54	19.2	0.6	7	81	2.3	460	17	4.3	50

by a common linear relationship and that there is little merit in choosing the local, event-based models.

There is however, an interesting observation about the sample points that are not covered by events when pattern discovery by optimization, with the dependency objective is performed in the 12 "high-price" dimensions. A group of 5 high-priced homes exhibited a dramatically different trend from the other expensive homes. Their average attribute values are recorded in Table 6.27. A quick comparison to the patterns in Table 6.25, confirms the remarkable deviation. These expensive homes are *close* to the city, explaining the larger industrial zoning, higher pollution, smaller lots, proximity to highways and higher student to pupil ratio. They are also very old and a fairly high tax rate is imposed. Another intriguing observation is that the prediction for these homes is quite poor, when using both global and local models. Their prices are grossly underestimated. For example, using a global model, the average absolute prediction error over the 5 homes is 18.2 whereas for the remaining homes the average is only 2.8, nearly a seven fold discrepancy in prediction error. It can be reasoned that since these homes resemble the low priced dwellings, the model would underestimate their value. Perhaps, an alternative model should be dedicated to these high priced homes.

In this experiment, although the knowledge of local dependencies did not enhance prediction, it did serve to verify that a single, global model is for the most part, satisfactory. However, discovery also identified local deviations from the overall dependency which may deserve special attention. Again, a comprehensive interpretation of the data set is immediately provided by the significant events, highlighting trends for low and high price homes.

6.2.6 Multivariate outliers: Virus data

In high dimensions, outliers are generally difficult to detect because they distort measures of location, scale and orientation [9]. Further, unlike the univariate case, outliers may arise as a result of systematic errors in a single dimension, a combination of dimensions or all dimensions. As a result, characterization of multivariate outliers is extremely challenging [51].

In this last case study, I exemplify the use of pattern discovery to detect outliers in high-dimensional, sparse data. The data set is composed of 18 measurements made on the protein coats of 38 *Tobamoviruses*, a type of rod-shaped virus which affects various crops [45]. Each measurement is the number of amino acid residues per molecule of coat protein. Due to the sparsity of this data set, outlier detection by thresholding density estimates is ineffective. In fact, concentration discovery consistently yielded negative events, reflecting the emptiness of 18-dimensional space. Fortunately, pattern discovery provides another alternative. Outliers can be detected as observations which deviate significantly from the dependencies of the majority.

Discovery in the full 18-dimensions with the dependency objective yielded 3 events, one strongly significant event and 2 mildly significant events, covering a total of 23 observations. This leaves 15 observations as candidate outliers, worthy of further investigation. Normally, we might proceed to obtain independent evidence to validate the outliers using methods such as linear constraints, Principal components, Andrews Curves [4], correlation test or gap test. Fortunately, some of this analysis has been previously conducted.

In the projection pursuit analysis of Ripley [101], point #2 was identified from among hundreds of views as a likely outlier. Fortuitously, from a linear constraint perspective, the sum of components of each observation also sheds additional light on possible outliers. Figure 6.50 is a box plot of the total amino acid residues for the 38 observations. The whiskers extend to plausible points while the cross-hairs indicate outliers. The boxplot identifies 8 outliers, only 6 of which are visible due to multiplicity of the totals. Of these 8 outliers, 4 are in common with the points isolated by pattern discovery. Figures 6.51 and 6.52 are plots of these outliers, with low and high total sums, respectively, alongside the significant event. The events

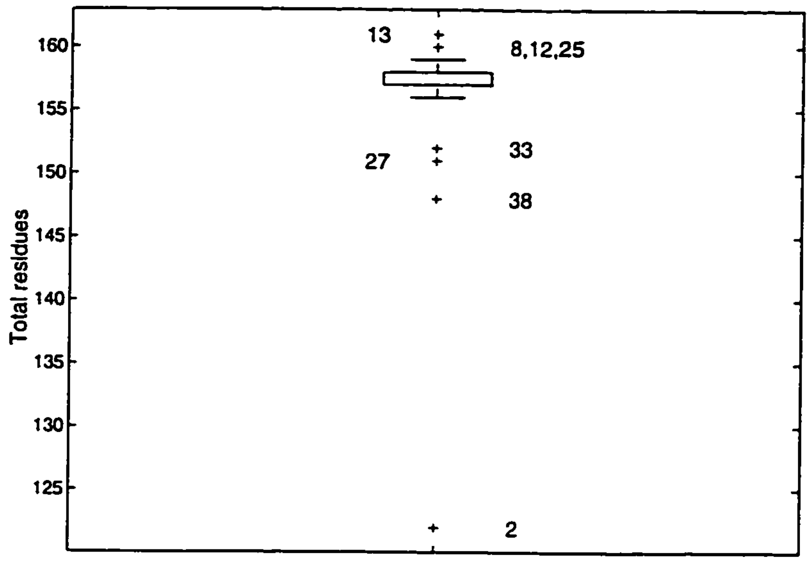


Figure 6.50: Box plot of total amino acid residues. The observation number appears beside each outlier.

are displayed with dotted lines and the outliers, with solid lines.

We can now easily interpret the type of deviations which render these points suspicious. Consider observation #2, the one starting with a lower X_1 value in Figure 6.51. It has substantially lower component values than the main event for $X_1, X_2, X_3, X_4, X_5, X_{14}$ and X_{17} and a higher value for X_{16}. On the other hand, the other observation on this plot, #38, is peculiarly low in its $X_1, X_2, X_6, X_9, X_{11}$ and X_{14} values along with an unusually high value for X_4 and X_7. The candidate outliers in Figure 6.52 possess similar types of subdimensional deviations. Thus, pattern discovery not only identifies suspicious points, but also reveals *how* they are different from the mainstream. Note that although candidate outliers were diagnosed on the basis of dependency, knowledge of the actual high-dimensional relationship was not required.

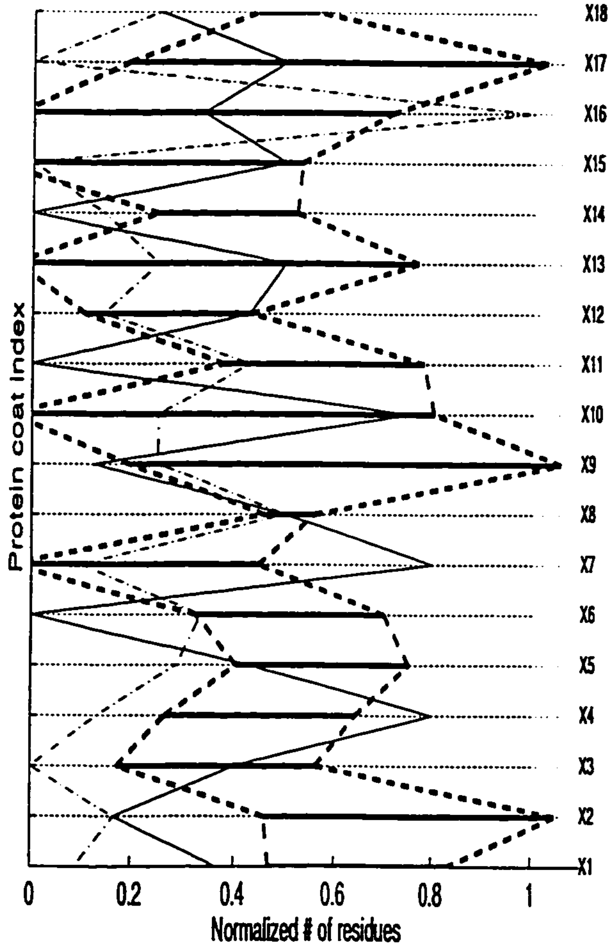


Figure 6.51: "Low" outliers (#2,#38)

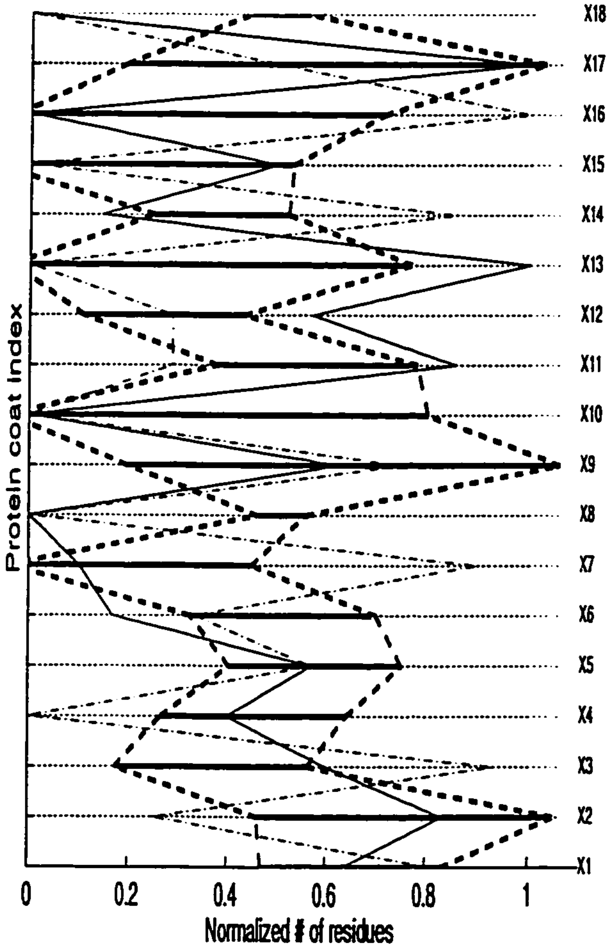


Figure 6.52: "High" outliers (#12,#13)

This final experiment shows that event-based discovery offers a straightforward approach to the detection of elusive high-dimensional outliers in sparse data. In addition to flagging suspicious points, discovery hints at the nature of the marginal and joint deviations.

6.2.7 Summary

The experiments of this chapter show that event-based pattern discovery can contribute valuable information to the collective understanding of a data set. As shown in the analysis of the housing and chemical-overt diabetes data, multivariate data sets can be directly unraveled by the synthesis algorithm, revealing subdimensional clusters and dependencies. With the thyroid diagnosis and PIMA diabetes data, discovery offers straightforward interpretation through simple rules and event plots. Throughout the chapter, we see that the detection of significant events can also complement other data analysis tools for classification, prediction and outlier detection. Tables 6.28 and 6.29 summarize the various experiments, the discovery objectives invoked and the illustrated properties.

Table 6.28: Summary of experiments

Basic experiments

Experiment	Data set	Discovery Objective	Illustrated properties
Discovery by different criteria	Artificial	Concentration, Dependency, Linearity	Types of organization detected by discovery
Noise tolerance	Artificial	Concentration, Dependency, Linearity	Types of "noise" that is rejected by discovery
Non-centralized noise	Artificial	Concentration	Robustness to non-centralized noise
Scale invariance	Housing data	Concentration	Scale insensitive pdf estimates
Partitioning Approximation to optimization	Lipid data Environment data	Concentration Dependency	2-dimensional discovery by partitioning
Event classifier	Artificial	Concentration	Convergence rate, Robustness to noise
High-dimensional Discovery	Artificial	Concentration	Detection of sub-dimensional clusters

Table 6.29: Summary of experiments

Case studies

Experiment	Data set	Discovery Objective	Illustrated properties
Exploratory analysis and interpretation	Thyroid disease	Concentration	Interpretability of discovered patterns
Local dependencies and significant features	Chemical-overt diabetes	Dependency, Concentration	Local dependencies, Locally important features
Pattern discovery and classification	PIMA diabetes	Concentration	Revealing data's limitations
Time-dependent Discovery	EMG data	Concentration	Detection of stationary patterns
Pattern discovery and prediction	Housing data	Dependency	Detection of local dependencies and deviations
Multivariate outliers	*Tobamoviruses* (virus data)	Dependency	Unveiling high-dimensional outliers in sparse data

Chapter 7

Conclusion

7.1 Summary of research completed

In this thesis, I have proposed a framework for pattern discovery, based upon probability theory and the definition of events in Euclidean space. An event is characterized by its volume, observed frequency, probability and statistic value. Patterns are defined as statistically significant events according to a discovery objective. Three such objectives are formulated and the pattern discovery problem is cast as an algorithm in mathematical optimization. The statistical backbone of event-based discovery is the test of homogeneity in a 2-way contingency table. The appropriate expressions have been derived for the statistical testing of candidate events. To perform discovery, a genetic algorithm is employed, guided by the sequential and recursive search strategies.

For low-dimensional data, the hierarchical maximum entropy discretization approach [126], is revised to satisfy theoretical constraints. Enhancements, including boundary refinement and adaptive parameters, are instrumental in providing a good approximation to optimization when the data is sufficiently dense. From the discovered events, a number of extensions are immediate. A probabilistic description and a general kernel method arise from the detected events and subsequently a classifier can be constructed. Tracking of temporal patterns is based on the concept of significant event shifts while event synthesis employs a bottom-up procedure to handle high-dimensional data.

Experiments illustrated the ease of applying and interpreting the wealth of information contained in events. Through various case studies, pattern discovery is seen to be a useful nonparametric tool in exploratory analysis, local feature selection and outlier detection.

7.2 Summary of contributions

The following list summarizes what I believe to be theoretical contributions of this thesis to pattern analysis, and especially to an event-based approach.

A) An event level framework for continuous data. A bridge is established between probability theory and pattern discovery on the basis that events are the fundamental information bearing entity in $\Re^d$. It is shown that some existing intelligent data analysis methods can be understood from an event level perspective. A unique characteristic of this framework is its natural accommodation of local organization in subspaces of less than the full dimensionality.

B) Consistent probability density estimate. It is argued that pdf estimation using maximum entropy recursive partitioning is asymptotically consistent.

C) Discovery by residual analysis. A unique application of a 2-way contingency table, with product-multinomial sampling is proposed for pattern discovery. The large sample maximum likelihood estimate of the residual variance is derived and a test of homogeneity of proportions is put forth as the main statistical hypothesis. Based on this construction, data of any dimensionality can be analyzed by a 2 dimensional table. Furthermore, different discovery objectives can take advantage of a common contingency table. The connection between Haberman's formula and projection matrices is established.

D) Discovery as optimization. Pattern discovery is posed as a general mathematical optimization problem, befitting of any discovery objective. Specifically, objective functions for concentration, dependence and linearity are formulated.

E) Support for projections. Theoretical arguments are given in support of synthesizing low-dimensional orthogonal projections by different discovery objectives.

From the methodological front, the following contributions are considered relevant to pattern discovery.

A) Enhanced recursive partitioning. The original hierarchical maximum entropy discretization scheme of Chiu [126] is revamped to adhere to theoretical assumptions. The development of boundary refinement and prescriptions for an adaptive partition size and an adaptive significance level, greatly enhance the applicability of recursive partitioning.

B) Recursive and sequential optimization. The idea of recursive optimization is proposed as a way to converge upon local optima by successive, crude searches. In particular, its viability with a genetic algorithm has been demonstrated. Another novel idea, sequential discovery is presented as a method to remove known structure and resume the search in a reduced space. The combination of these ideas is supported by the successful optimization of the non-smooth and discontinuous discovery objectives. It is believed that the recursive approach could be applied to general optimization problems.

C) High-dimensional discovery. The curse of dimensionality is overcome by assembling high-dimensional events from low dimensional orthogonal projections. The combination of statistical testing and a new event synthesis procedure enables the detection of subdimensional organization. High-dimensional and sparse data sets can be directly analyzed.

D) General kernel method. For the purpose of obtaining a smooth representation, pattern discovery provides the infrastructure to simultaneously determine the number of kernels, their location and their respective bandwidths. The only restriction is that the covariance matrix is positive definite. The progression from a discrete to continuous estimate, from compact to infinite support, eliminates the need for iteratively training the kernel parameters.

E) Time-dependent discovery. Using categorical analysis, event-based discovery provides a new, objective measure of significant changes over time. Unlike ad-hoc thresholds, the measure of a significant event shift is problem independent and facilitates the tracking of dynamic patterns while circumventing full retraining at every time instant.

7.3 Directions for future work

The theory developed in this thesis suggest some natural generalizations.

A) Abstract events. The concept of an event can be generalized further. Discovered events in Euclidean space $\Re^d$ can be invertibly mapped into points in an abstract space, Υ. Discovery can then be performed in Υ, where the discovered events now represent significant joint occurrences or dependencies among the original events in $\Re^d$. These abstract events could be applied in the investigation of relationships among events.

B) Small sample estimates. The developed residual test statistics are based upon asymptotic assumptions which may be violated when the expected cell counts dwindle [58] or when dimensionality explodes. Low sample scenarios may arise even with dense data, after several rounds of recursion. A resampling statistic [42] may be more viable in such circumstances.

C) Imprecise events. Events can be naturally generalized to have graded rather than crisp boundaries. These imprecise events would constitute fuzzy membership functions and could be applied to the analysis of data with possibilistic uncertainties.

The experimental findings suggest that several issues deserve further attention in future work.

A) Improved GA. The presently employed genetic algorithm is sluggish. The incorporation of elitism may hasten discovery of local optima by retaining sets of highly significant events across generations. In addition, diploidy and dominance [12] may further enhance the discovery of significant events by maintaining a broader diversity of genes within the population.

B) A better low-dimensional approximation. Although partitioning provides acceptable coverage and speed, the resulting representations can be profusely extravagant. A better low-dimensional approximation may adopt a maximum cohesiveness rather than a maximum entropy strategy. The former would be more akin to the the preservation of local structure as promoted by techniques such as projection pursuit. However, if the optimization of events can be sufficiently accelerated, then a low-dimensional approximation would be redundant.

C) Event synthesis with projection pursuit. To expand the scope of detectable high-dimensional structure, event synthesis may be combined with projection pursuit, where the latter determines "interesting" projections for merging.

Appendix A

Proofs of propositions

A.1 Derivation of the asymptotic distribution

In this section, I step through the derivation of the asymptotic distribution of the residual statistic. The derivation follows closely the approach of Christensen [26, p.388-9] in the derivation of the distribution os a different cell statistic. I will first clarify some required notation and definitions.

A.1.1 Notation and definitions

The o, O, o_p and O_p notation for describing limiting behaviour of real numbers and random variables is used. See [19, pages 459,475]. Let F be a function which maps from $\Re^s$ to $\Re^t$ with $F(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_t(\mathbf{x}))'$. The derivative, $d_x F$ denotes the $t \times s$ matrix of partial derivatives, $d_x F = [\frac{\partial f_i}{\partial x_j}]$. A log linear model is assumed to be of the form,

$$\mu \overset{\text{def}}{=} \log(\mathbf{m}) = \mathbf{X}b \qquad \text{(A.1)}$$

where $\mathbf{m}$ is the vector of expected frequencies, $\mathbf{X}$ is the model matrix and $\mathbf{b}$ is the vector of estimated parameters. In the maximum likelihood estimation of log-linear model parameters, a log-likelihood function suitable for all sampling schemes is [26, p.368],

$$f(\mathbf{b}) = \mathbf{n}'\mu - \mathbf{J}'\mathbf{m} \tag{A.2}$$

where $\mathbf{J}$ is the vector of 1's. From (A.1) it is evident that $\mu = \mu(\mathbf{b}) = X\mathbf{b}$ and $\mathbf{m} = \mathbf{m}(\mathbf{b}) = \exp(X\mathbf{b})$ are both functions of $\mathbf{b}$. To find maximum likelihood estimates (MLE) of model parameters, the first derivative of $f(\mathbf{b})$ is set to 0.

$$d_b f(\mathbf{b}) = d_\mu f \;\; d_b\mu \;\; = \;\; (\mathbf{n} - \mathbf{m}(\mathbf{b}))'X = 0 \tag{A.3}$$

The MLE of $\mathbf{b}$ is denoted $\hat{\mathbf{b}} = \hat{\mathbf{b}}(\mathbf{n})$ and is defined implicitly as the unique solution to Equation (A.3). The uniqueness stems from the fact that $f(\mathbf{b})$ is strictly convex (See Christensen [26, p.369]). In this way, the MLE $\hat{\mathbf{b}}$ uniquely defines the MLE $\hat{\mathbf{m}} = \hat{\mathbf{m}}(\mathbf{n}) = \mathbf{m}(\hat{\mathbf{b}})$. The question of existence of MLE's for log-linear models is rigorously addressed in Haberman [57].

A.1.2 Derivation

In a nutshell, the approach is to write the desired difference, $\hat{\mathbf{e}} = \mathbf{n} - \hat{\mathbf{m}}$ as a linear function of $\mathbf{n} - \mathbf{m}$, whose asymptotic distribution we know. Then, employing the delta method, the limiting distribution of the residual is obtained.

Two lemmas from Christensen [26] will be used in the derivation. They are included here for reference.

Lemma 1 *Let a be a scalar and let* $\mathbf{n}$ *be a $q \times 1$ vector of counts, then* $\hat{\mathbf{m}}(a\mathbf{n}) = a\hat{\mathbf{m}}(\mathbf{n})$

Lemma 2 *Let* $\mathbf{p} = \frac{\mathbf{m}}{N}$, *then* $\hat{\mathbf{m}}(\mathbf{p}) = \mathbf{p}$

To obtain $\mathbf{n} - \hat{\mathbf{m}}$ as linear function of $\mathbf{n} - \mathbf{m}$, the idea is to write the Taylor series expansion for $\hat{\mathbf{m}}$ as an implicit function of $\mathbf{n}$, around $\mathbf{m}$. Equivalently, expand $\hat{\mathbf{m}}$ as a function of the sample proportion, $\mathbf{s} = \frac{\mathbf{n}}{N}$ around $\mathbf{p}$, where $\mathbf{p} = \frac{\mathbf{m}}{N}$. This equivalent approach will facilitate the use of some proven results. The expansion is,

$$\hat{\mathbf{m}}(\mathbf{s}) = \hat{\mathbf{m}}(\mathbf{p}) + d_n\hat{\mathbf{m}}(\mathbf{p})(\mathbf{s} - \mathbf{p}) + O(||\mathbf{s} - \mathbf{p}||) \tag{A.4}$$

where $||\mathbf{z}|| = (\sum_i z_i^2)^{1/2}$. By a sequence of manipulations this equation will yield the desired asymptotic result. First, consider the derivative, $d_n\hat{\mathbf{m}}(\mathbf{n})$. Using the chain rule, it can be written as,

$$d_n\hat{\mathbf{m}}(\mathbf{n}) = d_{\hat{b}}\hat{\mathbf{m}}\; d_n\hat{\mathbf{b}} \tag{A.5}$$

The derivative of $\hat{\mathbf{b}}$ is obtained by applying the corollary to the implicit function theorem [26, pp.387-8].

$$d_n\hat{\mathbf{b}} = (X'D(\hat{\mathbf{m}})X)^{-1}X' \tag{A.6}$$

Here, $D(\hat{\mathbf{m}})$ is a diagonal matrix with the elements of $\hat{\mathbf{m}}$ on the main diagonal. To compute the derivative of $\hat{\mathbf{m}}$ note that $\hat{\mathbf{m}} = \exp(X\hat{\mathbf{b}})$. The $q \times p$ matrix of partial derivatives is,

$$d_b\hat{\mathbf{m}}(\mathbf{n}) = D(\hat{\mathbf{m}})X \tag{A.7}$$

Substituting (A.6) and (A.7) into (A.5), we have that

$$d_n \hat{\mathbf{m}}(\mathbf{n}) = D(\hat{\mathbf{m}}(\mathbf{n}))X(X'D(\hat{\mathbf{m}}(\mathbf{n}))X)^{-1}X' \tag{A.8}$$

Recall that in Equation (A.4) we require $d_n \hat{\mathbf{m}}(\mathbf{p})$. Since (A.8) is true for any n, we can substitute $\mathbf{p} = mN^{-1}$ in place of $\mathbf{n}$, obtaining

$$d_n \hat{\mathbf{m}}(\mathbf{p}) = D(\hat{\mathbf{m}}(\mathbf{p}))X(X'D(\hat{\mathbf{m}}(\mathbf{p}))X)^{-1}X' \tag{A.9}$$

Fortunately, lemma 2 states that $\hat{\mathbf{m}}(\mathbf{p}) = \mathbf{p}$. This leads to the simplification $D(\hat{\mathbf{m}}(\mathbf{p})) = D(\mathbf{p})$. For simplicity, we'll just denote $D(\mathbf{p})$ as D. Now the derivative can be compactly written as,

$$d_n \hat{\mathbf{m}}(\mathbf{p}) = DAD^{-1} \tag{A.10}$$

where $A \equiv X(X'DX)^{-1}X'D$. Substituting this derivative into Equation (A.4) and rearranging terms we have,

$$\hat{\mathbf{m}}(N^{-1}\mathbf{n}) - \hat{\mathbf{m}}(N^{-1}\mathbf{m}) - DAD^{-1}(N^{-1}\mathbf{n} - N^{-1}\mathbf{m}) = O(||N^{-1}\mathbf{n} - N^{-1}\mathbf{m}||) \tag{A.11}$$

where $\mathbf{s}$ and $\mathbf{p}$ have also been replaced by their respective definitions. By applying lemmas 1 and 2 in Christensen [26, p.387], the first 2 terms on the left-hand side simplifies to $N^{-1}\hat{\mathbf{m}} - N^{-1}\mathbf{m}$. Furthermore, the right-hand side is the difference between the sample proportion and its mean and thus by Tchebychev's Inequality is $O_p(N^{-1/2})$ or equivalently, $o_p(1)$. Incorporating this information and multiplying through by $N^{1/2}$, Equation (A.11) becomes,

$$N^{-1/2}\hat{\mathbf{m}} - N^{-1/2}\mathbf{m} - DAD^{-1}(N^{-1/2}\mathbf{n} - N^{-1/2}\mathbf{m}) = o_p(1) \qquad (A.12)$$

Multiplying through by −1, adding and subtracting $N^{-1/2}\mathbf{n}$ and letting $N \to \infty$ we arrive at,

$$N^{-1/2}(\mathbf{n} - \hat{\mathbf{m}}) - (I - DAD^{-1})N^{-1/2}(\mathbf{n} - \mathbf{m}) \xrightarrow{p} 0 \qquad (A.13)$$

The difference between $N^{-1/2}(\mathbf{n}-\hat{\mathbf{m}})$ and $(I-DAD^{-1})N^{-1/2}(\mathbf{n}-\mathbf{m})$ approaches 0 in probability as $N \to \infty$. Hence, by Rao [98, p.101], we know that the 2 expressions have the same limiting distribution. The limiting distribution of $N^{-1/2}(\mathbf{n}-\mathbf{m})$ is given by the large-sample multinomial result,

$$N^{-1/2}(\mathbf{n} - \mathbf{m}) \xrightarrow{\mathcal{D}} \mathcal{N}(0, D(\mathbf{p})(I - A_0)) \qquad (A.14)$$

where $A_0 = \mathbf{J}\mathbf{J}'D(\mathbf{p})$ is an orthogonal projection matrix. By a trivial application of the delta method, we obtain,

$$N^{-1/2}(\mathbf{n} - \hat{\mathbf{m}}) \xrightarrow{\mathcal{D}} \mathcal{N}(0, (I - DAD^{-1})'(D(I - A_0))(I - DAD^{-1})) \qquad (A.15)$$

Finally, by exploiting the fact that A is idempotent and that $A_0A = A_0$, we obtain the large-sample distribution of the simple residual,

$$N^{-1/2}(\mathbf{n} - \hat{\mathbf{m}}) \xrightarrow{\mathcal{D}} \mathcal{N}(0, D(I - A)) \qquad (A.16)$$

where $A = X(X'DX)^{-1}X'D$.

A.2 Closed-form estimate of the asymptotic variance of the residual

In this section, I attempt to link the the compact expression for the asymptotic covariance matrix given in Equation (4.23) to the more common expressions such as, $D(\hat{\mathbf{m}})(\mathbf{I} - A(\hat{\mathbf{m}}))$ which are built around the projection matrix A. In particular, the focus here is to highlight the connections between the generating class and the projection matrix. To the best of my knowledge, no prior attempt has been made to point out these equivalences. This understanding is crucial to the proper application of Equation (4.23).

For definitions of generating and intersection classes and decomposability the reader is referred to [57].

Generating class induces a projection matrix

We shall begin by relating the generating class to an orthogonal projection matrix. Suppose we have a table of q counts given by the vector $\mathbf{n} = (n_1, n_2, \ldots, n_q)'$. Suppose further that the log-linear model of interest is both hierarchical and decomposable. Thus, there exists a closed-form estimate $\hat{\mathbf{m}}$, in the form of Equation (4.22). Let the generating class be $\mathcal{S} = \{S_1, S_2, \ldots, S_G\}$. Consider the j^{th} element, S_j with $L_j = L(S_j)$ distinct levels. For S_j, define a $q \times 1$ binary-valued vector, $\mathbf{v}_{jk}$. This vector selects the cells which contribute to the marginal sum of the k^{th} level. A

1 is placed in the positions corresponding to the selected cells and 0's are placed everywhere else. Such a vector can be formed for each of the k levels, $k = 1, \ldots, L_j$, so that associated with each element S_j is a set of vectors,

$$\{\mathbf{v}_{jk}, \ \ k = 1, \ldots, L_j\} \tag{A.17}$$

Each column vector, $\mathbf{v}_{jk}$ determines a linear vector space $\mathcal{M}(S_{jk})^1$, spanned by a single vector,

$$\mathcal{M}(S_{jk}) = \text{span}(\mathbf{v}_{jk}) \tag{A.18}$$

The following proposition allows us to use these vectors to write the subspace associated with each generating element, S_j.

Proposition 3 *The set of L_j vectors $\{\mathbf{v}_{jk}\}$, $k = 1, \ldots, L_j$ associated with the generating element S_j has the following properties:*

(i) $L_j < q$, where q is the number of cells in the table.

(ii) The vector of ones, $\mathbf{J}$ is always representable by the vectors $\{\mathbf{v}_{jk}\}$.

Proof:

Consider the j^{th} generating element S_j with L_j levels. The $q \times 1$ column vector $\mathbf{v}_{jk}$, composed entirely of 1's and 0's, selects the cells contributing to the marginal sum of the k^{th} level, $k = 1, \ldots, L_j$. Note that the i^{th} cell, $i = 1, \ldots, q$, is selected exactly once by the set of vectors, $\{\mathbf{v}_{jk}\}$. As a result, these vectors are mutually

[1] All the linear vector spaces referred to here are subspaces of $\Re^q$, with vector addition and scalar multiplication defined as in Euclidean space.

orthogonal,

$$\mathbf{v}_{jk} \cdot \mathbf{v}_{jl} = 0 \quad \forall k \neq l \tag{A.19}$$

and hence linearly independent.

Statement (i) To verify statement (i), consider a table with z variables. Let the i^{th} variable have $L_i \geq 2$ levels. The total number of cells in the table will be $q = L_1 \times L_2 \times \ldots \times L_z$. Since only tables of 2 or more variables are of practical interest, we see that $q > L_i$, $i = 1, \ldots, z$.

Statement (ii) Let v be a vector within span$\{\mathbf{v}_{jk}\}$, i.e.,

$$\mathbf{v} = \sum_{k=1}^{L_j} \alpha_k \mathbf{v}_{jk} \tag{A.20}$$

where $\alpha_k \in \Re$, for all k. If we set, $\alpha_1 = \alpha_2 = \ldots = \alpha_{L_j} = 1$, we have,

$$\mathbf{v} = \sum_{k=1}^{L} \mathbf{v}_{jk} = \mathbf{J} \tag{A.21}$$

where $\mathbf{J}$ is the vector of 1's. The last equation is true because each of the q cells is selected exactly once by the vectors $\{\mathbf{v}_{jk}, \ \ k = 1, \ldots, L_j\}$.

◇

Applying the above proposition, the subspace of $\Re^q$ determined by S_j is thus,

$$\mathcal{M}(S_j) = \text{span}(\{\mathbf{v}_{jk}, \ \ k = 1, \ldots, L_j\}) \tag{A.22}$$

The vectors $\{\mathbf{v}_{jk}\}$ can be written for each S_j. Therefore, a generating class, $\mathcal{S} = \{S_1, \ldots, S_G\}$ will give rise to the vectors,

$$\{\mathbf{v}_{jk},\ \ j = 1, \ldots, G\ ,\ k = 1, \ldots, L_j\} \tag{A.23}$$

The subspace determined by the generating class $\mathcal{S}$ is given by,

$$\mathcal{M}(\mathcal{S}) = \mathrm{span}(\{\mathbf{v}_{jk},\ \ j = 1, \ldots, G,\ k = 1, \ldots, L_j\}) \tag{A.24}$$

The next proposition relates the subspace $\mathcal{M}(\mathcal{S})$ to the model matrix, X, of a log-linear model.

Proposition 4 *Let $\mathcal{M}(\mathcal{S})$ be defined as above. Let X be the $q \times p$ model matrix for a log-linear model,* $\log \mathbf{m} = X\mathbf{b}$. *Let $\mathbf{x}$ be a column of X. Then,*

(i) Every column of X is contained in $\mathcal{M}(\mathcal{S})$, i.e., $\mathbf{x} \in \mathcal{M}(\mathcal{S})$

(ii) col(X) = $\mathcal{M}(\mathcal{S})$

Proof:

Let $V = \{\mathbf{v}_{jk}, j = 1, \ldots, G, \ , k = 1, \ldots, L\}$. We know that any p linearly independent column vectors selected from V forms an equivalent parameterization of X for the log-linear model, $\log \mathbf{m} = X\mathbf{b}$ [26]. Define a matrix $\mathbf{M} = [M_1 M_2 \ldots M_p]$, where M_i are the column vectors that span $\mathcal{M}(\mathcal{S})$. Note that,

$$M_i \subset V \tag{A.25}$$

for all i. This is true because $\mathcal{M}(\mathcal{S}) \overset{\text{def}}{=} \text{span}(V)$. Hence, M is a parameterization of the model matrix X and we can write,

$$\text{col}(\mathbf{M}) = \text{col}(\mathbf{X}) = \mathcal{M}(\mathcal{S}) \tag{A.26}$$

$\diamond$

The subspace $\mathcal{M}(\mathcal{S})$ is therefore equivalent to the column space of the model matrix X, that is, $C(X)$. We can always form the standard perpendicular projection matrix onto $C(X)$ with the inner product defined with D.

$$P = X(X'DX)^{-1}X' \tag{A.27}$$

Since P is a perpendicular projection onto $C(X)$, $C(P) = C(X)$ [27]. This first set of arguments shows that the projection matrix P is induced by the generating class $\mathcal{S}$.

Decomposition of $\mathcal{M}(\mathcal{S})$

Next we argue that decomposability of the generating class implies decomposability of the subspace, $\mathcal{M}(\mathcal{S})$. Suppose that a generating class $\mathcal{S}$ is decomposable into $\mathcal{S}_1$ and $\mathcal{S}_2$. We know that $\mathcal{S}_1$ and $\mathcal{S}_2$ both determine subspaces, which, in general need not be orthogonal. From the previous section, we know that together, the linearly independent subset of vectors of $\mathcal{M}(\mathcal{S}_1)$ and $\mathcal{M}(\mathcal{S}_2)$ determine the subspace $\mathcal{M}(\mathcal{S})$.

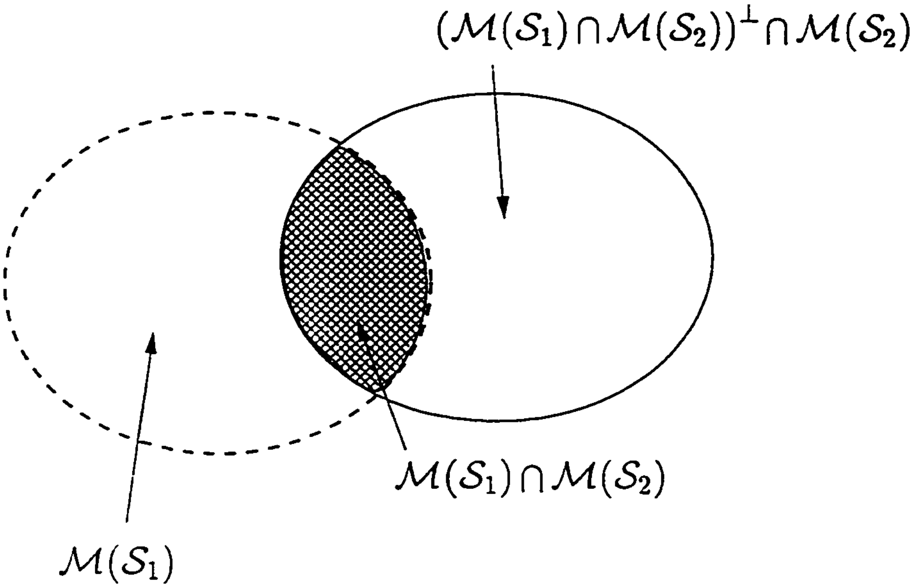

$(\mathcal{M}(\mathcal{S}_1) \cap \mathcal{M}(\mathcal{S}_2))^{\perp} \cap \mathcal{M}(\mathcal{S}_2)$
$\mathcal{M}(\mathcal{S}_1) \cap \mathcal{M}(\mathcal{S}_2)$
$\mathcal{M}(\mathcal{S}_1)$

Figure A.1: Decomposition of $\mathcal{M}(\mathcal{S})$

Hence, we can write,

$$\begin{aligned}
\mathcal{M}(\mathcal{S}) &= \mathcal{M}(\mathcal{S}_1) \bigcup \left(\mathcal{M}(\mathcal{S}_2) \setminus \left[\mathcal{M}(\mathcal{S}_1) \bigcap \mathcal{M}(\mathcal{S}_2)\right]\right) \\
&= \mathcal{M}(\mathcal{S}_1) \oplus \text{Space unique to } \mathcal{M}(\mathcal{S}_2) \\
&= \mathcal{M}(\mathcal{S}_1) \oplus \left[\left(\mathcal{M}(\mathcal{S}_1) \bigcap \mathcal{M}(\mathcal{S}_2)\right)^{\perp} \bigcap \mathcal{M}(\mathcal{S}_2)\right] \qquad \text{(A.28)}
\end{aligned}$$

Here, $\perp$ signifies the complement space and $\oplus$ is the direct sum. The last equality can be understood by looking at Figure A.1. The space $\mathcal{M}(\mathcal{S})$ is decomposed into a direct sum of 2 complementary subspaces. This is done to facilitate the matrix decomposition discussed next.

Decomposition of projection matrix

Recall that since P projects onto $\mathcal{M}(\mathcal{S})$, $C(P) = \mathcal{M}(\mathcal{S})$. Therefore, Equation (A.28) allows us to write P as a combination of projections,

$$P = P_1 + P_Z \tag{A.29}$$

where P_1 is a projection matrix onto $\mathcal{M}(\mathcal{S}_1)$ and P_Z is a projection matrix onto $(\mathcal{M}(\mathcal{S}_1) \bigcap \mathcal{M}(\mathcal{S}_2))^{\perp} \bigcap \mathcal{M}(\mathcal{S}_2)$. The projection P_Z can be further decomposed. Define a matrix P_2 which projects onto $\mathcal{M}(\mathcal{S}_2)$ and a matrix P_3 which projects onto $\mathcal{M}(\mathcal{S}_1) \bigcap \mathcal{M}(\mathcal{S}_2)$. The range space of P_Z can now be written as,

$$\left(\mathcal{M}(\mathcal{S}_1) \bigcap \mathcal{M}(\mathcal{S}_2)\right)^{\perp} \bigcap \mathcal{M}(\mathcal{S}_2) = C(P_3)^{\perp} \bigcap C(P_2) \tag{A.30}$$

From the combinations of projections [10, 27], we know that the matrix $P_2 - P_3$ projects onto this space if $P_2 D P_3 = P_3 D P_2 = P_3$. Fortunately, we can always write P_2 as 2 projection matrices onto complement subspaces,

$$P_2 = P_{21} + P_{22} \tag{A.31}$$

where P_{21} projects onto the same space as P_3, namely, $C(P_3) = \mathcal{M}(\mathcal{S}_1) \bigcap \mathcal{M}(\mathcal{S}_2)$ and P_{22} projects to the complement of $\mathcal{M}(\mathcal{S}_1) \bigcap \mathcal{M}(\mathcal{S}_2)$ in $\mathcal{M}(\mathcal{S})$. By a simple substitution of Equation (A.31), we see that,

$$P_2 D P_3 = (P_{21} + P_{22}) D P_3 = P_3 \tag{A.32}$$

$$\text{and } P_3DP_2 = P_3D(P_{21} + P_{22}) = P_3$$

We conclude therefore that $P_2 - P_3$ projects onto $(\mathcal{M}(\mathcal{S}_1)\bigcap\mathcal{M}(\mathcal{S}_2))^{\perp}\bigcap\mathcal{M}(\mathcal{S}_2)$. Finally, we can write the decomposition of P as

$$P = P_1 + P_2 - P_3 \tag{A.33}$$

where P projects onto $\mathcal{M}(\mathcal{S})$, P_1 projects onto $\mathcal{M}(\mathcal{S}_1)$, P_2 projects onto $\mathcal{M}(\mathcal{S}_2)$ and P_3 projects onto $\mathcal{M}(\mathcal{S}_1)\bigcap\mathcal{M}(\mathcal{S}_2)$. Note that the diagonal matrix D is nonsingular and thus we can relate the column spaces as follows [27],

$$C(PD) = C(P_1D + P_2D - P_3D) \tag{A.34}$$

Hence, post-multiplication by D does not affect the validity of the decomposition and we have,

$$PD = P_1D + P_2D - P_3D \tag{A.35}$$

Before going on, we will need the following proposition to bring the intersection class into the discussion.

Proposition 5 *Let $\mathcal{T}$ be the intersection class between $\mathcal{S}_1$ and $\mathcal{S}_2$. Then,*

$$\mathcal{M}(\mathcal{S}_1)\bigcap\mathcal{M}(\mathcal{S}_2) = \mathcal{M}(\mathcal{T}) \tag{A.36}$$

Proof:

Consider the generating class $\mathcal{S}$ with G elements. Let M be a number less than G. Denote the subclasses of the generating class as $\mathcal{S}_1 = \{S_1, S_2, \ldots, S_{M-1}\}$ and $\mathcal{S}_2 = \{S_M, S_{M+1}, \ldots, S_G\}$. Let $\mathcal{F} = \{S_i \cap S_j\}$ with $S_i \in \mathcal{S}_1$ and $S_j \in \mathcal{S}_2$. If the intersection class is empty, i.e., $\mathcal{F} = \{\emptyset\}$, then we know by Proposition 3, that,

$$\mathcal{M}(\mathcal{S}_1) \bigcap \mathcal{M}(\mathcal{S}_2) = \mathbf{J} \tag{A.37}$$

By definition [57] however, we have that $\mathcal{M}(\mathcal{F}) = \mathbf{J}$ when $\mathcal{T}$ is empty, so that the proposition is verified when $\mathcal{T} = \{\emptyset\}$.

Now consider the case when $\mathcal{T}$ is not empty. Let $\{\mathbf{v}_{jk}, j = 1, \ldots, M-1,\ \ k = 1, \ldots, L(j)\}$ represent the vectors associated with $\mathcal{S}_1$. We can construct $\mathcal{M}(\mathcal{S}_1)$ by selecting all the columns for S_1, the first generating element, and then adding linearly independent columns from other generating elements. In other words,

$$\mathcal{M}(\mathcal{S}_1) = \mathrm{span}(\{\mathbf{v}_{1k}\}, \Gamma_1) \quad k = 1, \ldots, L_1 \tag{A.38}$$

where Γ_1 are the columns associated with generating elements, S_2 to S_{M-1} which are linearly independent of $\{\mathbf{v}_{1k}\}$. The same procedure can be repeated to construct $\mathcal{M}(\mathcal{S}_2)$,

$$\mathcal{M}(\mathcal{S}_2) = \mathrm{span}(\{\mathbf{v}_{Mk}\}, \Gamma_2) \quad k = 1, \ldots, L_M \tag{A.39}$$

where Γ_2 are the columns associated with generating elements, S_{M+1} to S_G which are linearly independent of $\{\mathbf{v}_{Mk}\}$.

Now suppose that S_1 and S_M have a common variable, say W, i.e., $S_i \cap S_j = W$. Define an index set $\mathcal{I}$ such that if $i \in \mathcal{I}$ then $\mathbf{v}_{1i}$ is a vector which selects the k^{th} level of W. Now let $\{\mathbf{v}_w^k\}$ represent the subset of $\{\mathbf{v}_{1i}\}$ in which the k^{th} level of W is selected.

$$\{\mathbf{v}_w^k\} = \{\mathbf{v}_{1i}, i \in \mathcal{I}\} \tag{A.40}$$

Then, by summing the vectors $\{\mathbf{v}_w^k\}$, we can form the column vector, $\mathbf{w}_k$ associated with the k^{th} level of W. Thus, for each level k of W, we can write the associated vector as,

$$\mathbf{w}_k = \sum_{i \in \mathcal{I}} \mathbf{v}_{1i} = \sum \mathbf{v}_w^k \tag{A.41}$$

where the summation is over all vectors in which the k^{th} level of W is selected. This is always possible because the marginal sum for a variable W, contained in S_1. can be obtained from the marginal sum of S_1, simply by summing over all levels of variables in S_1 other than W. We can write Equation (A.41) for each level of W. Thus, there will be L_w new vectors, where L_w is the number of levels in the variable W.

The same procedure can be repeated with the columns of S_M to extract the columns associated with the common variable W. Hence, we can conclude that each vector in the set $\{\mathbf{w}_k\}$ lies in both the columns associated with S_1 and those of S_M, i.e.,

$$\mathbf{w}_k \in \operatorname{span}(\{\mathbf{v}_{1i}, \ i = 1, \ldots, L_1\}) \tag{A.42}$$

$$\mathbf{w}_k \in \operatorname{span}(\{\mathbf{v}_{Mi}, \ i = 1, \ldots, L_M\}) \tag{A.43}$$

for $k = 1, \ldots, L_w$. Now note that the vectors $\{\mathbf{w}_k\}$ select the marginal sums of W and determine the subspace $\mathcal{M}(\mathcal{W}) = \mathcal{M}(S_1 \cap S_M)$,

$$\mathrm{span}(\{\mathbf{w}_k\}) = \mathcal{M}(S_1 \cap S_M) \tag{A.44}$$

Therefore the intersection of $\mathcal{M}(\mathcal{S}_1)$ and $\mathcal{M}(\mathcal{S}_2)$ is given by,

$$\mathcal{M}(\mathcal{S}_1) \bigcap \mathcal{M}(\mathcal{S}_2) = \mathcal{M}(S_1 \cap S_M) \tag{A.45}$$

If we continue this procedure of checking elements in $S_i \in \mathcal{S}_1$ and $S_j \in \mathcal{S}_2$ in a pairwise fashion, we will obtain,

$$\begin{aligned} \mathcal{M}(\mathcal{S}_1) \bigcap \mathcal{M}(\mathcal{S}_2) &= \mathcal{M}(\{S_1 \cap S_M, S_1 \cap S_{M+1}, \ldots, S_{M-1} \cap S_G\}) \\ &= \mathcal{M}(\mathcal{T}) \end{aligned} \tag{A.46}$$

◇

Thus, in the above discussion, we can consider the matrix P_3 as projecting onto $\mathcal{M}(\mathcal{T})$.

Recursive decomposition

The above decomposition of the projection matrix can be recursively applied to P_1, P_2 and P_3 until there is a projection matrix corresponding to every level of every generating and intersecting class element, i.e. there is a projection onto every

$\mathcal{M}(S_{jk})$ and $\mathcal{M}(T_{jk})$.

$$\begin{aligned} PD &= P_1D + P_2D - P_3D \qquad \text{(A.47)} \\ &= (P_{11} + P_{12} - P_{13})D + (P_{21} + P_{22} - P_{23})D - (P_{31} + P_{32} - P_{33})D \\ &= \dots \\ &= \left(\sum_{j}^{G} \sum_{k=1}^{L(S_j)} P(S_{jk}) - \sum_{j}^{G-1} \sum_{k=1}^{L(T_j)} P(T_{jk}) \right) D \end{aligned}$$

where $P(S_{jk})$ is the projection matrix onto $\mathcal{M}(S_{jk})$ and $P(T_{jk})$ is the projection matrix onto $\mathcal{M}(T_{jk})$. The number of levels in S_j and T_j have been indicated by $L(S_j)$ and $L(T_j)$ respectively.

Simplifying projection matrices

Consider the projection matrix $P(S_{jk}) = \mathbf{v}_{jk}(\mathbf{v}_{jk}'D\mathbf{v}_{jk})^{-1}\mathbf{v}_{jk}'$, corresponding to the k^{th} level of the generating class element S_j. The expression $\mathbf{v}_{jk}'D(\hat{\mathbf{m}})\mathbf{v}_{jk}$ basically computes the marginal sum for the k^{th} level of generating element S_j, that is,

$$\mathbf{v}_{jk}'D(\hat{\mathbf{m}})\mathbf{v}_{jk} = \hat{\mathbf{m}}^{S_{jk}} \qquad \text{(A.48)}$$

Hence, we can write the projection matrix as,

$$P(S_{jk}) = \frac{1}{\hat{\mathbf{m}}^{S_{jk}}}\mathbf{v}_{jk}\mathbf{v}_{jk}' \qquad \text{(A.49)}$$

The main diagonal of the $q \times q$ matrix $\mathbf{v}_{jk}\mathbf{v}_{jk}'$ is just the elements of $\mathbf{v}_{jk}$. We can make 3 observations about the matrix $\mathbf{v}_{jk}\mathbf{v}_{jk}'$,

1. The matrix consists entirely of 1's and 0's.

2. Each of the q diagonal elements corresponds to exactly 1 cell in the table.

3. The non-zero diagonal elements correspond to cells which contribute to the k^{th} marginal sum.

Hence, the non-zero diagonal entries of the matrix $P(S_{jk})$ are simply $\frac{1}{\hat{\mathbf{m}}^{S_{jk}}}$.

When we sum over all the k levels of S_j, we obtain the the projection matrix onto $\mathcal{M}(S_j)$, namely,

$$P(S_j) = \sum_{k=1}^{L_j} P(S_{jk}) = \sum_{k=1}^{L_j} \frac{1}{\hat{\mathbf{m}}^{S_{jk}}} \mathbf{v}_{jk}\mathbf{v}_{jk}' \tag{A.50}$$

Note that the vectors $\{\mathbf{v}_{jk}\}$ consist of only 1's and 0's and are mutually orthogonal. Along with the above observations, we see that no two matrices, $\mathbf{v}_{jk}\mathbf{v}_{jk}'$ and $\mathbf{v}_{jl}\mathbf{v}_{jl}'$, $k \neq l$, will have non-zero entries in the same position. Hence, the non-zero main diagonal entries of $P(S_j)$ are the L_j inverted marginal totals, $\frac{1}{\hat{\mathbf{m}}^{S_{jk}}}$, corresponding to S_j. In particular, the i^{th} diagonal entry is given by,

$$\text{diag}_i\left(P(S_j)\right) = \frac{1}{\hat{\mathbf{m}}^{S_{jk}}} \tag{A.51}$$

where $\hat{\mathbf{m}}^{S_{jk}}$ is the only marginal sum of S_j to which cell i contributes. For the sake of brevity, we will not explicitly write the dependence of the right-hand side on the cell i.

When we sum $P(S_j)$ over the generating class elements $j = 1, \ldots, G$, the i^{th} diagonal entry of the resulting matrix becomes,

$$\operatorname{diag}_i\left(\sum_{j=1}^{G} P(S_j)\right) = \sum_{j=1}^{G} \frac{1}{\hat{\mathrm{m}}^{S_{jk}}} \tag{A.52}$$

In fitting the log-linear model, the initial constraint was that the marginal sums of the *estimated* expected frequencies must equal those of the observed frequencies. Hence, we can replace marginal totals $\frac{1}{\hat{\mathrm{m}}^{S_{jk}}}$ with $\frac{1}{\mathrm{n}^{S_{jk}}}$. The same arguments apply for simplification of the intersection class projection matrices. The diagonals of the matrix PD are thus given by,

$$\begin{aligned}
\operatorname{diag}_i(PD) &= \left[\operatorname{diag}_i\left(\sum_{j=1}^{G} P(S_j)\right) - \operatorname{diag}_i\left(\sum_{j=1}^{G-1} P(T_j)\right)\right] \times \operatorname{diag}_i(D(\hat{\mathrm{m}})) \\
&= \left[\sum_{j}^{G} \frac{1}{\hat{\mathrm{m}}^{S_{jk}}} - \sum_{j}^{G-1} \frac{1}{\hat{\mathrm{m}}^{T_{jk}}}\right] \times \operatorname{diag}_i\left(D(\hat{\mathrm{m}})\right) \qquad (A.53) \\
&= \hat{\mathrm{m}}_i \sum_{j}^{G} \frac{1}{\mathrm{n}^{S_{jk}}} - \hat{\mathrm{m}}_i \sum_{j}^{G-1} \frac{1}{\mathrm{n}^{T_{jk}}}
\end{aligned}$$

Asymptotic variance

The last step is to recognize that $A(\hat{\mathrm{m}}) = PD(\hat{\mathrm{m}})$, where P is the projection matrix of Equation (A.27). Therefore, the i^{th} diagonal element of the asymptotic covariance matrix, $D(\hat{\mathrm{m}})[I - A(\hat{\mathrm{m}})]$ is written as,

$$\begin{aligned}
\hat{c}_i &= \operatorname{diag}_i(D(\hat{\mathrm{m}}))\left[\operatorname{diag}_i(I) - \operatorname{diag}_i\left(A(\hat{\mathrm{m}})\right]\right) \qquad (A.54) \\
&= \hat{m}_i\left(1 - \hat{m}_i \sum_{j=1}^{G} \frac{1}{\mathrm{n}^{S_{jk}}} + \hat{m}_i \sum_{j=1}^{G-1} \frac{1}{\mathrm{n}^{T_{jk}}}\right)
\end{aligned}$$

For brevity of notation, we can omit the level subscript k and remember that the level is implied by the generating element j.

A.3 Recursive partitioning: Estimation of the maximum entropy of a partition

In reference to Section 5.1.3, recursive partitioning has a natural interpretation as the estimation of the maximum entropy of the initial partition. The entropy of a partition of q cells is defined as [107. 72],

$$H(P_1, P_2, \ldots, P_q) = \sum_{k=1}^{q} P_k \log P_k \tag{A.55}$$

where P_i are cell probabilities. The maximum entropy of a partition is achieved when all probabilities are equalized,

$$H_{max} = H(\frac{1}{q}. \frac{1}{q}, \ldots, \frac{1}{q}) \stackrel{\text{def}}{=} A(q) \tag{A.56}$$

Suppose that we partition a sample space into Q^d cells and that $M < Q^d$ of these cells are repartitioned. From the definition of entropy [107, p.393], we know that the branching property is satisfied. Hence, the maximum entropy of the initial partition into Q^d cells can be approximated as,

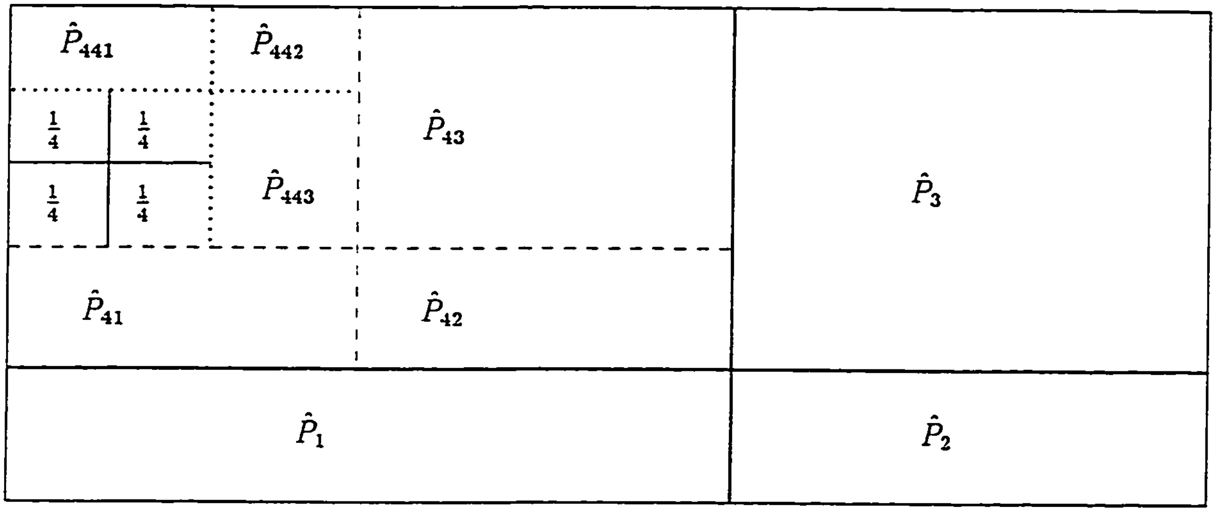
$$\hat{A}_1(Q^d) = H(\hat{P}_1, \ldots, \hat{P}_{Q^d - M}) + \sum_{j=1}^{M} \sum_{i=1}^{Q^d} \hat{P}_{jk} \hat{A}_{2j}(Q^d) \tag{A.57}$$

where $\hat{P}_i$ is the probability estimate for the i^{th} cell and $\hat{A}_{2j}$ is the estimated maximum entropy of the partition of cell j. The subscript 2 indicates that this estimate is obtained from the second level of partitioning. We see that the approximation is recursive. Eventually, at the r^{th} level of recursion, the estimate $\hat{A}_r(Q^d)$ will equal the true $A(Q^d) = log(Q^d)$, or perhaps due to a lack of observations may become smaller than some threshold value. In either case, the recursive estimation would cease.

To clarify, consider a simple example of 4 levels of recursive partitioning in 2-dimensions, where at each level, only 1 cell is repartitioned. Let the partition size be $Q = 2$. Further suppose that at the fourth level of recursion, true frequency equalization is achieved. The estimates of the maximum entropy at each level of recursion are then,

$$\begin{aligned}
\hat{A}_1(4) &= H(\hat{P}_1, \hat{P}_2, \hat{P}_3) + \sum_{i=1}^{4} \hat{P}_{4i}\hat{A}_2(4) \qquad \text{(A.58)}\\
\hat{A}_2(4) &= H(\hat{P}_{41}, \hat{P}_{42}, \hat{P}_{43}) + \sum_{i=1}^{4} \hat{P}_{44i}\hat{A}_3(4)\\
\hat{A}_3(4) &= H(\hat{P}_{441}, \hat{P}_{442}, \hat{P}_{443}) + \sum_{i=1}^{4} \hat{P}_{444}A_4(4)
\end{aligned}$$

At the final level of recursion, we have $A_4(4) = log(4)$ as the true value of the maximum entropy. The partitioning and the associated probabilities are shown in Figure A.3. To obtain a numeric value for the maximum entropy of the initial partition, simply back substitute each equation into its predecessor.



captionExample of recursive partitioning for maximum entropy estimation

A.4 Termination condition

In the derivation of the large sample variance of the residual [57, 2, 34, 26], the central limit theorem is applied to the summation of multinomials, yielding an asymptotically normal distribution. For this approximation to hold, the expected cell frequency should be sufficiently large, typically at least 25 samples. Let ζ_{min} denote this minimum expected cell frequency. We therefore constrain the expected frequency given by Equation (4.28),

$$\hat{m}_j = \frac{1}{2}(n_{1j} + n_{2j}) \geq \zeta_{min} \tag{A.59}$$

If we substitute for n_{2j} using Equation (4.34) and solve for the event volume, v_j, we arrive at a lower bound for v_j.

$$v_j \geq \frac{V_{TOT}}{n_{1+}}(2\zeta_{min} - n_{1j}) \ = \ \text{lower bound} \tag{A.60}$$

If the volume of an event falls below this lower bound, partitioning should not proceed. This condition is expressed by the second termination criterion.

A.5 Asymptotic consistency

The fundamental measure of correctness of a classification rule is its unconditional error rate, R_n, in the limit of an infinite training sample. A rule is Bayes risk consistent or efficient if

$$\lim_{n\to\infty} R_n = R^* \tag{A.61}$$

where R^* is the Bayes error rate or Bayes probability of error. This is the optimal error rate achievable and can occur only if the class conditional densities are completely specified. Here. I will argue that marginal maximum entropy partitioning is asymptotically Bayes optimal by showing that with infinite recursion, it provides consistent density estimates.

Definition 16 *A refinement, $\{E_{n+1}\}$, of a set $\{E_n\}$ is defined by the following property.*

$$\textit{For every } E_i \in \{E_{n+1}\} \;\exists\; \textit{exactly one } F_i \in \{E_n\} \textit{ such that } E_i \subseteq F_i \tag{A.62}$$

Alternately, we may interpret each F_i as being refined into Q^d subsets E_j, such that $F_i = \bigcup_{j=1}^{Q^d} E_j$.

We need to verify that marginal maximum entropy partitioning (MMEP) does in fact cover all the data.

Proposition 6 *Given any sample space $\Omega \subset \Re^d$ populated with N data points, $\mathbf{x} \in \Re^d$, marginal maximum entropy partitioning will cover each $\mathbf{x}$ with exactly one d-dimensional cell.*

Proof:

Suppose that we are given a sample space $\Omega \subset \Re^d$ and marginal maximum entropy partitioning is applied to each dimension, j, $j = 1 \ldots d$. Let Q be the number of partitions formed in each dimension. It is clear that in each dimension, there will be Q disjoint subsets $E_{j1}, E_{j2}, \ldots E_{jQ}$ of the original data Ω, i.e., $\Omega = \bigcup_{i=1}^{Q} E_{ji}$, $\forall j$. Therefore, every $\mathbf{x}$ will be contained in exactly 1 subset of each marginal partition.

$$\sum_{i=1}^{Q} I_{E_{ji}}(\mathbf{x}) = 1 \quad \forall j \tag{A.63}$$

Here, $I_{E_{ji}}(\mathbf{x})$ is the indicator function of the cell E_{ji} as defined previously. Next note that the intersection of d subsets, one from each of the marginal partitions, will yield a unique subset, i.e., $E_{\mathcal{J}} = \bigcap_{j=1}^{d} E_{j\mathcal{K}(j)}$. Here $\mathcal{K}(j) \in \{1, \ldots Q\}$ selects one subset of each marginal partition and $\mathcal{J} = \sum_{j=1}^{d} \mathcal{K}(j)10^{d-j}$ defines the composition of the resulting subset. The uniqueness of $E_{\mathcal{J}}$ is due to the fact that for each j, the sets E_{ji} are disjoint. The disjointness property can be expressed as,

$$E_{\mathcal{J}} \bigcap (\Omega \setminus E_{j\mathcal{K}(j)}) = \emptyset \quad \forall j \tag{A.64}$$

This means that no other intersection of marginal partition subsets can contain points in $E_{\mathcal{J}}$ and therefore $E_{\mathcal{J}}$ is uniquely determined. The uniqueness is summarized by,

$$\sum_{i=1}^{Q^d} I_{E_i}(\mathbf{x}) = \prod_{j=1}^{d} \left(\sum_{k=1}^{Q} I_{E_{jk}}(\mathbf{x}) \right) = 1 \qquad \forall \mathbf{x} \in \Omega = \bigcup_{i=1}^{Q^d} E_i \tag{A.65}$$

◇

So we have established that when applied to a given sample space, MMEP will in fact cover each data point uniquely, with a d-dimensional cell.

Next we characterize the behaviour of the estimate as the number of available samples becomes very large.

Proposition 7 *Suppose MMEP is recursively applied to a sample space $\Omega \subset \Re^d$ of N points. If $N \to \infty$ then,*

1. *The maximum number of allowable recursions, r_{max}, increases at the rate $\ln N$.*

2. *The ratio $\frac{n_r}{N} \to 0$. where n_r is the number of points in the cell containing $\mathbf{x}$ at the r^{th} recursion.*

3. *The volume of cells, $V_N \to 0$, but $NV_N \to \infty$*

4. *For each data point $\mathbf{x}$, there will be a cell center $\mathbf{c}$ such that, $|\mathbf{c} - \mathbf{x}| < \epsilon$, $0 < \epsilon << 1$*

Proof:

To prove the first item, we find a lower bound on the maximum number of recursions r_{max}. Let ξ represent the minimum allowable number of points per cell and let Q_i

be the number of partitions at the i^{th} level of recursion. For one particular subspace of the sample space, we may write,

$$\xi = \prod_{i=1}^{r_{max}} \left(\frac{1}{Q_i^d} \right) N \tag{A.66}$$

We note that if $Q_{min} = \min_i Q_i$, then

$$\frac{1}{Q_1} \frac{1}{Q_2} \cdots \frac{1}{Q_{r_{max}}} \leq \left(\frac{1}{Q_{min}} \right)^{r_{max}} \tag{A.67}$$

Using this result in (A.66) and taking logarithms yields,

$$\ln \xi \leq \ln N + d r_{max} \ln \left(\frac{1}{Q_{min}} \right) \tag{A.68}$$

Rearranging, we arrive at a lower bound for r,

$$r_{max} \geq A \ln N - B \tag{A.69}$$

where $A = \frac{1}{d \ln Q_{min}}$ and $B = A \ln \zeta$. As A and B are constants with respect to N, then r_{max} grows approximately as $\ln N$. Therefore, $r_{max} \to \infty$ as $N \to \infty$.

$\diamond$

Consider a sequence of cells, E_k, $k = 1 \ldots r$, containing a point $\mathbf{x}$, i.e. $\mathbf{x} \in E_r \subseteq E_{r-1} \subseteq \ldots \subseteq E_1$. By repeated application of Proposition 6 to each level of recursion, we know such a sequence of refinements exist. Let n_r be the number of points in

E_r. For fixed N, note that

$$n_r = \frac{n_{r-1}}{Q_r^d} = \frac{n_{r-2}}{Q_r^d Q_{r-1}^d} = \cdots = \frac{N}{\prod_{i=1}^r Q_i^d} \quad (A.70)$$

Rearranging,

$$\frac{n_r}{N} = \frac{1}{\prod_{i=1}^r Q_i^d} \quad \text{where } Q_i > 1 \ \forall i \quad (A.71)$$

We know that $r \to \infty$ and thus $\frac{n_r}{N} \to 0$ as $N \to \infty$.

◇

To prove that the volume decreases to 0, we note that in a particular subspace,

$$V_N = V_{Global} \prod_{i=1}^{r} \frac{1}{Q_i^d} \quad (A.72)$$

where V_{Global} is the constant global volume of the space occupied by the data. As $Q_i > 1$ for all i, and $r \to \infty$ as $N \to \infty$, therefore $\lim_{N \to \infty} V_N = 0$. However, we need to show that V_N decreases slower than $1/N$. Let $Q_i = (\frac{n_i}{\xi})^{1/d}$, where ξ denotes the minimum number of points per cell. This is a crude upper bound for Q_i. Thus we may write the volume as,

$$V_N = V_{Global} \prod_{i=1}^{r(N)} \frac{\xi}{n_i} \quad (A.73)$$

where we have written $r(N)$ to indicate that r is a function of N. Multiplying both sides by N, we obtain.

$$NV_N = V_{Global}\xi \prod_{i=1}^{r(N)} \frac{N}{n_i} \quad (A.74)$$

We know however that $\lim_{N\to\infty} \frac{n_i}{N} = 0$ and therefore we conclude that $NV_N \to \infty$ as $N \to \infty$.

◊

Consider again the sequence of cells E_k, $k = 1 \ldots r$, that contain $\mathbf{x}$. Let $\mathbf{c}_k$ be the centers of E_k. Let h_{kj} be the length of the j^{th} dimension of E_k,

$$h_{kj} = h_{1j} \prod_{i=1}^{r-1} \frac{1}{Q_i}, \quad Q_i > 1 \tag{A.75}$$

where h_{1j} is the original length, prior to any recursion. Define the diagonal of the cell E_k as $diag(E_k) = \sqrt{\sum_{j=1}^{d} h_{kj}^2}$. For $\mathbf{x} \in E_k$,

$$||\mathbf{x} - \mathbf{c}_k|| \leq \ diag(E_k) \tag{A.76}$$

where $||\cdot||$ is the Euclidean norm. Since $Q_i > 1$, $h_{kj} \to 0$ and therefore $diag(E_k) \to 0$ as $r \to \infty$. We conclude then that there exists an R such that for $r > R$ and $0 < \epsilon << 1$, $||\mathbf{x} - \mathbf{c}_r|| < \epsilon$. In other words, $\mathbf{c}_r$ can be arbitrarily close to $\mathbf{x}$.

◊

We are now ready to state the results describing the asymptotic properties of the pdf estimate.

Proposition 8 *Suppose we have a sample space $\Omega \in \Re^d$ from which we independently draw N samples $\mathbf{x}$ according to the probability law $f(\mathbf{x})$. The pdf estimate,*

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{K} I_{E_i}(\mathbf{x}) \frac{n_{E_i}}{NV_{E_i}} \tag{A.77}$$

based upon the final set of cells, $\{E_i\}$, $i = 1 \ldots K$, of a sequence of recursive maximum entropy partitions of Ω is asymptotically unbiased, i.e., for $\mathbf{x} \in \Omega$,

$$E[\hat{f}(\mathbf{x})] \longrightarrow f(\mathbf{x}) \quad \textit{as } N \to \infty \tag{A.78}$$

Here, $E[\cdot]$ denotes expectation over different random samplings.

Proof:

For a test point $\mathbf{x}_{test}$ contained in the cell E_i of the final partition,

$$\begin{aligned} E[\hat{f}(\mathbf{x}_{test})] &= E[\hat{f}_{E_i}] && \text{(A.79)} \\ &= E[\frac{n_i}{NV_i}] && \text{(A.80)} \\ &= \frac{P}{V_i} && \text{(A.81)} \end{aligned}$$

In the last equation we have exploited the fact that the number of points n_i falling within the cell E_i is binomially distributed with probability $P = \int_{E_i} f(\mathbf{x})d\mathbf{x}$, i.e., $n_i \sim Bin(N, P)$. Hence, $E[n_i] = NP$.

Assuming that $f(\mathbf{x})$ is continuous and bounded on E_i and that E_i is connected, then we may apply the mean value theorem to evaluate P. In other words, $\exists \zeta \in E_i$ such that

$$\int_{E_i} f(\mathbf{x})d\mathbf{x} = f(\zeta)V(E_i) \tag{A.82}$$

where $V(E_i) = \int_{E_i} d\mathbf{x}$. Employing this result in (A.81) yields,

$$E[\hat{f}(\mathbf{x}_{test})] = f(\zeta) \quad \zeta \in E_i \tag{A.83}$$

Now let $\mathbf{c}_i$ be the centre of E_i. By Proposition 7, we know that as $N \rightarrow \infty$,

$$||\zeta - \mathbf{c}_i|| < \epsilon_1 \qquad \text{(A.84)}$$

$$||\mathbf{x}_{test} - \mathbf{c}_i|| < \epsilon_2 \qquad \text{(A.85)}$$

where $0 < \epsilon_1, \epsilon_2 << 1$. To relate ζ and $\mathbf{x}_{test}$ we make use of the triangle inequality,

$$||\zeta - \mathbf{x}_{test}|| \leq ||\zeta - \mathbf{c}_i|| + ||\mathbf{x}_{test} - \mathbf{c}_i|| \qquad \text{(A.86)}$$

$$< \epsilon_1 + \epsilon_2 \qquad \text{(A.87)}$$

Hence as $N \rightarrow \infty$, $||\zeta - \mathbf{x}_{test}|| \rightarrow 0$, implying that $\zeta \rightarrow \mathbf{x}_{test}$. This leads to the conclusion that as $N \rightarrow \infty$,

$$E[\hat{f}(\mathbf{x}_{test})] \rightarrow f(\mathbf{x}_{test}) \qquad \text{(A.88)}$$

◊

Proposition 9 *The pdf estimate (A.77) when applied under the conditions as in Proposition 8, has asymptotically vanishing variance, i.e.,*

$$Var[\hat{f}(\mathbf{x})] \longrightarrow 0 \quad \textit{as } N \rightarrow \infty \qquad \text{(A.89)}$$

Proof:

For a test point $\mathbf{x}$ that falls inside a cell E_i,

$$\begin{aligned}
Var[\hat{f}(\mathbf{x})] &= E[\hat{f}(\mathbf{x})^2] - E[\hat{f}(\mathbf{x})]^2 && \text{(A.90)}\\
&= E\left[\frac{n_i^2}{N^2V_N^2}\right] - E\left[\frac{n_i}{NV_N}\right]^2 && \text{(A.91)}\\
&= \frac{1}{N^2V_N^2}\left(E[n_i^2] - E[n_i]^2\right) && \text{(A.92)}\\
&= \frac{1}{NV_N^2}P(1-P) && \text{(A.93)}
\end{aligned}$$

In the last inequality, we again have used the fact that $n_i \sim Bin(N, P)$ and thus $Var[n_i] = NP(1-P)$. Again if we assume that $f(\mathbf{x})$ is bounded and continuous over E_i and E_i is connected then we can apply the mean value theorem. Substituting, $P = f(\zeta)V_N$, $\zeta \in E_i$, in (A.93), we obtain

$$Var[\hat{f}(\mathbf{x})] = \frac{f(\zeta)}{NV_N} - \frac{f(\zeta)^2}{N} \qquad \text{(A.94)}$$

By Proposition 7 we know $NV_N \to \infty$ as $N \to \infty$ and thus the variance vanishes as $N \to \infty$.

◇

By Markoff's Theorem [90, p.212] we know that asymptotic unbiasedness (Proposition 8) and vanishing asymptotic variance (Proposition 9) imply the mean square convergence of $\hat{f}$ to f, i.e.

$$E[(\hat{f}(\mathbf{x}) - f(\mathbf{x}))^2] \longrightarrow 0 \qquad \text{as } N \to \infty \qquad \text{(A.95)}$$

By Tchebycheff's Inequality [90, p.113] we know that mean square convergence implies convergence in probability.

Since the recursive partitioning process is applied independently for each class, the above results imply that each class density estimate converges in probability to the respective true density.

A.6 Scale invariance

We verify Equations (5.13) and (5.14). For the sake of clarity, variables will be written explicitly in terms of events. By the frequency equalization tendency of MMEP, we see that along the k^{th} dimension of Ω', the partition points will occur at

$$a_1' = \gamma_1 a_1,\ a_2' = \gamma_2 a_2, \ldots, a_k' = \gamma_k a_d \tag{A.96}$$

where $a_1, a_2, \ldots, a_d$ are the partition points along the k^{th} dimension of Ω. Extending this argument to every dimension, $k = 1, \ldots, d$, we immediately see that event volumes are related as,

$$v(E_j') = \Gamma v(E_j) \tag{A.97}$$

where $\Gamma = \prod_{k=1}^{d} \gamma_k$. The frequency equalization tendency also ensures that event frequencies are invariant,

$$n(E_j) = n(E_j') \tag{A.98}$$

This result directly verifies Equation (5.13). To verify Equation (5.14), simply substitute the density definition (5.6) into both sides of (5.14). Simplifying we arrive at,

$$\frac{n(E_i)}{v(E_i)}\frac{v(E_j)}{n(E_j)} = \frac{n(E_i')}{v(E_i')}\frac{v(E_j')}{n(E_j')} \tag{A.99}$$

Using the relations, (A.97) and (A.98), the result (5.14) follows immediately.

A.7 Concentration: Justification for projection

In order to prove proposition 2, we need to show that each component of $\nabla_{x_i x_j}\hat{f}(\mathbf{x})$ is bounded by a non-negative integrable function defined on $\Re^d$ [6]. Due to the symmetrical form of the kernels $\psi_k(\mathbf{x})$. the partial derivatives of $\hat{f}(\mathbf{x})$ with respect to any of the variables are structurally equivalent. Thus, we only need to verify that,

$$\left|\frac{\partial \hat{f}(\mathbf{x})}{\partial x_i}\right| \leq g(\mathbf{y}) \tag{A.100}$$

where $\mathbf{y} = \{x_1, \ldots, x_j, \ldots, x_d\}$, $j \neq i$. Here, $g(\mathbf{y})$ is a non-negative integrable function of $\{x_j,\ j \neq i\}$. The above inequality must hold for all values of $\mathbf{x} = \{x_1, \ldots, x_d\}$. Since $\hat{f}(\mathbf{x})$ is a finite weighted sum of the kernels $\psi_k(\mathbf{x})$, $k = 1, \ldots, N$, it suffices to show that the partial derivative of a single kernel with respect to x_i is bounded, i.e.,

$$\left|\frac{\partial \psi_k(\mathbf{x})}{\partial x_i}\right| \leq g(\mathbf{y}) \tag{A.101}$$

We now proceed with the verification of (A.101). Recall that the k^{th} kernel has the form,

$$\psi_k(\mathbf{x}) = A\exp(-\frac{1}{2}(\mathbf{x}-\mu_k)'\Sigma_k^{-1}(\mathbf{x}-\mu_k)) \tag{A.102}$$

where $A = \frac{1}{(2\pi)^{d/2}\Delta(\Sigma)^{1/2}}$ is a constant for the given kernel. Let us first write the exponent $(\mathbf{x}-\mu_k)'\Sigma_k^{-1}(\mathbf{x}-\mu_k)$ of the kernel $\psi_k(\mathbf{x})$ explicitly. Denote the elements of Σ_k^{-1} as α_{ij} and recall that Σ_k^{-1} is symmetric. The following decomposition is true for any i, $i = 1, \ldots, d$.

$$\begin{aligned}(\mathbf{x}-\mu_k)'\Sigma_k^{-1}(\mathbf{x}-\mu_k) &= \alpha_{ii}(x_i-\mu_{ki})^2 + \sum_{\substack{j=1\\ j\neq i}}^{d} \alpha_{ij}(x_i-\mu_{ki})(x_j-\mu_{kj}) \qquad (A.103)\\ &+ \sum_{\substack{j=1\\ j\neq i}}^{d} \alpha_{jj}(x_j-\mu_{kj})^2 + \sum_{\substack{l=1\\ l\neq i}}^{d}\sum_{\substack{m=1\\ m\neq l}}^{d} \alpha_{lm}(x_l-\mu_{kl})(x_m-\mu_{km})\\ &\stackrel{\text{def}}{=} T_1+T_2+T_3+T_4\end{aligned}$$

Basically the terms have been separated into 2 groups. The first 2 terms are the perfect square and cross-product terms that contain x_i. The last 2 terms are the perfect square and cross-product terms which do not contain x_i. For the sake of brevity, I will denote these terms respectively as, T_1, T_2, T_3 and T_4 and their sum as $T = T_1 + T_2 + T_3 + T_4$.

Consider now the absolute value of the partial derivative of $\psi_k(\mathbf{x})$ with respect to x_i,

$$\left|\frac{\partial\psi_k(\mathbf{x})}{\partial x_i}\right| = A\left|\frac{\partial T_1}{\partial x_i} + \frac{\partial T_2}{\partial x_i}\right| e^{-\frac{1}{2}T} \qquad (A.104)$$

$$= A\left| -\alpha_{ii}(x_i-\mu_{ki}) - \frac{1}{2}\sum_{\substack{j=1\\ j\neq i}}^{d}\alpha_{ij}(x_j-\mu_{kj})\right| e^{-\frac{1}{2}T} \qquad (A.105)$$

By the triangle inequality, we have that

$$\left|\frac{\partial\psi_k(\mathbf{x})}{\partial x_i}\right| \leq A\left|\alpha_{ii}(\mu_{ki}-x_i)\right| e^{-\frac{1}{2}T} + \frac{A}{2}\left|\sum_{\substack{j=1\\j\neq i}}^{d}\alpha_{ij}(\mu_{kj}-x_j)\right| e^{-\frac{1}{2}T} \tag{A.106}$$

This serves as an upper bound to (A.105). Hence, we only have to verify that each term is integrable with respect to $\{x_j, j \neq i\}$. Let us consider each term individually.

Although the absolute value signs imply that there are 2 cases to consider in the first term, we will just examine the positive case, $\alpha_{ii}(\mu_{ki}-x_i) > 0$. Treatment of the negative case is similar. Note that $\alpha_{ii}(\mu_{ki}-x_i)$ is independent of x_j and thus is taken outside the integral with respect to $\{x_j, j \neq i\}$. Specifically,

$$\begin{aligned}\int_{-\infty}^{\infty} A\alpha_{ii}(\mu_{ki}-x_i)e^{-\frac{1}{2}T} &= A\alpha_{ii}(\mu_{ki}-x_i)\int_{-\infty}^{\infty} e^{-\frac{1}{2}T} &\text{(A.107)}\\ &= A\alpha_{ii}(\mu_{ki}-x_i)\; 2\int_{\mu_k}^{\infty} e^{-\frac{1}{2}T} &\text{(A.108)}\end{aligned}$$

where the last equality arises from the symmetry of $e^{-\frac{1}{2}T}$ about μ_k. Recall that T is just $(\mathbf{x}-\mu_k)'\Sigma^{-1}(\mathbf{x}-\mu_k)$. Since Σ^{-1} is positive definite, the quadratic form, $(\mathbf{x}-\mu_k)'\Sigma^{-1}(\mathbf{x}-\mu_k) > 0$ and the first term is therefore Riemman integrable.

Now consider the second term. Applying the triangle inequality again, we obtain,

$$\frac{A}{2}\left|\sum_{\substack{j=1\\j\neq i}}^{d}\alpha_{ij}(\mu_{kj}-x_j)\right| e^{-\frac{1}{2}T} \leq \frac{A}{2}\left(\sum_{\substack{j=1\\j\neq i}}^{d}\left|\alpha_{ij}(\mu_{kj}-x_j)\right|\right) e^{-\frac{1}{2}T} \tag{A.109}$$

Again, the absolute value suggests that each term in the summation entertains a positive and negative case. For the same reasons, we can just examine the positive situation. It is not hard to see that the integral of the right-hand side of (A.109) will be finite when integrating over small values of x_j. For large values of x_j, each term in the summation can be approximately written as,

$$\frac{A}{2}\alpha_{ij}(\mu_{kj} - x_j)\exp\left(-\frac{1}{2}(x_j - \mu_{kj})^2\right)\exp\left(-\frac{1}{2}\sum_{\substack{l=1\\l\neq j}}(x_l - \mu_{kl})^2\right) \tag{A.110}$$

since the quadratic terms will dominate. The integral of this expression can be evaluated first with respect to x_j and then with respect to the other variables, $x_l, l \neq j$. We notice that the first part of this expression has the form ye^{y^2} which is clearly Riemann integrable. The second exponential is of the form e^{z^2} which is also integrable. In conclusion, the expression on the right-hand side of (A.106) is Riemann integrable and hence Lebesgue integrable and is thus a valid upper bound.

Appendix B

Data : Simulation Procedures and Internet Sources

B.1 Simulated data

B.1.1 Concentration: 2 bivariate clusters

The data consisted of 100 uniform background noise points generated in the interval $[0, 8]$. Two clusters were located respectively, at $C_1(X_1, X_2) = (6, 5)$ and $C_2(X_1, X_2) = (2, 2)$. Thirty points were generated around each cluster centre, according to a bivariate uniform random distribution.

B.1.2 Dependency: nonlinear relationship in bivariate data

One-hundred (100) bivariate uniform random points were generated between −5 and 20. Thirty (30) points were sampled from the nonlinear dependency,

$$X_2 = 5\cos\left(\frac{5}{2}X_1\right) + \delta \tag{B.1}$$

where δ is a standard normal random variable.

B.1.3 Linearity: 3 planes

The data is governed by the system of equations,

$$\begin{aligned} x + (y+U) - (z+U) &= 8 \\ 6x - (y+U) - (z+U) &= 1 \\ -2x + (y+U) + (z+U) &= 17 \end{aligned} \tag{B.2}$$

where U is a standard uniform random variate. Twenty-five (25) points were generated on each plane for a total of 75 observations.

B.1.4 Noise tolerance

The defining equation of the underlying trend is,

$$y = 4(x-2)^2. \tag{B.3}$$

Zero-mean Gaussian noise with variance 0.09 was added to each y. A total of 67 points were generated according to this equation from $x = 1$ to $x = 3$. In addition, 30 outliers were added. These outliers were centered around the mean of y and were given a variance of 9.

B.1.5 Noise rejection

For the concentration discovery example, 100 bivariate uniform random data points were generated. Each variable ranged from 0 to 5. For the dependency example, the data is simply 81 univariate Gaussian points centered around 3, with variance 0.09. Finally, the linearity data was generated according to the pair of equations,

$$x = 5\cos(\theta) + 0.1\delta \tag{B.4}$$

$$y = 4\sin(\theta) + cos(\theta) + 0.1\delta \tag{B.5}$$

where δ is the standard normal variable and θ ranged from 0 to π. A total of 32 points were generated at constant intervals within this range of θ.

B.2 Real data sets

The availability of the real life data sets are summarized in Table B.1.

Table B.1: Location of real life data sets

Data set	Location
Thyroid disease	http://www.ics.uci.edu/ mlearn/MLSummary.html
Chemical and overt diabetes	http://lib.stat.cmu.edu/datasets/Andrews
PIMA diabetes	http://www.ics.uci.edu/ mlearn/MLSummary.html
Housing data	http://lib.stat.cmu.edu/datasets/boston
Virus data	http://lib.stat.cmu.edu/datasets/prnn
Lipid data	D.W. Scott [104]
Environment data	http://lib.stat.cmu.edu/datasets/visualizing.data

Bibliography

[1] R. Agrawal, H. Mannila, R. Srikant, H. Toivonon, and A. I. Verkamo. Fast discovery of association rules. In U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, *Advances in Knowledge Discovery and Data Mining*, chapter 12, pages 307–328. AAAI Press/The MIT Press, 1996.

[2] A. Agresti. *Categorical data analysis.* Wiley, New York, 1990.

[3] T.W. Anderson. Some nonparametric multivariate procedures based on statistically equivalent blocks. In P.R. Krishnaiah, editor, *Multivariate Analysis*, pages 5–28. Academic Press, 1966.

[4] D.F. Andrews. Plots of high-dimensional data. *Biometrics*, 28:125–136, 1972.

[5] R. Andrews, R. Cable, J. Diederich, S. Geva, M. Golea, R. Hayward, C. Ho-Stuart, and A. Tickle. Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge-based Systems*, 8(6):373–89, 1995.

[6] T.M. Apostol. *Mathematical Analysis.* Addison-Wesley Publishing Company, 2 edition, 1974.

[7] D. Asimov. The grand tour: A tool for viewing multidimensional data. *SIAM Journal on Scientific and Statistical Computing*, 6:128–143, 1985.

[8] S. Avner. Extraction of comprehensive symbolic rules from a multilayer perceptron. *Engineering Applications*, 9(2):137–43, 1996.

[9] V. Barnett. *Outliers in statistical data.* John Wiley & Sons, Chichester, 3 edition, 1994.

[10] A. Basilevsky. *Applied Matrix Algebra in the Statistical Sciences.* North Holland, 1983.

[11] J.V. Basmajian and D.J. Deluca. *Muscles alive: their functions revealed by electromyography.* Williams and Wilkins, Baltimore, 1985.

[12] D. Beasley, D.R. Bull, and R.R. Martin. An overview of genetic algorithms: Part 2 research techniques. *University Computing*, 15(4):170–181, 1993.

[13] R.E. Bellman. *Adaptive Control Processes: a guided tour.* Princeton University Press, 1961.

[14] D.A. Belsley, E. Kuh, and R.E. Welsch. *Regression diagnostics.* John Wiley and Sons, 1980.

[15] P.H. Bennett. Diabetes mellitus in American (PIMA) Indians. *The Lancet*, 2:125–128, 1971.

[16] J.C. Bezdek. A convergence theorem for fuzzy data clustering algorithms. *IEEE Transactions on pattern analysis and machine intelligence*, 2(1):1–8, 1980.

[17] J.C. Bezdek. Fuzzy models for pattern recognition. In J.C. Bezdek and S.K. Pal, editors, *Fuzzy models for pattern recognition: methods that search for structure in data*, chapter 1, pages 1–27. IEEE Press, 1992.

[18] C. Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, 1995.

[19] Y.M. Bishop, S.E. Fienberg, and P.W. Holland. *Discrete multivariate analysis.* MIT Press, 1975.

[20] E. Bloedorn and R. S. Michalski. Data-driven constrictive induction in AQ17-DCI: A method and experiments. Technical report, Machine Learning and Inference Lab., Center for Artificial Intelligence, George Mason University, 1991.

[21] S.M. Bruntz, W.S. Cleveland, B. Kleiner, and J.L. Warner. The dependence of ambient ozone on solar radiation, wind, temperature, and mixing height. In *Symposium on Atmospheric Diffusion and Air Pollution*, pages 125–128, Boston, 1974. American Meteorological Society.

[22] R. Cao. A comparative study of several smoothing methods in density estimation. *Computational statistics and data analysis*, 17:153–176, 1994.

[23] K.C.C. Chan and A.K.C. Wong. Apacs: A system for the automatic analysis and classification of conceptual patterns. *Computational Intelligence*, 6:119–131, 1990.

[24] D.K.Y. Chiu, B. Cheung, and A.K.C. Wong. Information synthesis based on hierarchical maximum entropy discretization. *Journal of Experimental and Theoretical Artificial Intelligence*, 2:117–129, 1990.

[25] D.K.Y. Chiu and A.K.C. Wong. Synthesizing knowledge: A cluster analysis approach using event covering. *IEEE Transactions on Systems, Man and Cybernetics*, 16(2):251–259, 1986.

[26] R.A. Christensen. *Log-linear models.* Springer-Verlag, 1990.

[27] R.A. Christensen. *Plane answers to complex questions : the theory of linear models.* Springer-Verlag, 2 edition, 1996.

[28] A. Ciampi, C.H. Chang, S.A. Hogg, and S. McKinney. Recursive partition: A versatile method for exploratory data analysis in biostatistics. In I.B. Mac Neil and G.J. Umphrey, editors, *Biostatistics*, volume 5, pages 23–50. Dordrecht, 1987.

[29] P. Clark and T. Niblett. The cn2 induction algorithm. *Machine Learning*, 3:261–283, 1989.

[30] W.S. Cleveland. *Visualizing data.* Hobart Press, Summit,New Jersey, 1993.

[31] W.S. Cleveland, E. Grosse, and W.M. Shyu. Regression by local fitting: Methods, properties and computational algorithms. *Journal of Econometrics*, 37:87–114, 1988.

[32] W.S. Cleveland and M.E. McGill, editors. *Dynamic Graphics for Statistics.* Wadsworth and Brooks/Cole, Pacific Grove CA, 1988.

[33] D. Coomans, I. Broeckaert, M. Jonckheer, and D.L. Massart. Comparison of multivariate discrimination techniques for clinical data. *Methods of Information in Medicine*, 22:93–101, 1983.

[34] C. Cox. An elementary introduction to maximum likelihood estimaton for multinomial models : Birch's theorem and the delta method. *The American Statistician*, 38(4):283–287, 1984.

[35] N.A.C. Cressie. *Statistics for Spatial Data.* John Wiley and Sons, 1991.

[36] A.P.A. daSilva, V.H. Quintana, and G.K.H. Pang. Solving data acquisition and processing problems in power systems using a pattern analysis approach. *IEE Proceedings-C*, 138(4), 1991.

[37] J.E. Dennis and V. Torczon. Direct search methods on parallel machines. *SIAM Journal of Optimization*, 1990.

[38] L.P. Devroye. A universal k-nearest neighbor procedure in discrimination. In *Proceedings of the IEEE Computer Society Conference on Pattern Recognition and Image Processing*, pages 142–147, 1978.

[39] T.G. Dietterich, H. Hild, and G. Bakiri. A comparison of id3 and backpropagation for english text-to-speech matching. *Machine Learning*, 18:51–80, 1995.

[40] M. Dolson. Discriminative nonlinear dimensionality reduction for improved classification. *International Journal of Neural Systems*, 5(4):313–333, 1994.

[41] R.O. Duda and P.H. Hart. *Pattern Classification and Scene Analysis.* John Wiley and Sons, 1973.

[42] B. Efron. *An introduction to the bootstrap.* Chapman & Hall, New York, 1993.

[43] B.S. Everitt. *The analysis of contingency tables.* Wiley, New York, 1977.

[44] A. Famili, W. Shen, R. Weber, and E. Simoudis. Data preprocessing and intelligent data analysis. *Intelligent Data Analysis*, 1(1):http://www.elsevier.com/locate/ida, 1996.

[45] C. Fauquet, D. Desbois, D. Fargette, and G. Vidal. Classification of furoviruses based on the amino acid composition of their coat proteins. In *Viruses with Fungal Vectors*, pages 19–38. Association of Applied Biologists, Edinburgh, 1988.

[46] R.N. Forthofer. *Public progam analsis: a new categorical data approach.* Lifetime Learning Publications, 1981.

[47] J.H. Friedman. A recursive partitioning decision rule for nonparametric classification. *IEEE Transactions on Computers*, pages 404–408, 1977.

[48] J.H. Friedman and J.W. Tukey. A projection pursuit algorithm for exploratory data analysis. *IEEE Transactions on Computers*, C-29(9):881–889, 1974.

[49] K.S. Fu. *Applications of Pattern Recognition.* CRC Press, Boca Raton, Florida, 1982.

[50] C.L. Giles and C.W. Omlin. Extraction, insertion and refinement of symbolic rules in dynamically driven recurrent networks. *Connection Science*, 5(3 and 4):307–328, 1993.

[51] R. Gnanadesikan and J.R. Kettenring. Robust estimates, resiudals and outlier detection with multiresponse data. *Biometrics*, 28:81–124, 1972.

[52] D.E. Goldberg. *Genetic algorithms in search, optimization and machine learning.* Addison-Wesley, 1989.

[53] C. Goodall. Examining residuals. In D.C. Hoaglin, F. Mosteller, and J.W. Tukey, editors, *Understanding Robust and Exploratory Data Analysis*, pages 211–243. John Wiley and Sons, Inc., 1983.

[54] L. Gordon and R.A. Olshen. Asymptotically efficient solutions to the classification problem. *The Annals of Statistics*, 6(3):515–533, 1978.

[55] H.A. Guevenir and I. Sirin. A genetic algorithm for classification by feature partitioning. In *Proceedings of the 5th International Conference on Genetic Algorithms*, pages 543–8, 1993.

[56] S. Haberman. *The analysis of qualitative data*, volume 1. Academic Press, 1978.

[57] S.J. Haberman. *The Analysis of Frequency Data.* University of Chicago Press, 1974.

[58] S.J. Haberman. A warning on the use of chi-square statistics with frequency tables with small expected cell counts. *Journal of the American Statistical Association*, 83(402):555–560, 1988.

[59] D. Harrison and D.L. Rubinfeld. Hedonic housing prices and the demand for clean air. *Journal of environmental economics and management*, 5:81–102, 1978.

[60] E.J Hartman, J.D. Keeler, and J.M Kowalski. Layered neural networks with gaussian hidden units as universal approximations. *Neural Computation*, 2(2):21–215, 1990.

[61] S. Haykin. *Neural Networks.* Macmillan and IEEE Press, 1994.

[62] R.J. Heine. Non-insulin dependent diabetes mellitus: A phenomenon of ageing? *International journal of epidemiology*, 20(1):18–23, 1991.

[63] R.J. Heine and J.M. Mooy. Impaired glucose tolerance and unidentified diabetes. *Post graduate journal of medicine*, 72(844):67–71, 1996.

[64] E.G. Henrichon and K.S. Fu. A nonparametric partitioning procedure for pattern classification. *IEEE Transactions on Computers*, C-18(7):614–624, 1969.

[65] K.W. Hipel and A.I. McLeod. *Time Series Modelling of Water Resources and Environmental Systems.* Elsevier, Amsterdam, 1994.

[66] H.Keen. Diabetes diagnosis. In K.G.M.M. Alberti, R.A. Defronzo, H. Keen, and P. Zimmet, editors, *International Textbook of Diabetes Mellitus*, volume 1, pages 19–30. John Wiley and Sons, 1992.

[67] D.C. Hoaglin, F. Mosteller, and J.W. Tukey, editors. *Understanding Robust and Exploratory data analysis.* John Wiley, NEW YORK, 1983.

[68] P.J. Huber. Projection pursuit (with discussion). *Annals of Statistics*, 13:435–525, 1985.

[69] B. Hudgins, P. Parker, and R.N. Scott. A neural network classifier for multifunction myoelectric control. *Proceedings of the IEEE Engineering in Medicine and Biology Conference*, 13:1454–1455, 1991.

[70] A. Inselberg. The plane with parallel coordinates. *The visual computer*, 1:69–91, 1985.

[71] A.K. Jain and R.C. Dubes. *Algorithms for clustering data.* Prentice Hall, Prentice Hall, N.J., 1988.

[72] E.T. Jaynes. Information theory and statistical mechanics I. *Physical Review*, 106(1):620–630, 1957.

[73] W.C. Knowler, D.J. Pettitt, P.J. Savage, and P.H. Bennett. Diabetes incidence in PIMA Indians: contributions of obesity and parental diabetes. *American Journal of Epidemiology*, 113(2):144–156, 1981.

[74] T. Kohonen. *Self-organization and associative memory.* Springer-Verlag, Berlin, 2nd edition. 1988.

[75] T. Kohonen. *Self-organizing maps.* Springer, Berlin, 1995.

[76] B. Kosko. *Neural networks and fuzzy systems.* Prentice Hall, 1992.

[77] H.O. Lancaster. *The Chi-squared Distribution.* John Wiley and Sons, 1969.

[78] A. Lapedes and R.Farber. How neural nets work. In D.Z. Anderson, editor, *Neural Information Processing Systems*, pages 442–456. American Institute of Physics, New York, 1988.

[79] M. Lascurain. *On Maximum Entropy Discretization and its Applications in Pattern Recognition.* PhD thesis, Systems Design Engineering, University of Waterloo, Waterloo, Ontario, Canada, 1983.

[80] H.Y. Lee and H.L. Ong. Visualization support for data mining. *IEEE Expert*, 11(October):69–75, 1996.

[81] E. Levin, N. Tishby, and S.A. Solla. A statistical approach to learning and generalization in layered neural networks. *Proceedings of the IEEE*, 78(10):1568–1574, 1990.

[82] B.F.J. Manly. *Multivariate Statistical Methods.* Chapman and Hall, 1994.

[83] J. Mao and A.K. Jain. Artificial neural networks for feature extraction and multivariate data projection. *IEEE Transactions on Neural Networks*, 6(2):296–317, 1995.

[84] J.S. Marron. A personal view of smoothing and statistics. In W. Hardle and M.G. Schimek, editors, *Statistical theory and computational aspects of smoothing*, pages 1–9. Physica-Verlag, 1996.

[85] T. Masters. *Practical Neural Network Recipes in C++*. Academic Press, 1993.

[86] T. Masters. *Advanced algorithms for neural networks*. John Wiley and Sons, 1995.

[87] W.S. Meisel and D.A. Michalopoulos. A partitioning algorithm with application in pattern classification and the optimization of decision trees. *IEEE Transactions on Computers*, C-22(1):93–103, 1973.

[88] J.P. Myles and D.J. Hand. The multi-class metric problem in nearest neighbour discrimination rules. *Pattern Recognition*, 23(11):1291–1297, 1990.

[89] H. Narazaki, M. Yamamoto, and T. Watanabe. Reorganizing knowledge in neural networks : an explanation mechanism for neural networks in data classification problems. *IEEE Transactions on Systems, Man and Cybernetics:Part B*, 26(1):107–17, 1996.

[90] A. Papoulis. *Probability, Random Variables and Stochastic Processes*. McGraw-Hill Inc., 1991.

[91] J. Park and I.W. Sandberg. Approximation and radial basis function networks. *Neural Computation*, 5(2):305–316, 1993.

[92] P. Parker and R.N. Scott. Signal processing for the multistate myoelectric channel. *Proceedings of the IEEE*, 65:662–674, 1977.

[93] E. Parzen. On estimation of probability density function and mode. *Annals of Mathematical Statistics*, 33:1065–1076, 1962.

[94] A. Patterson and T. Niblett. *ACLS User Manual.* Intelligent Terminals Ltd, 1983.

[95] Sidney C. Port. *Theoretical Probability for Applications.* John Wiley and Sons, 1994.

[96] J.R. Quinlan. Discovering rules by induction from large classes of examples. In D. Michie, editor. *Expert systems in the microelectronic age*, pages 168–201. Edinburgh University Press, Edinburgh, 1979.

[97] J.R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann, 1993.

[98] C.R. Rao. *Linear Statistical Inference and Its Applications.* John Wiley and Sons, 1973.

[99] G.M. Reaven and R. Miller. An attempt to define the nature of chemical diabetes using a multidimensional analysis. *Diabetologia*, 16:17–24, 1979.

[100] B.D. Ripley. *Spatial Statistics.* John Wiley and Sons, 1981.

[101] B.D. Ripley. *Pattern Recognition and Neural Networks.* Cambridge University Press, 1996.

[102] A. Rogers. *Statiscal analysis of spatial dispersion.* Pion Limited, 1974.

[103] J. Sammon. A non-linear mapping for data structure analysis. *IEEE Transactions on Computers*, 18:401–409, 1969.

[104] D.W. Scott. *Multivariate Density Estimation.* John Wiley and Sons, 1992.

[105] D.W. Scott, A.M. Gotto, J. S. Cole, and G. A. Gorry. Plasma lipids as collateral risk factors in coronary artery disease - a study of 371 males with chest pain. *Journal of Chronic Diseases*, 31:337–345, 1978.

[106] R. Setiono and H. Liu. Symbolic representation of neural networks. *Computer*, 29(3):71–77, 1996.

[107] C.E. Shannon. Mathematical theory of communication. *Bell Systems Technical Journal*, 27(3):379–423, 1948.

[108] B.W. Silverman. *Density Estimation for Statistics and Data Analysis.* Chapman and Hall, London, 1986.

[109] J.W. Smith, J.E. Everhart, W.C. Dickson, W.C. Knowler, and R.S. Johannes. Using the adap learning algorithm to forecast the onset of diabetes mellitus. In *Symposium on Computer Applications in Medical Care*, pages 261–265, New York, 1988. IEEE Computer Society Press.

[110] J. Sonquist. *Multivariate model building: The validation of a search strategy.* Institute for Social Research, Ann Arbor, 1970. Univ. of Michigan.

[111] D.F. Specht. Probabilistic neural networks. *Neural Networks*, 3(1):109–118, 1990.

[112] M. Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the royal statistical society*, 36(1):111–147, 1974.

[113] M.J. Symons. Clustering criteria and multivariate normal mixtures. *Biometrics*, 37:35–43, 1981.

[114] B.G. Tabachnick and L.S. Fidell. *Using Multivariate Statistics.* Harper Collins, 1989.

[115] R.A. Tapia and J.R. Thompson. *Nonparametric Probability Density Estimation.* John Hopkins University Press, 1978.

[116] T.Chau and A.K.C. Wong. Pattern discovery by residual analysis and recursive partitioning. *IEEE Data and Knowledge Engineering*, 1997. Submitted.

[117] G. Towell and J. Shavlik. The extraction of refined rules from knowledge based neural networks. *Machine Learning*, 131:71–101, 1993.

[118] G.J.G. Upton and B. Fingleton. *Spatial Data Analysis by Example*, volume 1 and 2. John Wiley and Sons, 1985.

[119] V. Vapnik. *Estimation of dependencies based on empirical data.* Springer-Verlag, 1982.

[120] E. Wan. Neural network classification: A bayesian interpretation. *IEEE Transactions on Neural Networks*, 1(4):303–305, 1990.

[121] P.D. Wasserman. *Advanced methods in neural computing.* Van Nostrand Reinhold, 1993.

[122] E.J. Wegman. Hyperdimensional data analysis using parallel coordinates. *Journal of the American Statistical Association*, 85(411):664–675, 1990.

[123] H. White. Learning in artificial neural networks : A statistical perspective. *Neural Computation*, 1:425–464, 1989.

[124] D.H. Wolpert. Stacked generalization. *Neural networks*, 5(2):241–259, 1992.

[125] A.K.C. Wong, K.C.C. Chan, and D.K.Y. Chiu. Learning sequential patterns for probabilistic inductive prediction. *IEEE Transactions on Systems, Man and Cybernetics*, 24:1524–1547, 1994.

[126] A.K.C. Wong, D.K.Y. Chiu, and B. Cheung. Information discovery through hierarchical maximum entropy discretization. In G. Piatetsky-Shapiro and W.J. Frawley, editors, *Knowledge Discovery in Databases*, pages 125–140. AAAI/MIT Press, 1987.

[127] A.K.C. Wong and D. Wang. DECA: A discrete-valued data clustering algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(4), 1979.

[128] A.K.C. Wong and Y. Wang. High-order pattern discovery from discrete-valued data sets. *IEEE Transactions on Knowledge and Data Engineering*, 1997. *To Appear.*

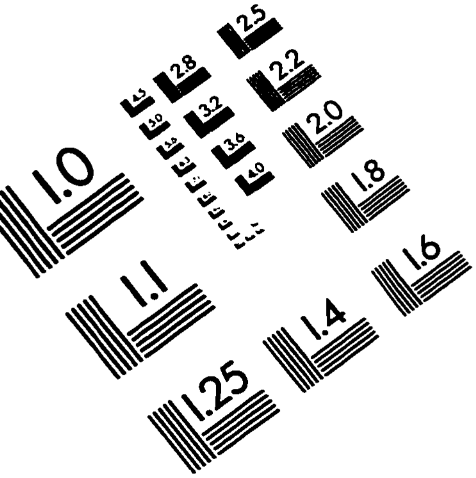
[129] P.W. Wong. *Machine discovery of complex functional forms*. PhD thesis, Systems Design Engineering, University of Waterloo, Waterloo, Ontario, Canada, 1991.

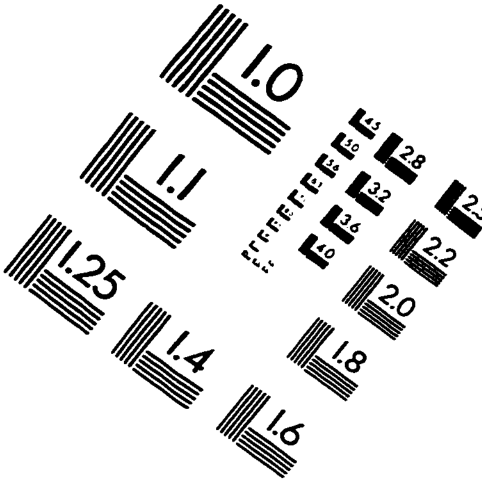
[130] R.R. Yager and D.P. Filev. *Essentials of fuzzy modeling and control.* John Wiley and Sons, 1994.

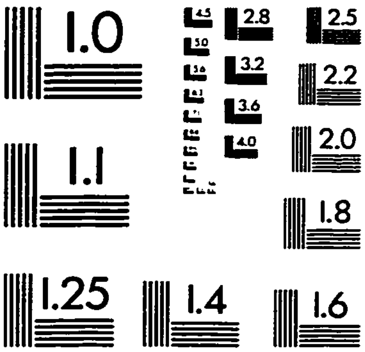
[131] T.Y. Young and K.S. Fu. *Handbook of Pattern Recognition and Image Processing.* Academic Press, 1986.

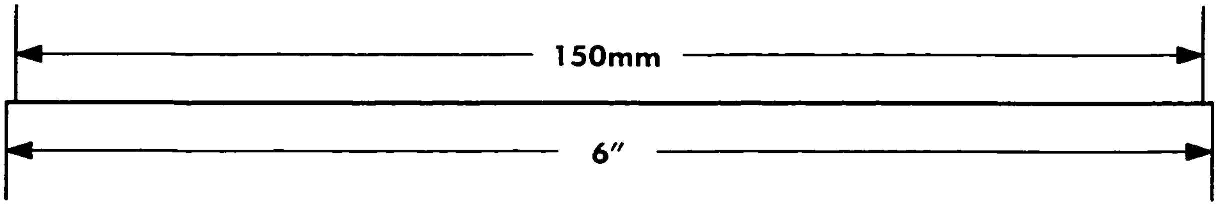
[132] Q. Zhu. Pattern classification in dynamic environments: Tagged feature-class representation and the classifiers. *IEEE Transactions on Systems, Man and Cybernetics*, 19(5):1203–1210, 1989.

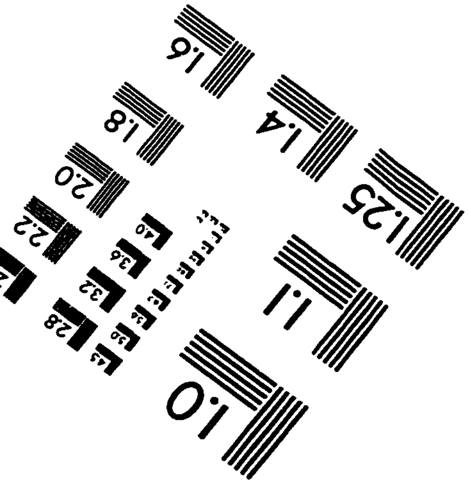
IMAGE EVALUATION TEST TARGET (QA-3)

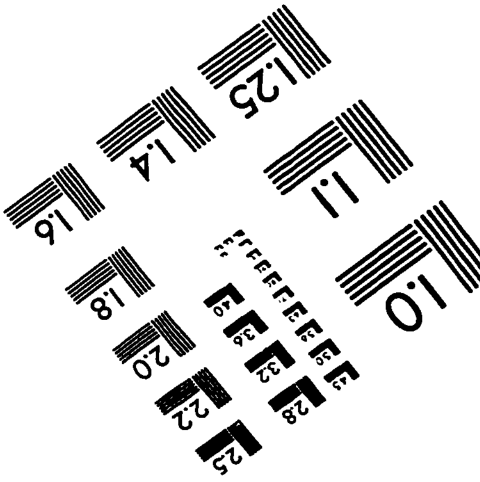












APPLIED IMAGE. Inc
1653 East Main Street
Rochester, NY 14609 USA
Phone: 716/482-0300
Fax: 716/288-5989