Contributions to Metric Methods in Data Mining

A Dissertation Presented

by

Richard A. Butterworth

Submitted to the Office of Graduate Studies, University of Massachusetts Boston, in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2006

Computer Science Department

Contributions to Metric Methods in Data Mining

A Dissertation Presented

by

Richard A. Butterworth

Approved as to style and content by:

Dan A. Simovici, Professor
Chairperson of Committee

William R. Campbell, Associate Professor
Member

Alfred Noël, Associate Professor
Member

Marc Pomplun, Assistant Professor
Member

Dan A. Simovici, Program Director
Computer Science Program

Peter Fejer, Chairperson
Computer Science Department

ABSTRACT

Contributions to Metric Methods in Data Mining

May 2006

Richard A. Butterworth, B.S., Virginia Military Institute
M.S., The University of North Carolina at Chapel Hill
M.S., University of Massachusetts Boston
Ph.D., University of Massachusetts Boston

Directed by Professor Dan A. Simovici

Data mining is an attempt to obtain information from a mass of data that is not easily discernable. Since many data mining techniques are designed for discrete data while frequently the data is actually continuous, there is a great need for reasonable approaches for converting the data from continuous to discrete. This thesis will examine one discretization technique and some of its implications. Clustering data is also an important field of study in data mining. This paper will examine a technique for clustering and uses thereof. The unifying topic is a distance function, called the Barthélemy-Monjardet distance, which will be used for both discretizing and clustering.

To my wife, Elvira, . . .

who encouraged, cajoled, and

inspired me.

ACKNOWLEDGMENTS

I wish to thank Dan Simovici who believed in me even when I didn't and without whose help I never would have finished; William Campbell who served me my first cup of Java; Marc Pomplun who tried to increase my intelligence; Alfred Noël who found and helped correct many more mistakes in this thesis than there should have been; Fred B. Wright who was the first professor to encourage me in graduate school; my colleagues at Massasoit Community College who picked up so many balls I dropped they could supply a sporting goods store; and, of course, my wife, Elvira Heybey Butterworth, who was more responsible for my sticking with it than I.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Preliminaries

'Begin at the beginning,' the King said gravely, 'and go on till you come to the end: then stop.'

– Lewis Carroll, *Alice in Wonderland*

The average Ph.D thesis is nothing but the transference of bones from one graveyard to another.

– J. Frank Dobie, *A Texan in England*

Here's hoping this is not that kind of average.

The questions remain the same. The answers are eternally variable.

– anon

1.1 Overview

The unifying theme of this dissertation is a distance function, the Generalized Barthélemy-Monjardet distance, described in Section 1.7, which is based on a generalization of the concept of entropy. This metric is used in a variety of ways,

such as discretization and clustering. An explanation of how this metric works and how it compares with other methods will be given. This chapter will attempt to give the reader some background as why this is important, and how the distance function was obtained. Later chapters will look at the theory behind the work, examine the various areas upon which these techniques have been brought to bear, and finally look at some experimental results.

1.2 Uses of Metrics in Data Mining

There many branches of data mining that utilize the concept of distance, discretization and clustering are two major ones.

Discretization : Many data mining techniques are designed to work well with discrete data but are overwhelmed, or just don't work, with continuous data. Whereas, continuous data is extremely common, for example, blood pressure and the amount a gene is expressed. To expand on this, continuous data will mean data where the number of values a particular attribute can assume is large relative to the data size. Thus, even if blood pressure is measured only to the nearest whole number, the number of values that blood pressure could take on is relatively large[1].

Clustering : Often datasets contain a huge number of attributes, e.g. datasets involving genes. These large number of attributes can be cumbersome and very time-consuming to analyze. There are also, often, attributes that are essentially noise and will actually detract from attempting to wrest information from the dataset by merely confusing classification. Clustering allows

[1]Large is definitely a fuzzy term here. A rule of thumb would be "large" means that the attribute takes on more than about 40 values.

one to reduce the number of attributes needed to analyze the data. In order to cluster one needs the concept of how close each attribute or collection of attributes is from each other, that is, one needs the concept of distance.

1.3 Building Toward Distance

The distance function used in this dissertation comes directly from the concept of a partition so Section 1.5 will introduce partitions. Succeeding sections will then introduce entropy and generalizations of it since conditional entropy is a measure of the impurity of partitions with each other, introduce distance functions especially those that are derived from entropy, and finally examine the distance function which is the focus of this paper.

1.4 Naming Conventions in This Paper

In this paper the word proposition will be used to indicate concepts which will be proved that are generally known. The word theorem will be used otherwise.

1.5 Partitions

Informally a partition of a set is a breaking up of the set into subsets such that the subsets are mutually exclusive, i.e. the intersection of two different subsets is empty.

A more formal definition of the concept of a partition would be: a *partition* of a non-empty set S is a non-empty collection of non-empty subsets of S, $\pi = \{P_i \mid i \in I\}$, such that $\bigcup\{P_i \mid i \in I\} = S$, and $i, j \in I$, $i \neq j$ implies $P_i \cap P_j = \emptyset$, see

figure 1.1.

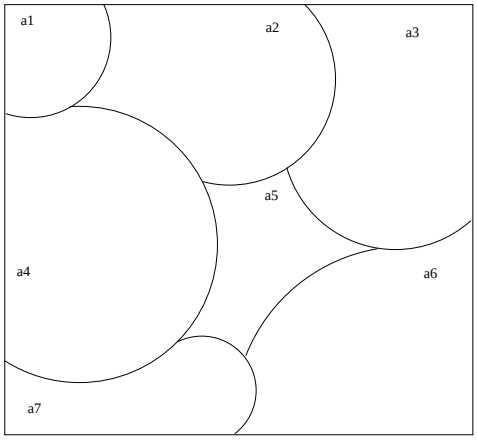


Figure 1.1: Simple Partition of a Set

Here is an example of a partition to attempt to make the various aspects of a partition clear. Suppose there is a dataset concerning people, their weight and blood pressure, see Table 1.1. Notice the data could be partitioned by weight or by blood pressure. You might note that it is already sorted on weight so partitioning on that would be trivial. The partition by weight would be:

$$\{\{s1, s2\}, \{s3, s5, s9, s11, s14, s7, s13\}, \{s4, s6, s8, s10, s12, s15\}\}$$

While the partition by blood pressure would be:

$$\{\{s1, s3, s5, s9, s11, s14\}, \{s7, s4, s6\}, \{s2, s13, s8, s19, s12, s15\}\}$$

Partitions can be generated using SQL by the `select T group by A` command. This partition is denoted π^A of the set of rows of table T.

Partitions have an **order** so they can be organized as a lattice [Gr91], see Figure 1.2, where for partitions A and B $A \leq B$ if each block of A is contained in a block of B.

Weight vs Blood Pressure

subject	weight	bpres
s1	under	low
s2	under	high
s3	norm	low
s5	norm	low
s9	norm	low
s11	norm	low
s14	norm	low
s7	norm	med
s13	norm	high
s4	over	med
s6	over	med
s8	over	high
s10	over	high
s12	over	high
s15	over	high

Table 1.1: Example Data Set

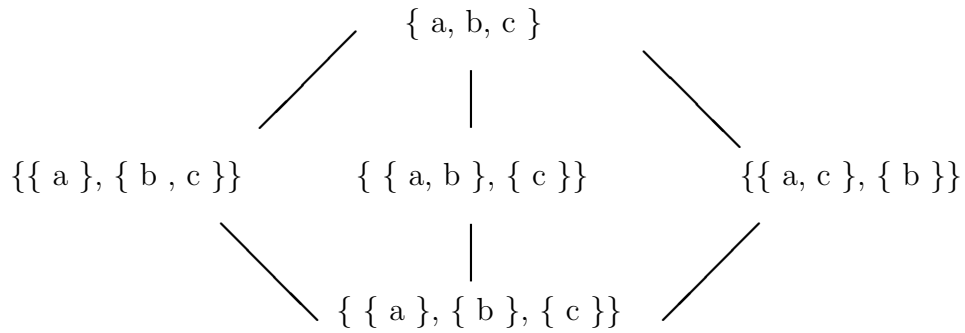


Figure 1.2: Example of the Lattice Structure of Partitions of the set {a, b, c}

The number of partitions of a set with n elements is $B(n)$, the Bell number [Bel34]. The Bell number may represented recursively as,

$$B(n+1) = \sum_{k=0}^{n} \binom{n}{k} B(k),\ B(0) = 1$$

Or, $B(n)$ may be represented as a sum of Stirling numbers of the second kind:

$$B(n) = \sum_{k=0}^{n} S(n,k) \text{ for } n \geq 1$$

$$S(n,k) = kS(n-1,k) + S(n-1,k-1),\ 1 \leq k < n$$

$$S(n,n) = S(n,1) = 1$$

where $S(n,k)$ is the number of ways to partition n objects k at a time.

As noted in the previous section there is interest not only in the basic idea of a partition also in comparing different partitions of the same dataset. This leads to the *trace of the partition*. For a subset L of S the *trace of the partition* π on the set L is the partition

$$\pi_L = \{P_i \cap L \mid 1 \leq i \leq k \text{ and } P_i \cap L \neq \emptyset\}$$

see figure 1.3 or look back at table 1.1 and notice that the trace of partition of blood pressure on normal weight is

$$\{\{s3, s5, s9, s11, s14\}, \{s7\}, \{s13\}\}$$

.

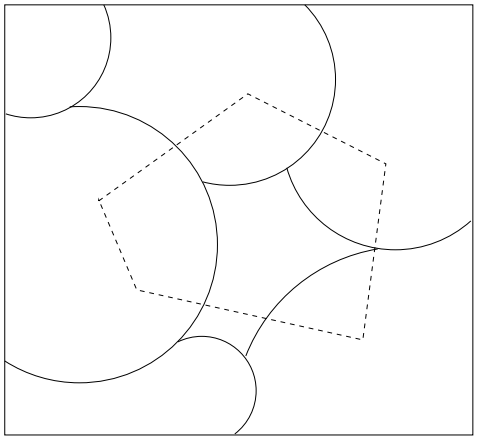


Figure 1.3: Trace of a Partition

1.6 Entropy

The distance function exploited in this paper comes from the concept of entropy which was defined and explained by Claude E. Shannon in the field of communications [SW63]. This idea has been used within the field of data mining to express the purity (or impurity) of a partition of a set [Fay91]).

Let T be a set, π a partition of T, and S a subset of T. To evaluate the purity of S relative to π, examine the trace of π_S of the partition π. The set S is π-pure if all its elements belong to the same block B of π. The more blocks of S intersects, the more impure S is relative to π.

This concept is extended to the relative purity of a partition σ with respect to a partition π. The purity of σ is defined in terms of the purity of the blocks of σ relative to π.

For example, suppose you had the table 1.2. Notice it is sorted on weight and, within that sorting, sorted on blood pressure. One can look at the table to get some idea of the purity of partition on blood pressure relative to weight. Notice that most of the normal weighted people have low blood pressure, whereas most of the overweight people have high blood pressure. The Generalized Barthélemy-Monjardet distance function will use a generalization of Shannon entropy to quantify this purity.

Weight vs Blood Pressure

subject	weight	bpres
s1	under	low
s2	under	high
s3	norm	low
s5	norm	low
s9	norm	low
s11	norm	low
s14	norm	low
s7	norm	med
s13	norm	high
s4	over	med
s6	over	med
s8	over	high
s10	over	high
s12	over	high
s15	over	high

Table 1.2: Example Data Set

1.6.1 Shannon Entropy

To be more formal as to what entropy is. If S is a finite set, a partition $\pi = \{B_1, \ldots, B_k\}$ generates a random variable:

$$X_\pi = \begin{pmatrix} 1 & 2 & \cdots & k \\ p_1 & p_2 & \cdots & p_k \end{pmatrix},$$

where $p_i = \frac{|B_i|}{|S|}$, i.e. p_i is the fraction of the number of tuples in the ith block. Entropy measures the dispersion of values of a random variable. Thus the Shannon entropy [SW63] of π may be defined as the entropy of the random variable X_π, i.e.:

$$\mathcal{H}(\pi) = -\sum_{i=1}^{k} p_i \log_2 p_i \tag{1.1}$$

As an example, notice that the entropy of the dataset represented in Table 1.2 partitioned on the attribute weight is:

$$\mathcal{H}(\pi) = -\frac{2}{15} log_2 \frac{2}{15} - \frac{7}{15} log_2 \frac{7}{15} - \frac{6}{15} log_2 \frac{6}{15}.$$

It can be noted that the maximum entropy for a k-valued random variable is obtained when $p_1 = \cdots = p_k = \frac{1}{k}$ and equals $log_2 k$.

Lemma 1.6.1 $\ln x \leq x - 1$ *for all* $x > 0$.

Proof:

Define the function $f : \mathbb{R}_{>0} \longrightarrow \mathbb{R}$ by $f(x) = x - 1 - \ln x$. Its derivative is $f'(x) = 1 - \frac{1}{x}$ and its second derivative is $f''(x) = \frac{1}{x^2}$ for $x > 0$. Thus, since $f'(1) = 0$ and $f''(x) > 0$, a minimum must occur at $x = 1$, i.e. the minimum is $f(1) = 0$. So, $0 \leq x - 1 - \ln x$. Therefore, $\ln x \leq x - 1$ for all $x > 0$. ∎

Proposition 1.6.2 $\mathcal{H}(S)$ *has a maximum of* $log_2 k$ *when* $p_1 = \cdots = p_k = \frac{1}{k}$.

Proof:

Note, since $\ln x \leq x - 1$, then $\ln \frac{1}{x} \geq 1 - x$.

Consider $log_2 k$ and $\mathcal{H}(S) = \sum_{i=1}^{k} -p_i log_2 p_i$

$$\begin{aligned}
\frac{\ln k}{\ln 2} - \mathcal{H}(S) &= \sum_{i=1}^{k} p_i \frac{\ln k}{\ln 2} - \sum_{i=1}^{k} p_i \frac{\ln\left(\frac{1}{p_i}\right)}{\ln 2} \\
&= \frac{1}{\ln 2} \sum_{i=1}^{k} p_i \ln (p_i k) \\
&\geq \frac{1}{\ln 2} \sum_{i=1}^{k} p_i \left(1 - \frac{1}{p_i k}\right) \\
&\geq \frac{1}{\ln 2} \left(\sum_{i=1}^{k} p_i - \frac{1}{k} \sum_{i=1}^{k} \frac{p_i}{p_i}\right) \\
&= \frac{1}{\ln 2} \left(1 - \frac{1}{k}(k)\right) \\
&= 0
\end{aligned}$$

Therefore, $log_2 k \geq \mathcal{H}(S)$. Finally note that

$$\begin{aligned}
\sum_{i=1}^{k} -\frac{1}{k} log_2 \frac{1}{k} &= -\frac{1}{k} \sum_{i=1}^{k} log_2 \frac{1}{k} \\
&= \frac{1}{k} \sum_{i=1}^{k} log_2 k \\
&= log_2 k
\end{aligned}$$

Thus $\mathcal{H}(S)$ has a maximum of $log_2 k$ which occurs when $p_1 = \cdots = p_k = \frac{1}{k}$. ∎

1.6.2 Shannon Conditional Entropy

So far only entropy with a single attribute partitioning the dataset has been discussed. Remember the more interesting question is how the partitions of a dataset by several attributes compare since the ultimate goal is to be able to compare how

a given attribute partitions the dataset versus the target attribute. For example, consider the previous dataset, represented by Table 1.2, it would be informative to compare how blood pressure partitions the dataset with how weight does to see if weight can predict, and how reliably, blood pressure. Conditional entropy will deal with this dilemma.

If π and σ are two partitions in $\mathsf{PART}(S)$, the average impurity of the blocks of σ relative to π is defined to be the *conditional entropy of π relative to σ*:

$$\mathcal{H}(\pi|\sigma) = \sum_{j=1}^{m} \frac{|Q_j|}{|S|}\mathcal{H}(\pi_{Q_j}),$$

where $\sigma = \{Q_1, \ldots, Q_m\}$ and $\mathcal{H}(\pi_{Q_j})$ is the entropy of the block Q_j with respect to π, see figure 1.4. For example, using the dataset from Table 1.2, where π is the partition using blood pressure, results in:

$$\mathcal{H}(\pi_{normal}) = -\frac{5}{7} \cdot \log_2\left(\frac{5}{7}\right) - \frac{1}{7} \cdot \log_2\left(\frac{1}{7}\right) - \frac{1}{7} \cdot \log_2\left(\frac{1}{7}\right).$$

And, $\mathcal{H}(\pi|\sigma)$, where π is the partition by blood pressure and σ is the partition by weight, is

$$\mathcal{H}(\pi|\sigma) = -\frac{2}{15}\mathcal{H}(\pi_{under}) - \frac{7}{15}\mathcal{H}(\pi_{norm}) - \frac{6}{15}\mathcal{H}(\pi_{over})$$

1.6.3 Generalizations of Shannon Entropy

Several authors have introduced generalizations of entropy, see [Dev74, Dar70, HC67]. The common nature of these generalizations has been highlighted in [SJ02], where a unified axiomatization was introduced. In particular, Daróczy suggested a generalization of the Shannon entropy [Dar70] adding a parameter, β. Daróczy's β-entropy for a partition $\pi = \{P_1, \ldots, P_k\} \in \mathsf{PART}(S)$ is:

$$\mathcal{H}_\beta(\pi) = \frac{1}{2^{1-\beta} - 1}\left(\sum_{i=1}^{k}\left(\frac{|P_i|}{|S|}\right)^\beta - 1\right), \beta > 0. \tag{1.2}$$

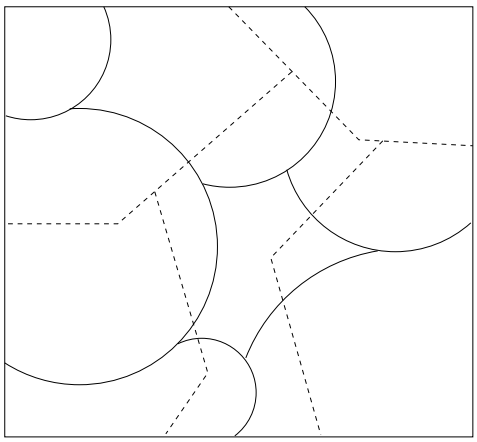


Figure 1.4: Two partitions of a set – solid and dotted lines

Note that this is an extension of Shannon entropy since $\lim_{\beta\to 1}\mathcal{H}_\beta(\pi)$ is the Shannon entropy.

Proposition 1.6.3

$$\mathcal{H}_\beta(\pi) = \frac{1}{2^{1-\beta}-1}\left(\sum_{i=1}^{k}\left(\frac{|P_i|}{|S|}\right)^\beta - 1\right), \beta > 0.$$

is an extension of Shannon entropy, i.e. $\lim_{\beta\to 1}\mathcal{H}_\beta(\pi) = \mathcal{H}(\pi)$.

Proof:

$$\begin{aligned}
\lim_{\beta\to 1}\mathcal{H}_\beta(\pi) &= \lim_{\beta\to 1}\frac{\sum_{i=1}^{k}\left(\frac{|P_i|}{|S|}\right)^\beta - 1}{2^{1-\beta}-1}\text{ , now use l'Hôpital's Rule}\\
&= \lim_{\beta\to 1}\frac{\sum_{i=1}^{k}\left(\frac{|P_i|}{|S|}\right)^\beta \cdot \ln\left(\frac{|P_i|}{|S|}\right)}{-2\cdot 2^{-\beta}\cdot \ln 2}\text{ , simplify}\\
&= \sum_{i=1}^{k} -\left(\frac{|P_i|}{|S|}\right)\log_2\left(\frac{|P_i|}{|S|}\right)\\
&= \mathcal{H}(\pi).
\end{aligned}$$

∎

Daróczy also generalized the concept of conditional entropy. For $\sigma, \pi \in \mathsf{PART}(S)$, where $\pi = \{P_1, \ldots, P_k\}$ and $\sigma = \{Q_1, \ldots, Q_m\}$, the Daróczy's conditional β-entropy $\mathcal{H}_\beta(\pi|\sigma)$ is given by

$$\mathcal{H}_\beta(\pi|\sigma) = \sum_{j=1}^{m} \left(\frac{|Q_j|}{|S|}\right)^\beta \mathcal{H}_\beta(\pi_{Q_j}),$$

and thus, using Equation 1.2 we can obtain an equation which is more efficient to implement:

$$\begin{aligned}
\mathcal{H}_\beta(\pi|\sigma) &= \sum_{j=1}^{m} \left(\frac{|Q_j|}{|S|}\right)^\beta \mathcal{H}_\beta(\pi_{Q_j}) \\
&= \sum_{j=1}^{m} \left(\frac{|Q_j|}{|S|}\right)^\beta \cdot \frac{1}{2^{1-\beta}-1} \cdot \left(\sum_{i=1}^{k} \left(\frac{|P_i \bigcap Q_j|}{|Q_j|}\right)^\beta - 1\right) \\
&= \frac{1}{(2^{1-\beta}-1)|S|^\beta} \left(\sum_{i=1}^{k}\sum_{j=1}^{m} |P_i \cap Q_j|^\beta - \sum_{j=1}^{m} |Q_j|^\beta\right). \qquad (1.3)
\end{aligned}$$

Daróczy's conditional β-entropy is what will be used to create the Generalized Barthélemy-Monjardet distance function.

1.7 Distance Functions

The discussion now turns to distance functions (or metrics) so an introduction as to what a distance function is would be appropriate. A distance function is a function with the following properties:

1. $d(x, y) = 0$ iff $x = y$
2. $d(x, y \geq 0$
3. $d(x, y) = d(y, x)$
4. $d(x, y) \leq d(x, z) + d(z, y)$

Note, a dissimilarity is similar to a metric except that property 4 (the triangle property) need not apply. There are many distance functions that are used in data mining today, such as [WM97] :

Minkowsky : $D(x,y) = (\sum_{i=1}^{m} |x_i - y_i|^r)^{1/r}$

Euclidean : $D(x,y) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2}$

Manhattan : $D(x,y) = \sum_{i=1}^{m} |x_i - y_i|$

Canberra : $D(x,y) = \sum_{i=1}^{m} \left| \frac{x_i - y_i}{x_i + y_i} \right|$

Chebychev : $D(x,y) = \max_{i=1}^{m} |x_i - y_i|$

Quadratic :

$$D(x,y) = (x-y)^T Q (x-y) = \sum_{j=1}^{m} \left(\sum_{i=1}^{m} (x_i - y_i) q_{ji} \right) (x_j - y_j)$$

- Q is a problem-specific positive definite $m \times m$ weight matrix

Mahalanobis : $D(x,y) = [detV]^{1/m} (x-y)^T V^{-1} (x-y)$

- V is the co-variance matrix of $A_1 \ldots A_m$
- A_j is the vector of values for attribute j occurring in the training set instances $1 \ldots n$

Correlation :

$$D(x,y) = \frac{\sum_{i=1}^{m} (x_i - \widehat{x_i})(y_i - \widehat{y_i})}{\sqrt{\sum_{i=1}^{m} (x_i - \widehat{x_i})^2 \sum_{i=1}^{m} (y_i - \widehat{y_i})^2}}$$

- $\widehat{x_i} = \widehat{y_i}$ and is the average value for attribute i occurring in the training set

Chi-square : $D(x,y) = \sum_{i=1}^{m} \frac{1}{sum_i} \left(\frac{x_i}{size_x} - \frac{y_i}{size_y} \right)^2$

- sum_i is the sum of all values for attribute i occurring in the training set
- $size_x$ is the sum of all values in the vector x

Kendall's Rank Correlation :

$$D(x,y) = 1 - \frac{2}{n(n-1)} \sum_{i=1}^{n} \sum_{j=1}^{i-1} sign(x_i - x_j) sign(y_i - y_j)$$

- Definition of $sign(x)$

$$sign(x) = \begin{cases} -1, & \text{if } x < 0 \\ 0, & \text{if } x = 0 \\ 1, & \text{if } 0 < x \end{cases}$$

1.7.1 The Distance Function

Now all the pieces can be combined. The technique to be used in this paper for discretizing and clustering is to measure the dissimilarity between attributes. One way to do that would be to examine the distance between them. López de Màntaras proved that the function $d : \mathsf{PART}(S) \times \mathsf{PART}(S) \longrightarrow \mathbb{R}$ defined by: $d(\pi, \sigma) = \mathcal{H}(\pi|\sigma) + \mathcal{H}(\sigma|\pi)$, where $\mathcal{H}$ is the Shannon entropy is a metric [Man91]. In [SJ03] Simovici and Jaroszewicz showed a generalization of de Màntaras's distance function, i.e. that the function $d_\beta : \mathsf{PART}(S) \times \mathsf{PART}(S) \longrightarrow \mathbb{R}$ given by

$$\begin{aligned} d_\beta(\pi,\sigma) &= \mathcal{H}_\beta(\pi|\sigma) + \mathcal{H}_\beta(\sigma|\pi) \\ &= \frac{1}{(2^{1-\beta}-1)|S|^\beta} \left(\sum_{i=1}^{k} \sum_{j=1}^{m} |P_i \cap Q_j|^\beta - \sum_{j=1}^{m} |Q_j|^\beta + \right. \\ &\quad \left. \sum_{i=1}^{k} \sum_{j=1}^{m} |P_i \cap Q_j|^\beta - \sum_{i=1}^{k} |P_i|^\beta \right) \\ &= \frac{1}{(2^{1-\beta}-1)|S|^\beta} \left(2 \cdot \sum_{i=1}^{k} \sum_{j=1}^{m} |P_i \cap Q_j|^\beta - \sum_{i=1}^{k} |P_i|^\beta - \sum_{j=1}^{m} |Q_j|^\beta \right). \quad (1.4) \end{aligned}$$

is a distance. This is the distance function to be used in the rest of this paper.

CHAPTER 2

Discretization

Es brillig war. Die schlichte Tovern
Wirrten und wimmelten in Waben;
Und aller-mümsige Burggoven
Dis mohmem Räth' ausgraben.
– *Through the Looking Glass*

2.1 Various Discretization Strategies

There are several basic discretization strategies which are standard:

1. binning
   (a) equiwidth
   (b) equidepth
2. cluster analysis
3. natural partitioning
4. entropy-based partitioning

Binning is taking the data and distributing it approximately equally. Equiwidth is dividing the range of the attribute under consideration such that each bin is the

same width. Equidepth is dividing the range such that each bin has approximately the same number of data.

Cluster analysis is examining the data looking for groupings. This is best seen by thinking of the data points as objects in a two-dimensional space and looking to see how they group.

Natural partitioning is dividing the range of the attribute in a 'natural' way. For example, if one was looking at weights, divide the weights into groups of 5 kilograms, e.g. $(40, 45]$, $(45, 50]$, and so on.

Entropy-based partitioning is grouping the data using entropy. The present strategy, to be explained below, will use a variant of this.

2.2 How To Partition Data – Cutpoints and An Example

In Table 1.2 blood pressure was divided into three categories low, medium, and high; weight was divided into under, normal, and over. But how is continuous data aggregated into such discrete groups? One approach is to use the aforementioned generalized entropy to do the job. First potential cutpoints, i.e. appropriate places to divide the tuples, need to be determined.

If T is a table and A is an attribute of T, the set of members of the domain of A that occur under A in T will be referred to as *the active domain of* A; this set is denoted by $\mathsf{adom}_T(A)$, or, if there is no risk of confusion, simply by $\mathsf{adom}(A)$. The partition of the set of tuples of T that corresponds to a partition π of $\mathsf{adom}_T(A)$ is denoted by π_*. A block of π_* consists of all tuples whose A-projections belong to the same block of π.

Discretization of a numeric attribute A involves selecting a set of cutpoints

$S = \{t_1, \ldots, t_\ell\}$ in the active domain of the attribute $\mathsf{adom}(A)$, where $t_1 < t_2 < \cdots < t_\ell$. This set of cutpoints creates a partition $\pi^S = \{Q_0, \ldots, Q_\ell\}$ of $\mathsf{adom}(A)$, where $Q_i = \{a \in \mathsf{adom}(A) \mid t_{i-1} \leq a < t_i\}$ for $0 < i \leq \ell + 1$, where $t_0 = -\infty$ and $t_{\ell+1} = +\infty$. If the set S consists of a single cutpoint t we shall denote π^S simply by π^t. The discretization process consists of replacing each value that falls in the block Q_i of π^S by i for $0 \leq i \leq \ell$. Thus we have gone from continuous data to discrete data which could be viewed as ordered if needed.

The following example should help to clarify how this works.

Example 2.2.1 To assess the influence of physical parameters of individuals for the risk for hypertension data is collected concerning the `height` and `weight` in the table `MEASUREMENTS`. A classifier is constructed based on the training data set shown in Table 2.1.

The active domains of the attributes of this table are:

$$\begin{aligned}
\mathsf{adom}(\texttt{subject}) &= \{s1, \ldots, s15\} \\
\mathsf{adom}(\texttt{weight}) &= \{160, 165, 175, 180, 183, 185, 188, 190, \\
&\quad 190, 198, 200, 202, 205, 210, 212, 228\} \\
\mathsf{adom}(\texttt{height}) &= \{60, 62, 64, 65, 67, 68, 69, 70, 71, 72, 74\} \\
\mathsf{adom}(\texttt{bpres}) &= \{\text{low}, \text{med}, \text{high}\}.
\end{aligned}$$

The attribute `bpres` of this table determines a 3-block partition

$$\pi_{\texttt{bpres}} = \{B_{low}, B_{med}, B_{high}\}$$

of the set of rows (identified here by the subject number), where each block contains

MEASUREMENTS

subject	weight	height	bpres
s1	160	62	low
s2	165	65	high
s3	180	67	low
s4	210	70	med
s5	185	72	low
s6	228	74	med
s7	190	68	med
s8	200	71	high
s9	198	69	low
s10	205	68	high
s11	183	64	low
s12	202	74	high
s13	175	60	high
s14	188	70	low
s15	212	65	high

Table 2.1: Training Data Set

the individuals having the same category of blood pressure readings:

$$\begin{aligned} B_{low} &= \{s_1, s_3, s_5, s_9, s_{11}, s_{14}\} \\ B_{med} &= \{s_4, s_6, s_7\} \\ B_{high} &= \{s_2, s_8, s_{10}, s_{12}, s_{13}, s_{15}\}. \end{aligned}$$

In order to classify individuals based on their risk for hypertension one needs to discretize the attributes `weight` and `height`. The simplest discretization of `weight` involves splitting $\mathsf{adom}(\texttt{weight})$ into two classes $Q_{\leq}^{t_{weight}}$ and $Q_{>}^{t_{weight}}$, defined by:

$$\begin{aligned} Q_{\leq}^{t_{weight}} &= \{w \in \mathsf{adom}(\texttt{weight}) \mid w \leq t_{weight}\} \\ Q_{>}^{t_{weight}} &= \{w \in \mathsf{adom}(\texttt{weight}) \mid w > t_{weight}\}, \end{aligned}$$

where t_{weight} is a cutpoint. The cutpoint t_{weight} should be selected such that the sets of tuples whose weight belong to $Q_{\leq}^{t_{weight}}$ and $Q_{>}^{t_{weight}}$ are as "pure" as possible relative to the partition π_{bpres}. In other words, one needs to choose t_{weight} such that each of these sets is scattered over as few blocks of π_{bpres} as possible. Intuitively, if the tuples whose weight components belong to $Q_{\leq}^{t_{weight}}$ were included in B_{low}, this would be a strong indication that low weight favors low blood pressure. □

Example 2.2.2 In the case of the `MEASUREMENTS` data shown in Example 2.2.1, t_{weight} must be chosen such that $\mathcal{H}(\pi_{bpres}|\pi_*^{t_{weight}})$ is minimal.

Figure 2.1 shows the distribution of the individual weights and marks the class where each point belongs. □

Example 2.2.3 The partition $\pi_{weight,bpres}$ consists of the blocks:

$$\begin{aligned} &\{\{w_1\}, \{w_2, w_{13}\}, \{w_3, w_{11}, w_5, w_{14}\}, \{w_7\}, \\ &\{w_9\}, \{w_8, w_{12}, w_{10}\}, \{w_4\}, \{w_{15}\}, \{w_6\}\} \end{aligned}$$

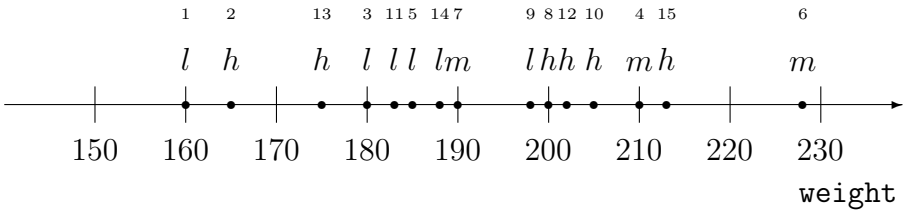


Figure 2.1: Distribution of the `weight` values

The boundary points of this partition are

$$w_1, w_2, w_{13}, w_3, w_{14}, w_7, w_9, w_8, w_{10}, w_4, w_{15}, w_6.$$

Here we denoted by w_h the weight of the individual s_h for $1 \leq h \leq 15$. □

2.3 Choosing Cutpoints

Fayyad [Fay91] showed that to obtain the least value of Shannon's conditional entropy $\mathcal{H}(\pi_{bpres}|\pi_*^{t_{weight}})$ the cutpoint t_{weight} may be chosen among the boundary points of the the partition $\pi_{weight,bpres}$. This is a powerful result that limits drastically the number of possible cut points and improves the tractability of the discretization.

Theorem 2.3.1 *If C is a cutpoint for attribute A that minimizes the measure $\frac{|T_1|}{|T|}\mathcal{H}(T_1) + \frac{|T_2|}{|T|}\mathcal{H}(T_2)$, where $T_1 \subset T$ and $T_2 = T - T_1$, for example set T, then C is a boundary point.*
(Fayyad's Theorem)

Simovici and Butterworth showed that the same choice of cutpoints may be made for a broader class of impurity measures, namely the impurity measures related to generalized conditional entropy [SB04].

Theorem 2.3.2 *Let T be a table where the class of the tuples is determined by the attribute A and let $\beta \in (1, 2]$. If S is a set of cutpoints such that the conditional entropy $\mathcal{H}_\beta(\pi_A|\pi_*^S)$ is minimal among the set of cutpoints with the same number of elements, then S consists of boundary points of the partition $\pi_{B,A}$ of $\mathsf{adom}(B)$.*
(proved in Chapter 4)

Moreover, when the purity of the partition $\pi_*^{t_{weight}}$ is replaced as a discretization criterion by the minimality of the entropic distance, d_β, between the partitions π_{bpres} and $\pi_*^{t_{weight}}$ (introduced in [SJ03]) the same method for selecting the cutpoint can be applied [SB04].

Theorem 2.3.3 *If $\beta \in (1, 2]$. If S is a set of cutpoints such that the distance $d_\beta(\pi_A, \pi_*^S)$ is minimal among the set of cutpoints with the same number of elements, then S consists of boundary points of the partition $\pi_{B,A}$ of $\mathsf{adom}(B)$.*
(proved in Chapter 4)

Discretizing $\mathsf{adom}(B)$ involves seeking a set of cutpoints such that $d_\beta(\pi_A, \pi_*^S) = \mathcal{H}_\beta(\pi_A|\pi_*^S) + \mathcal{H}_\beta(\pi_*^S|\pi_A)$ is minimal. In other words, seek a set of cutpoints such that the partition π_*^S induced on the set of rows is as close as possible to the target partition π_A. Initially, before adding cutpoints, $S = \emptyset$, $\pi_*^S = \omega$, and therefore $\mathcal{H}_\beta(\pi_A|\omega) = \mathcal{H}_\beta(\pi_A)$. Observe that as the set S grows the entropy $\mathcal{H}_\beta(\pi_A|\pi_*^S)$ decreases. Therefore the use of conditional entropy $\mathcal{H}_\beta(\pi_A|\pi_*^S)$ tends to favor large cutpoint sets for which the partition π_*^S is small in the partial ordered set $(\mathsf{PART}(S), \leq)$. In the extreme case, every point would be a cutpoint, a situation

that is clearly unacceptable. But, using the Generalized Barthélemy-Monjardet distance, this is counterbalanced by $\mathcal{H}_\beta(\pi_*^S|\pi_A)$ which will increase as the number of cutpoints increases. Note that initially $\mathcal{H}_\beta(\pi_*^S|\pi_A) = \mathcal{H}_\beta(\omega|\pi_A) = 0$ and as S increases it increases. Thus while the Fayyad-Irani technique halts the discretization process using the principle of minimum description length, the above strategy will continue until d_β stops decreasing.

More formally, the generalized conditional entropy is dually monotonic in its first argument and monotonic in its second, that is $\pi \leq \pi'$ implies $\mathcal{H}_\beta(\pi|\sigma) \geq \mathcal{H}_\beta(\pi'|\sigma)$, i.e. as the first argument becomes finer, more blocks, entropy increases. While $\sigma \leq \sigma'$ implies $\mathcal{H}_\beta(\pi|\sigma) \leq \mathcal{H}_\beta(\pi|\sigma')$, i.e. as the second argument becomes finer entropy decreases, as Simovici and Jaroszewicz showed in [SJ03].

2.4 The Generalized Barthélemy-Monjardet distance and Näive Bayes

The discretization method lends itself to use in the Näive Bayes algorithm particularly since Näive Bayes assumes the attributes are independent, and the method is myopic in its approach which means it assumes the attributes are independent.[BSS04]

Näive Bayesian classification is based on Bayes theorem and an assumption that the effect of an attribute value on a given class is independent of the values of the other attributes. It should be noted that it also needs the attributes to be nominal, so results were compared within Weka between this discretization method and the method used within Weka. The results were quite encouraging as will be seen in the next chapter.

CHAPTER 3

Discretization – Experimental Results

Any sufficiently advanced technology is indistinguishable from a rigged demo.

– Andy Finkel

Maier's Law: If the facts do not conform to the theory, they must be disposed of.

– N.R. Maier, "American Psychologist", March 1960 Corollaries: (1)

3.1 Discretization Algorithm and Experimental Results

This discretization algorithm was tested on several machine learning data sets from UCI [BM98] that have numerical attributes. After discretizations were performed with several values of β (typically $\beta \in \{1.5, 1.8, 1.9, 2\}$), the decision trees were built on the discretized data sets using the WEKA J48 variant of C4.5 [WF00]. The size, number of leaves, and accuracy of the trees are described in Table 3.1, where trees built using the Fayyad-Irani discretization method of J48 are designated as "standard". Figure 3.1 shows how the distance varies as the cardinality of the set of cutpoints increases on an artificial dataset.

The following charts show that the discretization technique has a significant impact of the size and accuracy of the decision trees and that an appropriate choice

Database	Experimental Results			
	Discretization method	Size	Number of leaves	Accuracy (stratified cross-validation)
heart-c	*standard*	*51*	*30*	*79.20*
	$\beta = 1.5$	20	14	77.36
	$\beta = 1.8$	28	18	77.36
	$\beta = 1.9$	35	22	76.01
	$\beta = 2.0$	54	32	76.01
glass	*standard*	*57*	*30*	*57.28*
	$\beta = 1.5$	32	24	71.02
	$\beta = 1.8$	56	50	77.10
	$\beta = 1.9$	64	58	67.57
	$\beta = 2.0$	92	82	66.35
ionosphere	*standard*	*35*	*18*	*90.88*
	$\beta = 1.5$	15	8	95.44
	$\beta = 1.8$	19	12	88.31
	$\beta = 1.9$	15	10	90.02
	$\beta = 2.0$	15	10	90.02
iris	*standard*	*9*	*5*	*95.33*
	$\beta = 1.5$	7	5	96
	$\beta = 1.8$	7	5	96
	$\beta = 1.9$	7	5	96
	$\beta = 2.0$	7	5	96
diabetes	*standard*	*43*	*22*	*74.08*
	$\beta = 1.8$	5	3	75.78
	$\beta = 1.9$	7	4	75.39
	$\beta = 2.0$	14	10	76.30

Table 3.1: Experimental Results

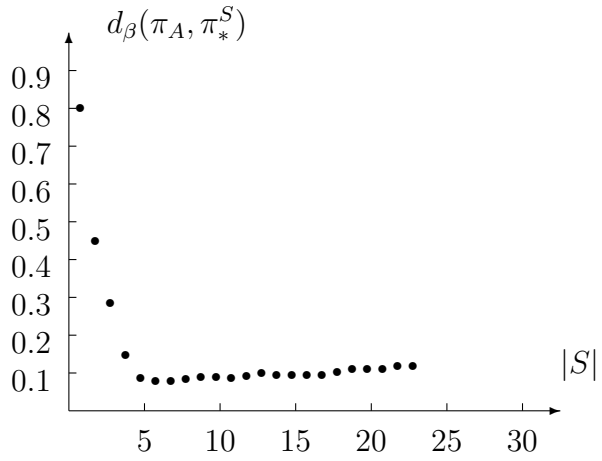
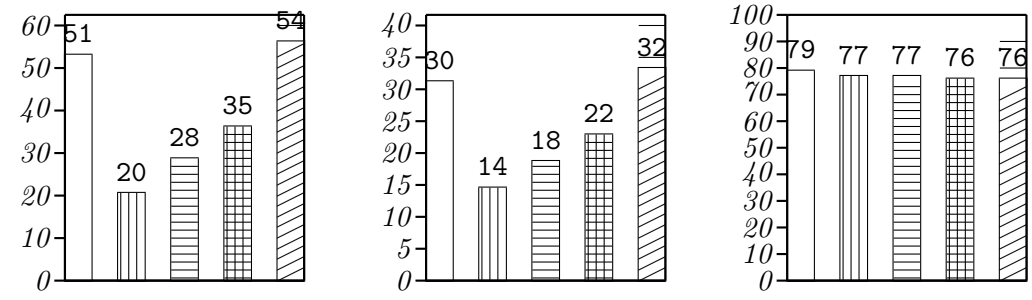


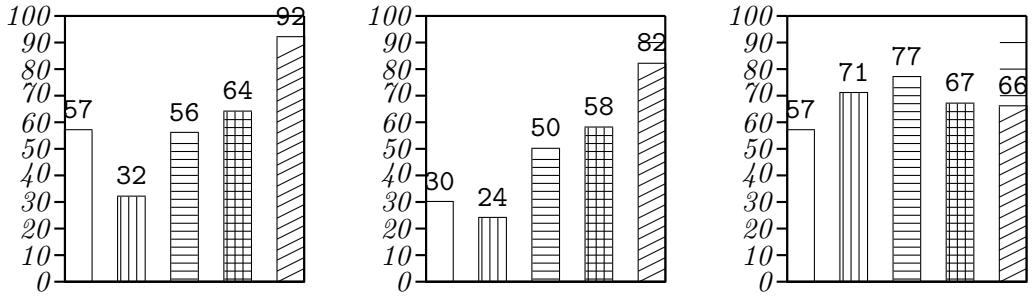
Figure 3.1: Variation of Distance with the Cardinality of the Set of Cutpoints

of β can reduce significantly the size and number of leaves of the decision trees, roughly maintaining the accuracy (measured by stratified 5-fold cross validation) or even increasing the accuracy as shown by the experiments on the glass data set. (see Figure 3.2).



Tree size Number of leaves Accuracy

Heart-c Data Set



Tree size Number of leaves Accuracy

Glass Data Set

Discretization method:

standard $\beta = 1.5$ $\beta = 1.8$ $\beta = 1.9$ $\beta = 2.0$

Figure 3.2: Experimental Results for the Heart-c and Glass Data Sets

CHAPTER 4

Discretization – Theorems

This is a one line proof... if we start sufficiently far to the left.

– Cambridge University math professor

4.1 Introduction

This chapter will show the proofs of Theorems 2.3.2 and 2.3.3. Theorem 2.3.2 is a generalization of Fayyad's Theorem 2.3.1 extending the choosing of cutpoints from Shannon entropy to generalized entropy. The second, Theorem 2.3.3, extends this to the Generalized Barthélemy-Monjardet distance.

4.2 Some Partition Notation

The following is some of the notation needed to follow the proofs:

$\mathsf{PART}(S)$: will denote the set of all partitions of a dataset, S.

$\pi \leq \sigma$: Suppose there are two partitions of the set S, π and σ, then this could be denoted by $\pi, \sigma \in \mathsf{PART}(S)$. Write $\pi \leq \sigma$ if each block of π is included in a block of σ.

$\pi_1 \wedge \pi_2$: If $\pi_1, \pi_2 \in \mathsf{PART}(S)$, then $\pi_1 \wedge \pi_2$ denotes the partition whose blocks are

all non-empty intersections of the form $K \cap H$, where $K \in \pi_1$ and $H \in \pi_2$. Thus $\pi_1 \wedge \pi_2 \leq \pi_1$ and $\pi_1 \wedge \pi_2 \leq \pi_2$.

$\pi_1 \vee \pi_2$: denotes the partition whose blocks, D_k, are the smallest blocks such that for every K_i in π_1 and H_j in π_2 there is some D_{k_i} and D_{k_j} such that $K_i \subseteq D_{k_i}$ and $H_j \subseteq D_{k_j}$. Note that $\pi_1 \leq \pi_1 \vee \pi_2$ and $\pi_2 \leq \pi_1 \vee \pi_2$.

π_A: is a partition of the set of tuples of a table determined by the values of an attribute A, as an aside recall that SQL computes such a partition using the `group by A` option of a `select phrase`.

$\pi_{B,A}$: If the tuples are first sorted on attribute B and then partitioned on attribute A, this partition would be denoted $\pi_{B,A}$ of $\mathsf{adom}(B)$ and consists of the longest runs of *consecutive* B-components of the tuples in this list that belong to the *same block* K of the partition π_A.

$\langle x \rangle$, $x^{\downarrow}$, $x^{\uparrow}$: The *boundary points* of the partition $\pi_{B,A}$ are the least and the largest elements of each of the blocks of the partition $\pi_{B,A}$. If $x \in \mathsf{adom}(B)$ is a tuple, the block of $\pi_{B,A}$ that contains x will be denoted by $\langle x \rangle$ and the least and largest elements of $\langle x \rangle$ will be denoted by $x^{\downarrow}$ and $x^{\uparrow}$, respectively.

π_*: Suppose π is a partition of $\mathsf{adom}_T(B)$, then π_* denotes the partition of T that corresponds to π. A block of π_* consists of all tuples whose B-projections belong to the same block of π. Note that $\pi_{B,A*} \leq \pi_A$ for any attribute B.

4.2.1 Finding A Cutpoint

To find a cutpoint Fayyad-Irani used the Shannon entropy:

Theorem 2.3.1 *If C is a cutpoint for attribute A that minimizes the measure*

$\frac{|T_1|}{|T|}\mathcal{H}(T_1) + \frac{|T_2|}{|T|}\mathcal{H}(T_2)$, *where* $T_1 \subset T$ *and* $T_2 = T - T_1$, *for example set* T, *then* C *is a boundary point.*

Here is a generalization of this:

Theorem 2.3.2 *Let* T *be a table where the class of the tuples is determined by the attribute* A *and let* $\beta \in (1, 2]$. *If* S *is a set of cutpoints such that the conditional entropy* $\mathcal{H}_\beta(\pi_A|\pi_*^S)$ *is minimal among the set of cutpoints with the same number of elements, then* S *consists of boundary points of the partition* $\pi_{B,A}$ *of* $\mathsf{adom}(B)$.

Proof: Note that since $\pi_{B,A*} \leq \pi_A$ for any $t \in \mathsf{adom}(B)$, the set $\langle t \rangle$ is included in some block P_g of the partition π_A (see Figure 4.1).

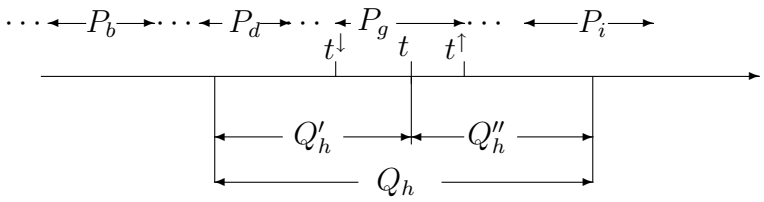


Figure 4.1: Position of Q_h relative to other blocks

The proof is by induction on the number of cutpoints $\ell = |S|$. If $\ell = 0$, the statement is immediate since in this case π_*^S is the one-class partition ω_T of the set of tuples T. Suppose that the statement holds for a set of cutpoints that contain ℓ elements and let $Z = S \cup \{t\}$, where $S = \{t_1, \ldots, t_\ell\}$ is a set of cutpoints that is a subset of the set of boundary points of $\pi_{B,A}$, $|S| = \ell$ and $t \notin S$.

Let $\pi_A = \{P_1, \ldots, P_k\}$ and $\pi_*^S = \{Q_0, \ldots, Q_\ell\}$. The conditional entropy

$\mathcal{H}_\beta(\pi_A|\pi_*^S)$ is given by:

$$\mathcal{H}_\beta(\pi_A|\pi_*^S) = \frac{1}{(2^{1-\beta}-1)|T|^\beta}\left(\sum_{i=1}^{k}\sum_{j=0}^{\ell}|P_i\cap Q_j|^\beta - \sum_{j=0}^{\ell}|Q_j|^\beta\right).$$

Suppose that the new cut point t is placed between t_{h-1} and t_h. Then, the partition π_*^Z is obtained from π_*^S by splitting Q_h in Q'_h and Q''_h. Also, t is located between two cutpoints $t^\downarrow$ and $t^\uparrow$ of the partition $\pi_{B,A}$. By a previous remark the set of tuples whose B-component is included in the interval $\langle t\rangle = [t^\downarrow, t^\uparrow]$ is a subset of a block P_g of the partition π_A.

The variation of the entropy caused by the introduction of the split in Q_h is given by:

$$\begin{aligned}
&\mathcal{H}_\beta(\pi_A|\pi_*^Z) - \mathcal{H}_\beta(\pi_A|\pi_*^S)\\
&= \frac{1}{(2^{1-\beta}-1)|T|^\beta}\left(\sum_{i=1}^{k}\sum_{j=0,j\neq h}^{\ell}|P_i\cap Q_j|^\beta - \sum_{j=0,j\neq h}^{\ell}|Q_j|^\beta\right)\\
&\quad+\frac{1}{(2^{1-\beta}-1)|T|^\beta}\left(\sum_{i=1}^{k}|P_i\cap Q'_h|^\beta + \sum_{i=1}^{k}|P_i\cap Q''_h|^\beta - |Q'_h|^\beta - |Q''_h|^\beta\right)\\
&\quad-\left(\sum_{i=1}^{k}\sum_{j=0}^{\ell}|P_i\cap Q_j|^\beta - \sum_{j=0}^{\ell}|Q_j|^\beta\right). \qquad (4.1)
\end{aligned}$$

Since the partition π_*^S is such that $\mathcal{H}_\beta(\pi_A|\pi_*^S)$ achieves a local minimum, it follows that the difference $\mathcal{H}_\beta(\pi_A|\pi_*^Z) - \mathcal{H}_\beta(\pi_A|\pi_*^S)$ needs to have a local minimum in order for $\mathcal{H}_\beta(\pi_A|\pi_*^Z)$ to achieve a local minimum.

The number of tuples in the sets $P_i \cap Q_j$ for $i \neq g$ and $j \neq h$ is unaffected by the split of Q_h since $\langle t\rangle \subseteq P_g$. To make it simpler to see use a constant K (independent on t) to replace the parts of Equation 4.1 such that the variation in

entropy can be written as

$$\begin{aligned}\mathcal{H}_\beta(\pi_A|\pi_*^Z) - \mathcal{H}_\beta(\pi_A|\pi_*^S) &= \frac{1}{(2^{1-\beta}-1)|T|^\beta}\left(K + |P_g \cap Q'_h|^\beta + |P_g \cap Q''_h|^\beta\right. \\ &\quad \left. -|Q'_h|^\beta - |Q''_h|^\beta\right).\end{aligned}$$

Denote $n = |\langle t \rangle|$, and let μ be the number of tuples whose B-component is in $(t^\downarrow, t]$. Then, the number of tuples whose B-component is in $(t, t^\uparrow]$ is $n-\mu$. Denote by a, b the number of tuples in Q'_h and Q''_h whose B-component is less than $t^\downarrow$ and $t^\uparrow$, respectively. With these notations we can write

$$\begin{aligned}\mathcal{H}_\beta(\pi_A|\pi_*^Z) - \mathcal{H}_\beta(\pi_A|\pi_*^S) &= \frac{1}{(2^{1-\beta}-1)|T|^\beta}\left(K + \mu^\beta + (n-\mu)^\beta\right. \\ &\quad \left. -(a+\mu)^\beta - (b+n-\mu)^\beta\right),\end{aligned}$$

If μ is regarded as a continuous variable varying in the interval $[0, n]$, then one needs to examine the variation of the real-valued function

$$F(\mu) = \frac{1}{(2^{1-\beta}-1)|T|^\beta}\left(K + \mu^\beta + (n-\mu)^\beta - (a+\mu)^\beta - (b+n-\mu)^\beta\right),$$

on the interval $[0, n]$. The second derivative of this function is:

$$\begin{aligned}F''(\mu) &= \frac{\beta(\beta-1)}{(2^{1-\beta}-1)|A|^\beta}\left(\mu^{\beta-2} + (n-\mu)^{\beta-2}\right. \\ &\quad \left. -(a+\mu)^{\beta-2} - (b+n-\mu)^{\beta-2}\right).\end{aligned}$$

Since $\beta > 1$, $\frac{\beta(\beta-1)}{2^{1-\beta}-1} < 0$. Also, for $1 \leq \beta < 2$ it is true that $\mu^{\beta-2} - (a+\mu)^{\beta-2} > 0$ and $(n-\mu)^{\beta-2} - (b+n-\mu)^{\beta-2} > 0$, which imply that the second derivative is negative on $[0, n]$. This proves that the minimum of this function is attained either for $\mu = 0$ or for $\mu = n$, that is, in one of the $\pi_{B,A}$-boundary points.

The case $\beta = 2$ is immediate since in this situation F is a linear function of μ.

∎

The next theorem is a companion to Fayyad's result and makes use of the same hypothesis as Theorem 2.3.2 and is similar except it deals with the Generalized Barthélemy-Monjardet distance function.

Theorem 2.3.3 *If $\beta \in (1, 2]$. If S is a set of cutpoints such that the distance $d_\beta(\pi_A, \pi_*^S)$ is minimal among the set of cutpoints with the same number of elements, then S consists of boundary points of the partition $\pi_{B,A}$ of $\mathsf{adom}(B)$.*

Proof: As before the argument is by induction on $|S|$ and the basis $|S| = 0$ is vacuous. Suppose that the statement is true for $|S| = \ell$, so S consists of boundary points of the partition $\pi_{B,A}$.

The conditional entropy $\mathcal{H}_\beta(\pi_*^S|\pi_A)$ is given by

$$\mathcal{H}_\beta(\pi_*^S|\pi_A) = \frac{1}{(2^{1-\beta}-1)|T|^\beta}\left(\sum_{i=1}^{k}\sum_{j=0}^{\ell}|P_i \cap Q_j|^\beta - \sum_{i=1}^{k}|P_i|^\beta\right).$$

If a new cutpoint t is added between the boundary points t_{h-1} and t_h to obtain the new set of cutpoints $Z = S \cup \{t\}$, the new value of the conditional entropy is:

$$\begin{aligned}\mathcal{H}_\beta(\pi_*^Z|\pi_A) &= \frac{1}{(2^{1-\beta}-1)|T|^\beta}\left(\sum_{i=1}^{k}\sum_{j=0, j\neq h}^{\ell}|P_i \cap Q_j|^\beta + \right.\\ &\left.\sum_{i=1}^{k}|P_i \cap Q'_h|^\beta + \sum_{i=1}^{k}|P_i \cap Q''_h|^\beta - \sum_{i=1}^{k}|P_i|^\beta\right).\end{aligned}$$

Thus, we have:

$$\begin{aligned}\mathcal{H}_\beta(\pi_*^Z|\pi_A) - \mathcal{H}_\beta(\pi_*^S|\pi_A) &= \frac{1}{(2^{1-\beta}-1)|T|^\beta}\left(\sum_{i=1}^{k}|P_i \cap Q'_h|^\beta + \right.\\ &\left.\sum_{i=1}^{k}|P_i \cap Q''_h|^\beta + \sum_{i=1}^{k}|P_i \cap Q_h|^\beta\right).\end{aligned}$$

Since $\langle t\rangle \subseteq P_g$ only the intersections that contain P_g depend on the position of the new cutpoint t. Therefore, the variation of the conditional entropy can be written

as

$$\begin{aligned}
&\mathcal{H}_\beta(\pi_*^Z|\pi_A) - \mathcal{H}_\beta(\pi_*^S|\pi_A) \\
&\quad = \frac{1}{(2^{1-\beta}-1)|T|^\beta}\left(H + |P_g \cap Q'_h|^\beta + |P_g \cap Q''_h|^\beta - |P_g \cap Q_h|^\beta\right),
\end{aligned}$$

where H is a constant that does not depend on t. Using the notation previously introduced produces

$$\begin{aligned}
&\mathcal{H}_\beta(\pi_*^Z|\pi_A) - \mathcal{H}_\beta(\pi_*^S|\pi_A) \\
&\quad = \frac{1}{(2^{1-\beta}-1)|T|^\beta}\left(H + \mu^\beta + (n-\mu)^\beta - n^\beta\right).
\end{aligned}$$

The second derivative real-valued function G defined by

$$G(\mu) = \frac{1}{(2^{1-\beta}-1)|T|^\beta}\left(H + \mu^\beta + (n-\mu)^\beta - n^\beta\right)$$

for $\mu \in (0, n]$ is

$$G''(\mu) = \frac{\beta(\beta-1)}{(2^{1-\beta}-1)|T|^\beta}\left(\mu_{\beta-2} + (n-\mu)^{\beta-2}\right)$$

and is clearly negative.

The variation of the distance $d_\beta(\pi_A, \pi_*^Z) - d_\beta(\pi_A, \pi_*^S)$ is the sum of the variations of the entropies $\mathcal{H}_\beta(\pi_A|\pi_*^Z) - \mathcal{H}_\beta(\pi_A|\pi_*^S)$ and $\mathcal{H}_\beta(\pi_*^Z|\pi_A) - \mathcal{H}_\beta(\pi_*^S|\pi_A)$. With the above notation, this variation equals $F(\mu) + G(\mu)$, where F is the function introduced in the proof of Theorem 2.3.2. Since $F''(\mu) + G''(\mu) < 0$, the minimum value of the distance can be attained only when t coincides with either $t^\downarrow$ or with $t^\uparrow$. ∎

CHAPTER 5

Discretization – Coding

If builders built buildings the way programmers wrote programs, then the first woodpecker to come along would destroy civilization.

–Anonymous

5.1 Introduction

First, here is a pseudo-code explanation of how discretization is done:
The algorithm shown below is used for discretizing an attribute B. It makes successive passes over the table and, at each pass it adds a new cutpoint chosen among the boundary points of $\pi_{B,A}$.

Input: `A table` T`, a class attribute` A`,`
`and a real-valued attribute` B`.`

Output: `A discretized attribute` B`.`

Method: `sort table` T `on attribute` B`;`
`compute the set` BP `of boundary points of partition` $\pi_{B,A*}$;
$S = \emptyset$; $d = \infty$;
`while` BP $\neq \emptyset$ `do`
`let` $t =$ arg min $_{t \in \mathsf{BP}} d_\beta(\pi_A, \pi_*^{S \cup \{t\}})$;
`if` $d \geq d_\beta(\pi_A, \pi_*^{S \cup \{t\}})$ `then`

```
            begin
                S = S ∪ {t}; BP = BP − {t};
                d = d_β(π_A, π*^S)
            end
        else exit while loop;
    end while
    for π*^S = {Q_0, ..., Q_ℓ} replace
    every attribute in Q_i by i for 0 ≤ i ≤ ℓ.
```

The while loop is running for as long as there exist candidate boundary points and it is possible to find a new cutpoint t such that the distance $d_\beta(\pi_A, \pi_*^{S\cup\{t\}})$ is less than the previous distance $d_\beta(\pi_A, \pi_*^S)$. An experiment performed on a synthetic dataset shows that a substantial amount of time (about 78% of the total time) is spent on decreasing the distance by the last 1% (see Figure 3.1). Therefore, in practice run a search for a new cutpoint only if $|d - d_\beta(\pi_A, \pi_*^{S\cup\{t\}})| > 0.01d$.

The discussion will now turn to an implementation some of the functions used. At first the distance function was written in Java using Oracle. Although this was successful, it was painfully slow. Thus, System R was used to do much of preliminary work, and Weka was used to do some of the analysis, e.g. creating the Naïve Bayes and J48 classifiers.

5.2 A Brief Overview of System R

System R is a public version of System S where S was created by Bell Labs. It is designed as an environment for doing statistics. There is a large complement of functions to allow one to do statistics from simple functions like mean to sophisti-

cated ones like creating dendrograms. There is also a language which is functional and object-oriented included with the environment. Since it is an interactive environment, it allows the programmer to build it up in pieces, trying each piece as one goes, and putting them together. It also has the capability of incorporating functions from other languages, especially compiled languages such as C or FORTRAN, which allows the programmer to speed up the running considerably. Because R is interactive, it is ideal as a test bed to implement functions quickly and relatively painlessly.

As was said before, there are techniques that are designed to work with discrete data so one needs to discretize data which is continuous. The theory behind this has been described previously in Chapter 1. The implementation of this in R used the following steps:

- call **getdistab(exponent, tab, tar, ..., stopnum=20)**
  - It goes through each column (attribute) calling finddiscol(tab, i, tar, exponent).
- call **finddiscol(tab, i, tar, exponent)**
  - calls **sortcol2(tab, att, tar)** which sorts the data on the designated attribute, att, and the target attribute, tar.
  - calls **getcutpts(tab, att, tar, index)** which gets the cutpoints for the given attribute.
  - calls **getdiscol** which returns the column discretized.
    - calls **setdiscol(index, att, tar, cutpts, exponent, stopnum** which returns the column with the latest discretization done.

* calls **finddist(a1, a2, exponent** which actually finds the distance between the two attributes, a1 and a2.

The above inputs are:

exponent: the exponent to be used in the Generalized Barthélemy-Monjardet distance,

tab: the table of tuples,

tar: the class (or target) attribute,

stopnum: the maximum number of blocks to be found (to ensure it will stop in a reasonable time if it might create a partition with a huge number of blocks),

i: the column number of the column to be operated on,

att: an attribute,

cutpts: a vector of the cutpoints

5.2.1 General Comment

The information obtained was exported to Weka for analysis, and the results imported back to R to create graphs.

CHAPTER 6

Clustering Introduction

A gaggle of swans, a pride of lions, a murder of crows

6.1 Introduction

This chapter will introduce attribute selection, i.e. how to pick good attributes for predicting the class. According to Tsamardinos and Aliferis [TA03], there is a basic concept one needs to worry about when choosing attributes: relevance. What relevance means will be clarified below.

First, why limit the attributes? Why not use all the attributes? Theoretically the more attributes the better. However, in learning experiments involving many practical algorithms a reasonable selection of the attributes, rather than the full set, often yields models with better generalization performance [DHS01]. Second, using all the attributes may be very expensive. Third, smaller models are easier to understand and less computationally expensive when performing inference and prediction. A follow-on to the previous comment, if the object is to increase the understanding of the interrelationship of the attributes and what will cause which result, smaller is considerably easier to understand. Fourth, some of the attributes may be redundant, and thus, not all of them are necessary to increase accuracy. Finally, some of the attributes may be noise, i.e. they are irrelevant to the class

and will only confuse a classifier.

6.2 Relevance

As data mining is used to study data sets with more and more attributes, the number of attributes tends to overwhelm many strategies for analyzing the data. Thus, there is a real need to try to filter the attributes and keep only the truly relevant ones.

The first problem is ascertaining exactly what is a relevant attribute. This has been studied by various authors, such as [BL97]. The basic issue relative to relevance is *relevant to what*. Different definitions may be appropriate depending on what the goal is. For example, a common idea would be to have the attributes be relevant to the class attribute. Of course, even that demands clarification.

Suppose we have a Table, T, with a set of attributes, A_i, such that each attribute has a domain $\mathsf{adom}(A_i)$. The learning algorithm is given a set S of training data taken from the table where each datum is a tuple consisting of all the attributes including the class attribute. Although the learning algorithm sees only the sample S, it is often helpful to postulate two additional quantities: a probability distribution function D over the whole table and a target function t from examples to labels. Now S can be viewed as having been produced by repeatedly selecting examples D and then labeling them according to the function t.

Given this setup, relevance can be viewed as "relevant to the target attribute".

Definition 6.2.1 *(**Relevant to the target**) An attribute A_i is relevant to a target attribute c if there exists a pair of tuples, A and B, in the table such that A and B differ only in their assignment to A_i and $c(A) \neq c(B)$.*

In other words, an attribute A_i is relevant if there exists some tuple in the table such that changing the value of A_i affects the value of the class attribute.

Notice that this notion has the obvious drawback that the learning algorithm has access to only the tuples in S and thus can not necessarily determine whether or not an attribute is relevant to the entire table, T. Further, if each of the attributes is redundant, e.g. each attribute has a duplicate, then it is not possible to find a tuple where they differ by a single attribute. To remedy this, John, Kohavi, and Pfleger [JKP94] suggest the following two definitions.

Definition 6.2.2 *(**Strongly Relevant to the Sample/Distribution**) An attribute A_i is strongly relevant to sample S if there exists tuples A and B in S that differ only in their assignment to A_i and have different labels (or have different distributions of labels if they appear in S multiple times). Similarly, A_i is strongly relevant to target c and distribution D if there exists tuples A and B having non-zero probability over D that differ only in their assignment to A_i and satisfy $c(A) \neq c(B)$.*

This is the same as Definition 6.2.1 except A and B are only required to be in S.

Definition 6.2.3 *(**Weakly Relevant to the Sample/Distribution**) An attribute A_i is weakly relevant to sample S (or to target c and distribution D) if it is possible to remove a subset of the attributes so that A_i becomes strongly relevant.*

These notions of relevance are useful in attempting to decide which attributes to keep and which to ignore. Attributes that are strongly relevant are generally important to keep no matter what since removing them would add ambiguity to

the sample. Attributes that are weakly relevant may or may not be important to keep depending on which other attributes are ignored.

6.3 Filters

Filters are an approach to attribute selection that introduces a separate process for this purpose that occurs before the basic induction step [JKP94]. The preprocessing step uses general characteristics of the training set to select some attributes and exclude others. Note, that filters are independent of the induction algorithm that will use their output so they can be combined with any such method.

Perhaps the simplest filtering strategy is to evaluate each attribute individually based on its correlation with the target function, e.g. use the Generalized Barthélemy-Monjardet distance, and then choose the k attributes with the highest value, i.e. closest to the class attribute.

6.4 Wrappers

Wrappers are a method named by John et al. [JKP94]. The typical wrapper algorithm searches the same space of attribute subsets as filter methods, but it evaluates alternative sets by running some induction algorithm on the training data and using the estimated accuracy of the resulting classifier as its metric. That is, it repeatedly takes a set of attributes; uses the given classifier, e.g. J48; obtains results; and compares these results to see which is the best set of attributes for classification.

The essential argument for wrapper approaches is that the induction method that will use the attribute subset should provide a better estimate of accuracy

than a separate measure that may have an entirely different inductive bias. The major disadvantage of wrapper methods over filter methods is the computational cost since the wrapper method calls the induction algorithm for each attribute set considered.

CHAPTER 7

Clustering Attributes

Felson's Law: To steal ideas from one person is plagiarism; to steal from many is research.

– Anonymous

7.1 Introduction

The performance and the robustness of classification algorithms is strongly influenced by the dimensionality of the data. Also, the usefulness of the results yielded by classifiers is increased when relatively few features [1] are involved in the classification. Thus, selecting relevant features for the construction of classifiers has received a great deal of attention. A lucid taxonomy of algorithms for feature selection is discussed in [ZJ96]. A more recent reference is [GE03]. Several approaches to feature selection have been explored, including wrapper techniques [KJ97], support vector machines [BGL00], neural networks [KWR01] and prototype-based feature selection [HCB03]; the last is close to the approach used in this paper.

The central idea of this chapter is to introduce an algorithm for feature selection that clusters attributes using the Generalized Barthélemy-Monjardet distance and then uses a hierarchical clustering for feature selection. [BPS05]

[1] Feature and attribute are used interchangeably in data mining.

Hierarchical algorithms generate clusters that are placed in a cluster tree. Clusterings are obtained by extracting those clusters that are situated at a given height in this tree. The intent is to show that in building classifiers one needs to retain only an attribute that is centrally situated in each cluster and this data compression can be achieved with little or no penalty in terms of the accuracy of the classifier produced.

To give a more rigorous structure to the argument, consider the following: an *object system* is a pair $\mathcal{S} = (S, H)$, where S is set called the set of objects of $\mathcal{S}$ and $H = \{A_1, \ldots, A_m\}$ is a set of mappings defined on S. Assume that for each mapping A_i (referred to as an attribute of $\mathcal{S}$) there exists a nonempty set E_i called the domain of A_i such that $A_i : S \longrightarrow E_i$ for $1 \leq i \leq m$. The value of an attribute A_i on an object t is denoted by $t[A_i]$. This terminology is consistent with the terminology used in relational databases, where a table can be regarded as an object system; however, the notion of object system is more general because objects have an identity as members of the set S, instead of being regarded as just m-tuples of values. In this spirit, $t[A_i]$ will denote the *projection of* t *on* A_i.

Let S be a set. As introduced in 1.5, a *partition on* S is a non-empty collection of subsets of S indexed by a set I, $\pi = \{B_i \mid i \in I\}$ such that $\bigcup_{i \in I} B_i = S$ and $i \neq j$ implies $B_i \cap B_j = \emptyset$. The sets B_i are commonly referred to as the *blocks of the partition* π. The set of partitions on S is denoted by $\mathsf{PART}(S)$.

An attribute A of an object system $\mathcal{S} = (S, H)$ generates a partition π^A of the set of objects S, where two objects belong to the same block of π^A if they have the same projection on A. B_a^A denotes the block of π^A that consists of all tuples of S whose A-component is a.

The set of partitions of a set can be naturally equipped with a partial order.

For $\pi, \sigma \in \mathsf{PART}(S)$ write $\pi \leq \sigma$ if every block B of π is included in a block of σ, or equivalently, if every block of σ is an exact union of blocks of π. This partial order generates a lattice structure on $\mathsf{PART}(S)$ [2]; this means that for every two partitions $\pi, \pi' \in \mathsf{PART}(S)$ there is a least partition π_1 such that $\pi \leq \pi_1$ and $\pi' \leq \pi_1$ and there is a largest partition π_2 such that $\pi_2 \leq \pi$ and $\pi_2 \leq \pi'$. The first partition is denoted by $\pi \vee \pi'$, while the second is denoted by $\pi \wedge \pi'$ [3].

7.2 Distance between partitions and the Pearson index

To introduce a metric on the set of partitions of a finite set define the mapping $v : \mathsf{PART}(S) \longrightarrow \mathbb{R}$ by $v(\pi) = \sum_{i=1}^{n} |B_i|^2$, where $\pi = \{B_1, \ldots, B_n\}$. The mapping v is a lower valuation on $\mathsf{PART}(S)$, that is,

$$v(\pi \vee \sigma) + v(\pi \wedge \sigma) \geq v(\pi) + v(\sigma) \tag{7.1}$$

for $\pi, \sigma \in \mathsf{PART}(S)$.

For every lower valuation v the mapping $d : (\mathsf{PART}(S))^2 \longrightarrow \mathbb{R}$ defined by $d(\pi, \sigma) = v(\pi) + v(\sigma) - 2 \cdot v(\pi \wedge \sigma)$ is a metric on $\mathsf{PART}(S)$ (see [BL95, Bar78, Mon81]). d will be referred to as the *Barthélemy-Monjardet distance*[4].

Using the cardinalities of the blocks of the partitions yields

$$d(\pi, \sigma) = \sum_i |B_i|^2 + \sum_j |C_j|^2 - 2 \sum_i \sum_j |B_i \cap C_j|^2,$$

where $\pi = \{B_1, \ldots, B_n\}$ and $\sigma = \{C_1, \ldots, C_p\}$.

This metric was used for the development of an incremental clustering algorithm [SS04]; in that paper Simovici and Singla used it to cluster attributes.

[2] see fig. 1.2

[3] see 1.5 and 4.2 for more information

[4] Note, this is the Generalized Barthélemy-Monjardet distance with $\beta = 2$.

For a partition $\pi = \{B_1, \ldots, B_n\}$ denote by $M(\pi)$ and $m(\pi)$ the largest and the smallest size of a block of π.

Let $\pi = \{B_1, \ldots, B_n\}$, $\sigma = \{C_1, \ldots, C_p\}$ be two partitions. The *contingency matrix* of π and σ is the matrix $P_{\pi,\sigma}$ whose entries are given by $p_{ij} = |B_i \cap C_j|$ for $1 \leq i \leq n$ and $1 \leq j \leq p$. The Pearson χ^2 association index can be written in this context as:

$$\chi^2_{\pi,\sigma} = \sum_i \sum_j \frac{(p_{ij} - |B_i||C_j|)^2}{|B_i| \cdot |C_j|}$$

It is well-known (See [Agr97]) that the asymptotic distribution of this index is a χ^2-distribution with $(n-1)(p-1)$ degrees of freedom.

Theorem 7.2.1 *If S is a finite set and $\pi, \sigma \in \mathsf{PART}(S)$, where $\pi = \{B_1, \ldots, B_n\}$ and $\sigma = \{C_1, \ldots, C_p\}$, then*

$$\frac{v(\pi) + v(\sigma) - d(\pi,\sigma)}{2M(\pi)M(\sigma)} - 2np + |S|^2 \leq \chi^2_{\pi,\sigma} \leq \frac{v(\pi) + v(\sigma) - d(\pi,\sigma)}{2m(\pi)m(\sigma)} - 2np + |S|^2.$$

Proof:

Note that

$$\chi^2_{\pi,\sigma} = \sum_i \sum_j \frac{p_{ij}^2}{|B_i| \cdot |C_j|} - 2np + |S|^2.$$

Since $m(\pi)m(\sigma) \leq |B_i||C_j| \leq M(\pi)M(\sigma)$:

$$\frac{p_{ij}^2}{M(\pi)M(\sigma)} \leq \frac{p_{ij}^2}{|B_i| \cdot |C_j|} \leq \frac{p_{ij}^2}{m(\pi)m(\sigma)}.$$

Thus,

$$\frac{v(\pi \wedge \sigma)}{M(\pi)M(\sigma)} - 2np + |S|^2 \leq \chi^2_{\pi,\sigma} \leq \frac{v(\pi \wedge \sigma)}{m(\pi)m(\sigma)} - 2np + |S|^2$$

Since $d(\pi,\sigma) = v(\pi) + v(\sigma) - 2\sum_i \sum_j p_{ij}^2$, it follows that

$$\frac{v(\pi) + v(\sigma) - d(\pi,\sigma)}{2M(\pi)M(\sigma)} - 2np + |S|^2 \leq \chi^2_{\pi,\sigma} \leq \frac{v(\pi) + v(\sigma) - d(\pi,\sigma)}{2m(\pi)m(\sigma)} - 2np + |S|^2,$$

which concludes the argument. ∎

Note that for partitions π, σ if size of the blocks of each partition stay the same but the distance between the partitions varies, the Pearson coefficient decreases as the distance increases, and, thus, the probability that π and σ are independent increases with the distance. This suggests that partitions that are correlated are close in the sense of the Barthélemy-Monjardet distance; therefore it would seem appropriate to say, if attributes are clustered using the corresponding distance between partitions clusters of attributes could be replaced by their centroids and, thereby, drastically reduce the number of attributes involved in a classification without significant decreases in accuracy of the resulting classifiers.

CHAPTER 8

Cluster Algorithms – Experimental Results

Williams and Holland's Law: If enough data is collected, anything may be proven by statistical methods.

– Anonymous

There are known knowns. There things we know that we know. There are known unknowns. That is to say, there are things that we know we don't know. But there are also unknown unknowns. There are things we don't know that we don't know.

– Defense Secretary Donald Rumsfeld

8.1 Experimental Validation

Nine data sets were examined: Anneal,Hepatitis, Ionosphere, Lymph, Mushrooms, Soybean, Splice, Voting, and Zoo[1]. In each case, starting from the matrix $(d(\pi^{A_i}, \pi^{A_j}))$ of Barthélemy-Monjardet distances[2] between the partitions of the attributes $A_1, \ldots, A_n$, the attributes were clustered using an agglomerative hierarchical algorithm [KR90]. Nine set were used to examine the results for data sets whose properties were quite different.

[1] from the UCI data sets

[2] the Generalized Barthélemy-Monjardet distance with $\beta = 2$

Clusterings were extracted from the tree produced by the algorithm by cutting the tree at various heights starting with the maximum height of the tree created above (corresponding to a single cluster) and working down to a height of 0 (which consists of single-attribute clusters). A 'representative' attribute was chosen for each cluster as the attribute that has the minimum total distance to the other members of the cluster, again using the Barthélemy-Monjardet distance. The J48 and the Naïve Bayes algorithms of the WEKA package [WF00] were used for constructing classifiers on data sets obtained by projecting the initial data sets on the sets of representative attributes. In addition, the data set was also run through the CSF [Hal99] and Wrapper methods within Weka.

The CSF, correlation-based feature selection, is particularly interesting since it uses a method that is quite unlike the method used in this paper. The present method clusters attributes that are similar and picks the one closest to all the in the group. While CSF picks attributes that are highly correlated with the class attribute, yet uncorrelated with each other.

It should be noted that while the the Barthélemy-Monjardet distance method is not always better then the other two methods, as will be seen below, it does have the advantage of showing the user how various attributes cluster which could be of significant benefit by itself.

To give a fuller of explanation of the results, consider the UCI Mushroom data set [BM98] which consists of 8124 instances with 22 nominal attributes. The class distribution is 51.8% edible and 48.2% poisonous. The following figures and tables show:

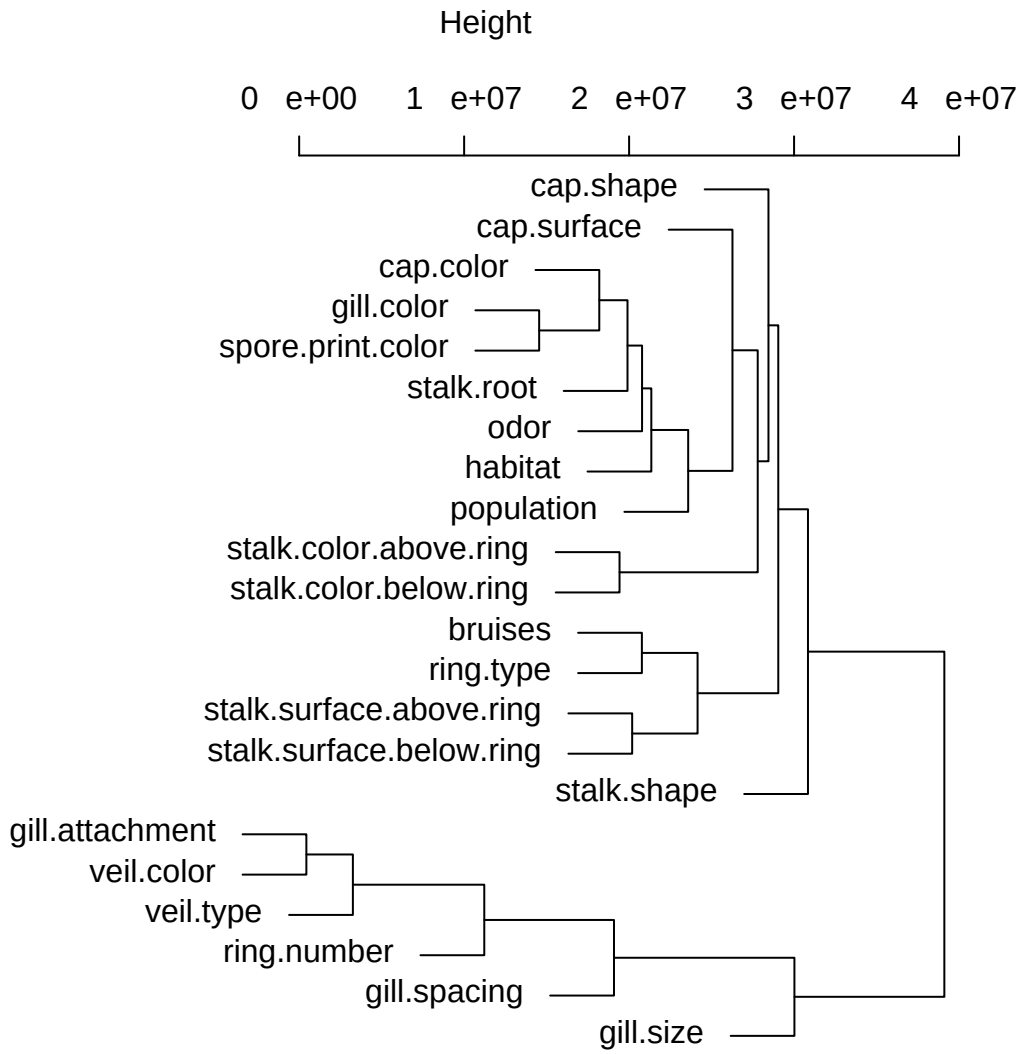
- the dendrogram created, Figure 8.1.
- the attributes used by and the results of the CSF and Wrapper methods built

in to Weka, Table 8.3.

- the heights, accuracies of Naïve Bayes and J48, and the attributes used at each clustering height[3], Table 8.1.
- what the Vi's in the previous table stand for in the data set, see Table 8.2.
- graphs of Naïve Bayes and J48 accuracies at each clustering height, Figure 8.2 and Figure 8.3.
- One might note that the Barthélemy-Monjardet distance results take more attributes to get the same results as the other two methods.

[3]i.e. each height where attributes are clustered

Figure 8.1: Mushroom Data Set – Dendrogram



Mushroom - NB & J48

Height	NB	J48	Attributes Used
0	95.8	100.0	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V14 V15 V16 V17 V18 V19 V20 V21 V22
436,456	95.7	100.0	V01 V02 V03 V04 V05 V07 V08 V09 V10 V11 V12 V13 V14 V15 V16 V17 V18 V19 V20 V21 V22
3,257,492	95.8	100.0	V01 V02 V03 V04 V05 V07 V08 V09 V10 V11 V12 V13 V14 V15 V17 V18 V19 V20 V21 V22
11,222,285	95.6	100.0	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V14 V15 V19 V20 V21 V22
14,552,216	97.0	100.0	V01 V02 V03 V04 V05 V06 V07 V08 V10 V11 V12 V13 V14 V15 V19 V20 V21 V22
18,198,460	97.7	100.0	V01 V02 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V14 V15 V19 V21 V22
19,086,682	97.4	100.0	V01 V02 V04 V05 V08 V09 V10 V11 V12 V13 V14 V15 V16 V19 V21 V22
19,423,488	97.5	100.0	V01 V02 V04 V05 V08 V09 V10 V11 V12 V13 V15 V16 V19 V21 V22
19,906,429	97.9	100.0	V01 V02 V04 V05 V08 V09 V10 V12 V13 V15 V16 V19 V21 V22
20,196,288	98.3	100.0	V01 V02 V04 V05 V08 V09 V10 V13 V15 V16 V19 V21 V22
20,776,000	98.3	100.0	V01 V02 V05 V08 V09 V10 V13 V15 V16 V19 V21 V22
20,789,022	90.4	100.0	V01 V02 V08 V09 V10 V13 V15 V16 V19 V21 V22
21,350,859	90.0	99.7	V01 V02 V08 V09 V10 V13 V15 V16 V19 V21
23,586,212	89.1	99.2	V01 V02 V08 V09 V10 V13 V15 V16 V19
24,155,040	89.6	99.0	V01 V02 V08 V09 V10 V15 V16 V19
26,271,030	88.6	96.3	V01 V08 V09 V10 V15 V18 V19
27,804,551	88.3	96.1	V01 V08 V09 V10 V16 V19
28,455,166	88.3	95.4	V08 V09 V10 V16 V19
29,049,718	91.1	97.7	V08 V10 V16 V20
30,026,619	86.8	92.9	V10 V16 V20
30,849,706	80.5	80.5	V9 V16
39,119,363	77.6	77.6	V19

Table 8.1: Mushroom Data Set – NB & J48

Code	Actual Name	Code	Actual Name
V1	cap_shape	V13	stalk_surface_below_ring
V2	cap_surface	V14	stalk_color_above_ring
V3	cap_color	V15	stalk_color_below_ring
V4	bruises	V16	veil_type
V5	odor	V17	veil_color
V6	gill_attachment	V18	ring_number
V7	gill_spacing	V19	ring_type
V8	gill_size	V20	spore_print_color
V9	gill_color	V21	population
V10	stalk_shape	V22	habitat
V11	stalk_root	V23	class
V12	stalk_surface_above_ring		

Table 8.2: Mushroom Names and Codes

Mushroom - NB & J48

Attributes	odor	
Type	NB	J48
CSF	95.8	98.5
Wrapper	98.5	98.5

Table 8.3: Mushroom Data Set – CSF & Wrapper

Figure 8.2: Mushroom Data Set – NB

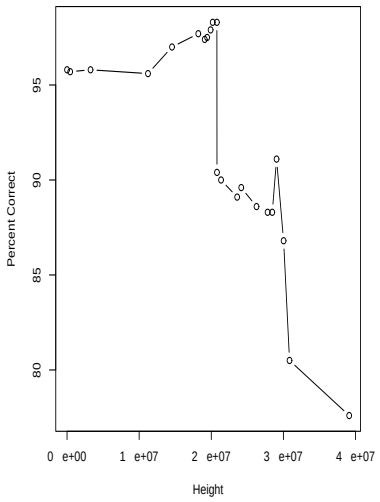
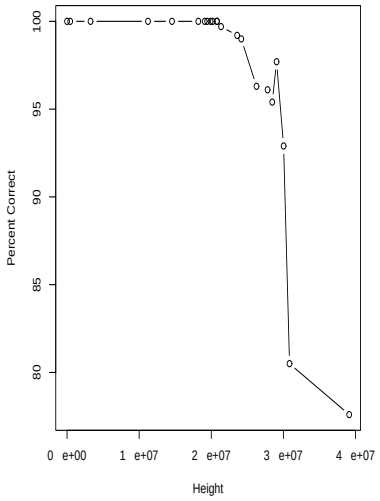


Figure 8.3: Mushroom Data Set – J48



In order to see how well this clustering technique really works, it was tried on eight other data sets. The data sets vary by the number of attributes and the number of items in each set. The data sets vary from having a moderate number of attributes, 16, to quite a few, 61. In addition, the number of items vary from 44 to 3198. Here are the results:

- Anneal
  - Number of Attributes: 38
  - Number in Data Set: 798
  - Dendrogram of attributes, Figure 8.4
  - Table of Näive Bayes and J48 at various heights, Table 8.4
  - Table of Näive Bayes and J48 for the CSF and Wrapper methods incuded in Weka, Table 8.6
  - Graph of Näive Bayes and J48 at various heights, Figures 8.5 and 8.6
  - The Barthélemy-Monjardet distance does better than the other two methods with the same number of attributes.
  - The graph of the Barthélemy-Monjardet distance suggests there is some noise in the data since the results actually degrade after seven attributes in the Näive Bayes approach.
- Hepatitis
  - Number of Attributes: 20
  - Number in Data Set: 154
  - Dendrogram of attributes, Figure 8.7
  - Table of Näive Bayes and J48 at various heights, Table 8.7

  - Table of Näive Bayes and J48 for the CSF and Wrapper methods incuded in Weka, Table 8.9
  - Graph of Näive Bayes and J48 at various heights, Figures 8.8 and 8.9
  - The Barthélemy-Monjardet distance does not do quite as well the other two. In fact, it takes most of the attributes to get good results.

- Ionosphere[4]
  - Number of Attributes: 34
  - Number in Data Set: 350
  - Dendrogram of attributes, Figure 8.10
  - Table of Näive Bayes and J48 at various heights, Table 8.10
  - Table of Näive Bayes and J48 for the CSF and Wrapper methods incuded in Weka, Table 8.11
  - Graph of Näive Bayes and J48 at various heights, Figures 8.11 and 8.12
  - The Barthélemy-Monjardet distance did about as well as the other two.

- Lymph
  - Number of Attributes: 19
  - Number in Data Set: 147
  - Dendrogram of attributes, Figure 8.13
  - Table of Näive Bayes and J48 at various heights, Table 8.12
  - Table of Näive Bayes and J48 for the CSF and Wrapper methods incuded in Weka, Table 8.14

[4]This was first discretized with $\beta = 1.8$

  - Graph of Näive Bayes and J48 at various heights, Figures 8.14 and 8.15
  - The Barthélemy-Monjardet distance does about the same as the Näive Bayes strategy but a bit worse than J48. However, note that its accuracy increases substantially with only a few attributes.

- Soybean
  - Number of Attributes: 36
  - Number in Data Set: 667
  - Dendrogram of attributes, Figure 8.16
  - Table of Näive Bayes and J48 at various heights, Table 8.15
  - Table of Näive Bayes and J48 for the CSF and Wrapper methods incuded in Weka, Table 8.17
  - Graph of Näive Bayes and J48 at various heights, Figures 8.17 and 8.18
  - The Barthélemy-Monjardet distance does not do quite as well as the other two.

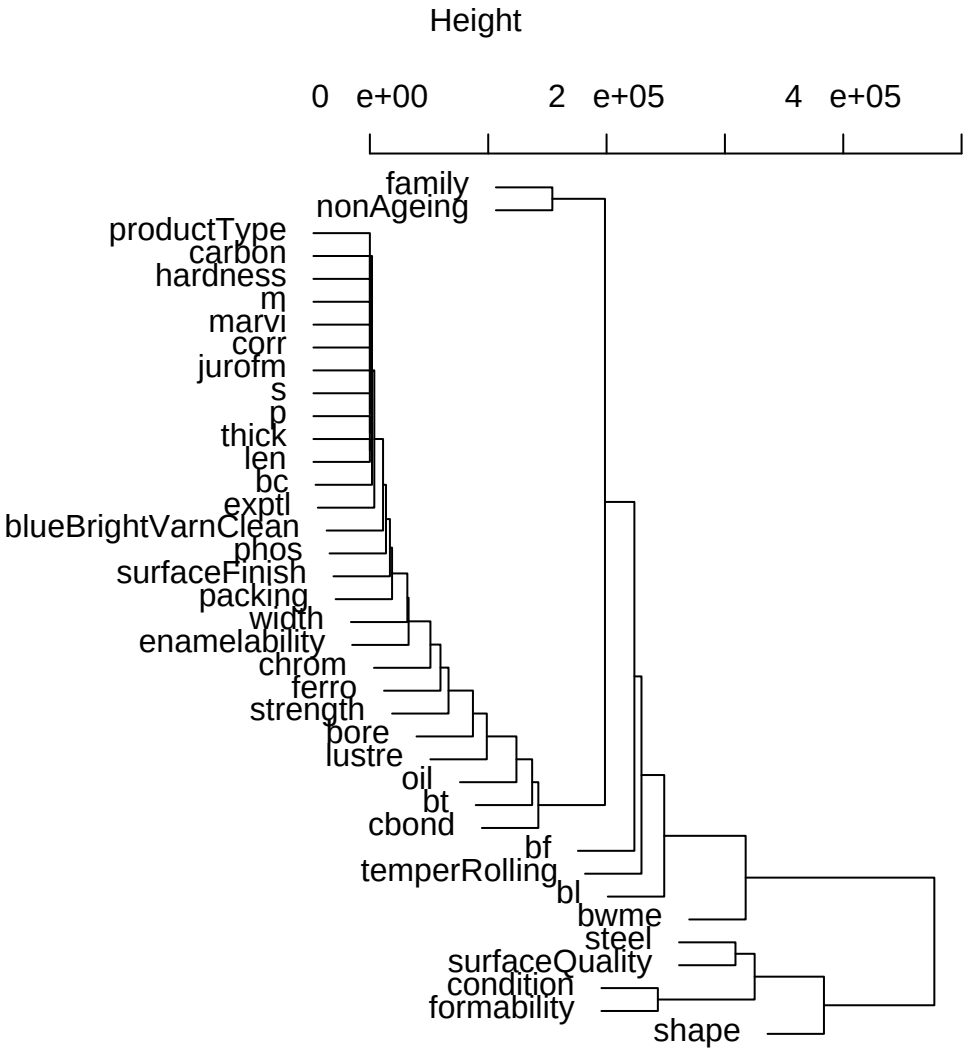
- Splice (Gene Splicing)
  - Number of Attributes: 61
  - Number in Data Set: 3189
  - There is no dendrogram nor graphs since there are so many attributes, and it would be impossible to read.
  - Table of Näive Bayes and J48 at various heights, Table 8.18
  - Table of Näive Bayes and J48 for the CSF and Wrapper methods incuded in Weka, Table 8.19

  - The Barthélemy-Monjardet distance does not do quite as well as the other two.

- Voting
  - Number of Attributes: 17
  - Number in Data Set: 434
  - Dendrogram of attributes, Figure 8.19
  - Table of Näive Bayes and J48 at various heights, Table 8.20
  - Table of Näive Bayes and J48 for the CSF and Wrapper methods incuded in Weka, Table 8.22
  - Graph of Näive Bayes and J48 at various heights, Figures 8.20 and 8.21
  - The Barthélemy-Monjardet distance does not do quite as well as the other two.

- Zoology
  - Number of Attributes: 16
  - Number in Data Set: 100
  - Dendrogram of attributes, Figure 8.22
  - Table of Näive Bayes and J48 at various heights, Table 8.23
  - Table of Näive Bayes and J48 for the CSF and Wrapper methods incuded in Weka, Table 8.25
  - Graph of Näive Bayes and J48 at various heights, Figures 8.23 and 8.24
  - The Barthélemy-Monjardet distance does about as well as the other two.

Figure 8.4: Anneal Data Set – Dendrogram



Anneal - NB & J48

Height	NB	J48	Attributes Used
0	90.2	97.0	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V14 V15 V16 V17 V18 V19 V20 V21 V22 V23 V24 V25 V26 V27 V28
1,794	90.2	96.7	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V13 V14 V15 V16 V17 V18 V19 V20 V21 V22 V23 V24 V25 V26 V27 V28
3,733	90.0	96.7	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V13 V14 V15 V16 V17 V18 V19 V21 V22 V23 V24 V25 V26 V27 V28
11,130	89.9	97.0	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V13 V14 V15 V16 V17 V18 V19 V21 V23 V24 V25 V26 V27 V28
13,606	89.7	96.9	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V13 V14 V15 V16 V17 V19 V21 V23 V24 V25 V26 V27 V28
16,932	89.9	96.8	V01 V02 V03 V04 V05 V06 V07 V08 V10 V11 V13 V14 V15 V16 V17 V19 V21 V23 V24 V25 V26 V27 V28
18,748	89.2	96.5	V01 V02 V03 V04 V05 V06 V07 V08 V10 V11 V13 V14 V15 V16 V17 V19 V21 V23 V24 V25 V26 V27
31,660	89.3	96.5	V01 V02 V03 V04 V05 V06 V07 V08 V10 V11 V13 V14 V15 V16 V17 V19 V21 V23 V24 V26 V27
32,725	89.1	96.0	V01 V02 V03 V04 V05 V06 V07 V08 V10 V13 V14 V15 V16 V17 V19 V21 V23 V24 V26 V27
51,149	89.1	96.2	V01 V02 V03 V04 V05 V06 V07 V08 V10 V13 V14 V15 V16 V19 V21 V23 V24 V26 V27
59,658	89.1	96.0	V01 V02 V03 V04 V05 V06 V07 V08 V10 V13 V14 V15 V16 V19 V23 V24 V26 V27
66,464	86.1	95.0	V01 V02 V03 V04 V05 V06 V08 V10 V13 V14 V15 V16 V19 V23 V24 V26 V27
87,086	87.2	95.1	V01 V02 V03 V04 V05 V06 V08 V10 V13 V14 V15 V16 V19 V23 V24 V26
98,810	86.7	94.6	V01 V02 V03 V04 V05 V06 V08 V10 V13 V14 V15 V16 V19 V24 V26

Anneal - NB & J48 (cont'd)

Height	NB	J48	Attributes Used
123,852	87.5	94.4	V01 V02 V03 V04 V05 V06 V08 V10 V13 V14 V15 V16 V19 V24
137,119	88.0	94.5	V01 V02 V03 V04 V05 V06 V08 V10 V13 V15 V16 V19 V24
142,355	87.8	94.5	V01 V02 V03 V04 V05 V06 V08 V10 V13 V15 V16 V24
154,176	86.6	94.3	V02 V03 V04 V05 V06 V08 V10 V13 V15 V16 V24
198,647	86.6	94.4	V02 V03 V04 V05 V06 V10 V13 V15 V16 V24
223,498	85.8	94.2	V02 V03 V04 V05 V06 V10 V15 V16 V24
229,500	84.1	94.5	V02 V03 V05 V06 V10 V15 V16 V24
243,348	91.0	93.9	V02 V03 V06 V10 V15 V16 V24
248,733	91.1	94.0	V02 V03 V06 V10 V15 V24
308,928	87.5	88.6	V02 V06 V10 V15 V24
317,545	88.2	88.6	V02 V06 V10 V24
325,139	76.3	76.3	V02 V06 V24
383,698	76.3	76.3	V02 V06
476,978	76.3	76.3	V02

Table 8.4: Anneal Data Set – NB & J48

Code	Actual Name	Code	Actual Name
V1	FAMILY	V15	BWME
V2	LEN	V16	BL
V3	STEEL	V17	CHROM
V4	TEMPERROLLING	V18	PHOS
V5	CONDITION	V19	CBOND
V6	FORMABILITY	V20	EXPTL
V7	STRENGTH	V21	FERRO
V8	NONAGEING	V22	BLUEBRIGHTVARNCLEAN
V9	SURFACEFINISH	V23	LUSTRE
V10	SURFACEQUALITY	V24	SHAPE
V11	ENAMELABILITY	V25	WIDTH
V12	BC	V26	OIL
V13	BF	V27	BORE
V14	BT	V28	PACKING

Table 8.5: Anneal Names and Codes

Anneal - NB & J48

Attributes	family, steel, temperRolling, surfaceQuality, chrom, ferro	
Type	NB	J48
CSF	89.3	91.8
Wrapper	89.3	91.8

Table 8.6: Anneal Data Set – CSF & Wrapper

Figure 8.5: Anneal Data Set – NB

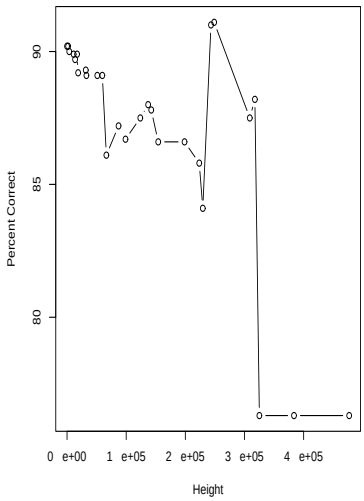


Figure 8.6: Anneal Data Set – J48

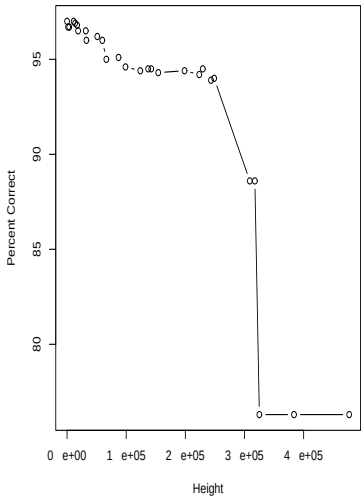
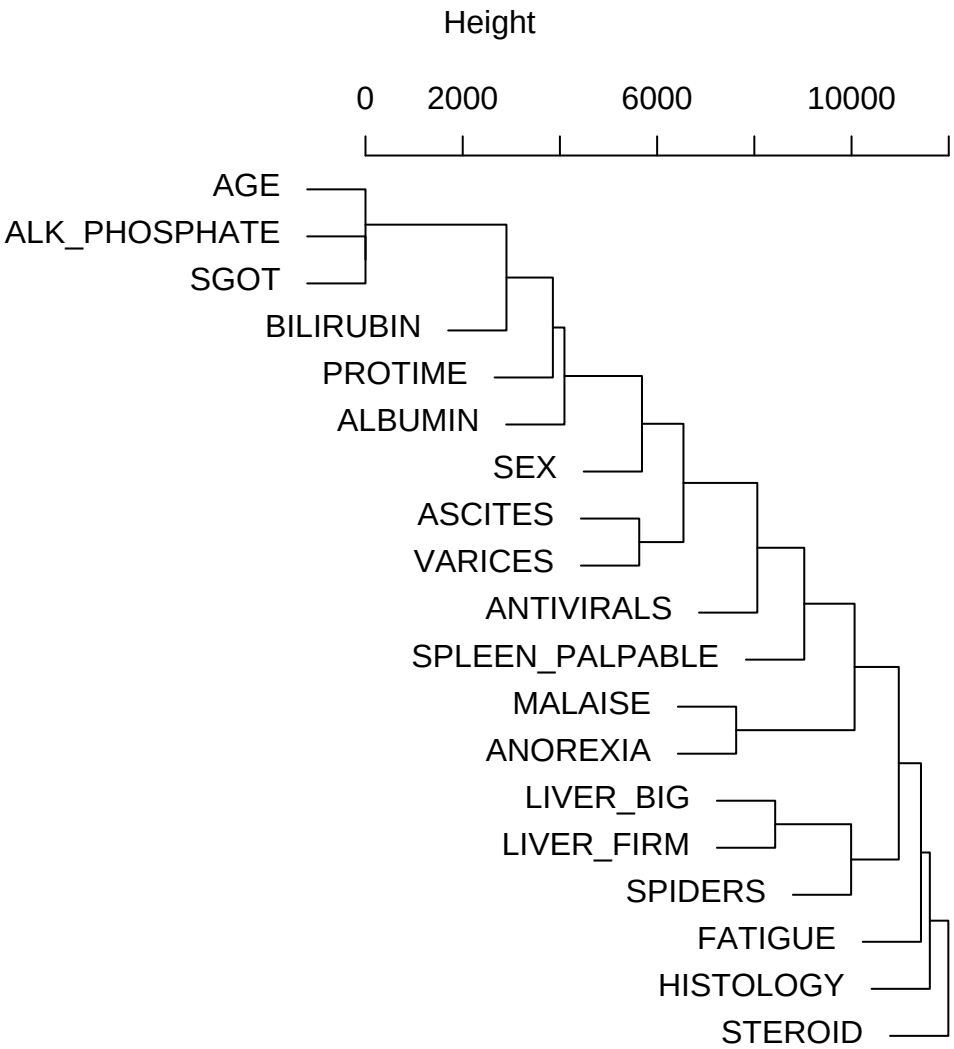


Figure 8.7: Hepatitis Data Set – Dendrogram



Hepatitis - NB & J48

Height	NB	J48	Attributes Used
0	85.1	82.5	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V14 V16 V17 V18 V19
2900	85.7	83.8	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V17 V18 V19
3978	85.1	83.8	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V17 V19
4194	83.1	84.4	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V19
5632	79.9	81.2	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V13 V16 V19
5936	79.2	81.2	V01 V03 V04 V05 V06 V07 V08 V09 V10 V11 V13 V16 V19
6534	77.9	76.0	V01 V03 V04 V05 V06 V07 V08 V09 V10 V11 V16 V19
7626	77.3	77.9	V01 V03 V04 V05 V07 V08 V09 V10 V11 V16 V19
8277	79.2	79.2	V01 V03 V05 V07 V08 V09 V10 V11 V16 V19
8448	77.3	76.6	V01 V03 V05 V07 V09 V10 V11 V16 V19
9048	77.3	79.2	V01 V03 V05 V07 V09 V11 V16 V19
10,023	76.0	79.2	V01 V03 V05 V07 V09 V16 V19
10,274	74.7	79.2	V01 V03 V05 V09 V16 V19
11,014	77.3	79.2	V01 V03 V05 V16 V19
11,451	75.3	79.2	V01 V03 V16 V19
11,590	78.6	79.2	V01 V03 V16
11,734	79.2	79.2	V03 V16
14,303	79.2	79.2	V16

Table 8.7: Hepatitis Data Set – NB & J48

Code	Actual Name	Code	Actual Name
V1	AGE	V11	SPIDERS
V2	SEX	V12	ASCITES
V3	STEROID	V13	VARICES
V4	ANTIVIRALS	V14	BILIRUBIN
V5	FATIGUE	V15	ALK_PHOSPHATE
V6	MALAISE	V16	SGOT
V7	ANOREXIA	V17	ALBUMIN
V8	LIVER_BIG	V18	PROTIME
V9	LIVER_FIRM	V19	HISTOLOGY
V10	SPLEEN_PALPABLE	V20	Class

Table 8.8: Hepatitis Names and Codes

Hepatitis - NB & J48

Attributes	age, sex, malaize, spiders, ascites, varices, bilirubin, albumin, protime, histology	
Type	NB	J48
CSF	87.7	81.3
Wrapper	85.2	80.6

Table 8.9: Hepatitis Data Set – CSF & Wrapper

Figure 8.8: Hepatitis Data Set – NB

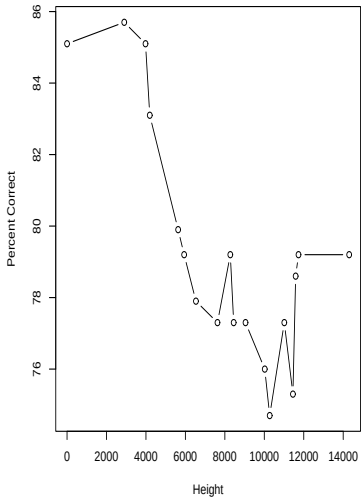


Figure 8.9: Hepatitis Data Set – J48

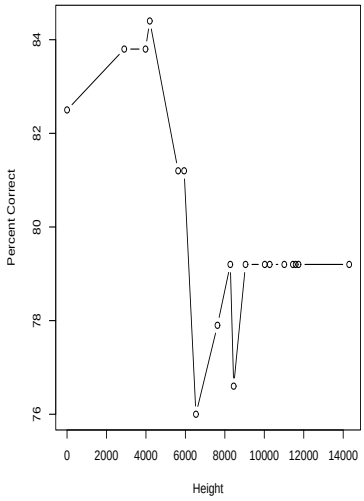
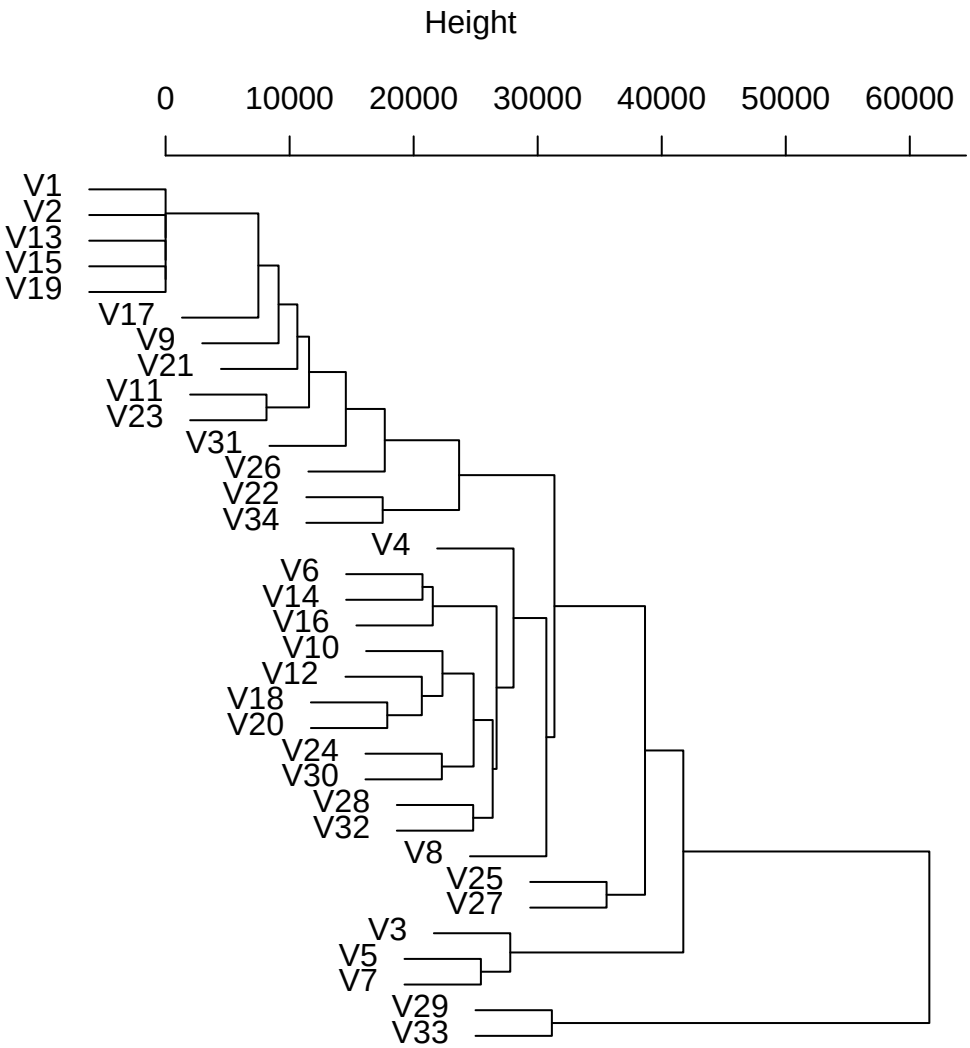


Figure 8.10: Ionosphere Data Set – Dendrogram



Ionosphere - NB & J48

Height	NB	J48	Attributes Used
0	90.9	88.1	V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V14 V16 V17 V18 V19 V20 V21 V22 V23 V24 V25 V26 V27 V28 V29 V30 V31 V32 V33 V34
7480	90.9	89.1	V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V14 V16 V18 V19 V20 V21 V22 V23 V24 V25 V26 V27 V28 V29 V30 V31 V32 V33 V34
8136	90.3	88.9	V03 V04 V05 V06 V07 V08 V09 V10 V12 V14 V16 V18 V19 V20 V21 V22 V23 V24 V25 V26 V27 V28 V29 V30 V31 V32 V33 V34
9110	90.6	88.6	V03 V04 V05 V06 V07 V08 V10 V12 V14 V16 V18 V19 V20 V21 V22 V23 V24 V25 V26 V27 V28 V29 V30 V31 V32 V33 V34
10,623	90.9	88.3	V03 V04 V05 V06 V07 V08 V10 V12 V14 V16 V18 V19 V20 V22 V23 V24 V25 V26 V27 V28 V29 V30 V31 V32 V33 V34
11,565	89.4	90.6	V03 V04 V05 V06 V07 V08 V10 V12 V14 V16 V18 V19 V20 V22 V24 V25 V26 V27 V28 V29 V30 V31 V32 V33 V34
14,530	90.6	89.4	V03 V04 V05 V06 V07 V08 V10 V12 V14 V16 V18 V19 V20 V22 V24 V25 V26 V27 V28 V29 V30 V32 V33 V34
17,502	90.6	88.9	V03 V04 V05 V06 V07 V08 V10 V12 V14 V16 V18 V19 V20 V22 V24 V25 V26 V27 V28 V29 V30 V32 V33
17,671	91.4	89.4	V03 V04 V05 V06 V07 V08 V10 V12 V14 V16 V18 V19 V20 V22 V24 V25 V27 V28 V29 V30 V32 V33
17,872	91.1	88.6	V03 V04 V05 V06 V07 V08 V10 V12 V14 V16 V19 V20 V22 V24 V25 V27 V28 V29 V30 V32 V33
20,658	91.1	90.9	V03 V04 V05 V06 V07 V08 V10 V14 V16 V18 V19 V22 V24 V25 V27 V28 V29 V30 V32 V33

Ionosphere - NB & J48 (cont'd)

Height	NB	J48	Attributes Used
20,741	91.7	88.3	V03 V04 V05 V07 V08 V10 V14 V16 V18 V19 V22 V24 V25 V27 V28 V29 V30 V32 V33
21,541	91.7	87.7	V03 V04 V05 V06 V07 V08 V10 V18 V19 V22 V24 V25 V27 V28 V29 V30 V32 V33
22,272	91.7	88.6	V03 V04 V05 V06 V07 V08 V10 V18 V19 V22 V24 V25 V27 V28 V29 V32 V33
22,324	91.4	88.9	V03 V04 V05 V06 V07 V08 V19 V20 V22 V24 V25 V27 V28 V29 V32 V33
23,663	92.3	89.4	V03 V04 V05 V06 V07 V08 V19 V20 V24 V25 V27 V28 V29 V32 V33
24,800	92.3	89.7	V03 V04 V05 V06 V07 V08 V19 V20 V24 V25 V27 V29 V32 V33
24,833	92.6	89.1	V03 V04 V05 V06 V07 V08 V18 V19 V25 V27 V29 V32 V33
25,420	90.0	90.0	V03 V04 V06 V07 V08 V18 V19 V25 V27 V29 V32 V33
26,371	90.6	90.3	V03 V04 V06 V07 V08 V18 V19 V25 V27 V29 V33
26,685	91.4	90.6	V03 V04 V07 V08 V18 V19 V25 V27 V29 V33
27,788	89.4	90.0	V04 V05 V08 V18 V19 V25 V27 V29 V33
28,053	90.0	91.4	V05 V08 V18 V19 V25 V27 V29 V33
30,699	89.4	89.7	V05 V18 V19 V25 V27 V29 V33
31,144	88.3	89.4	V05 V18 V19 V25 V27 V33
31,350	88.0	91.1	V05 V19 V25 V27 V33
35,554	89.4	87.1	V05 V19 V25 V33
38,652	88.3	86.3	V05 V17 V33
41,740	84.3	84.3	V17 V33
61,576	67.6	67.6	V21

Table 8.10: Ionosphere Data Set – NB & J48

Ionosphere - NB & J48

Attributes	Both: V03, V04, V05, V07, V08, V14, V27, V28, V33 NB also: V35	
Type	NB	J48
CSF	93.7	90.3
Wrapper	91.2	89.7

Table 8.11: Ionosphere Data Set – CSF & Wrapper

Figure 8.11: Ionosphere Data Set – NB

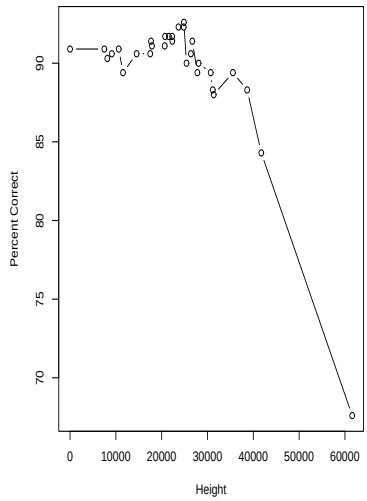


Figure 8.12: Ionosphere Data Set – J48

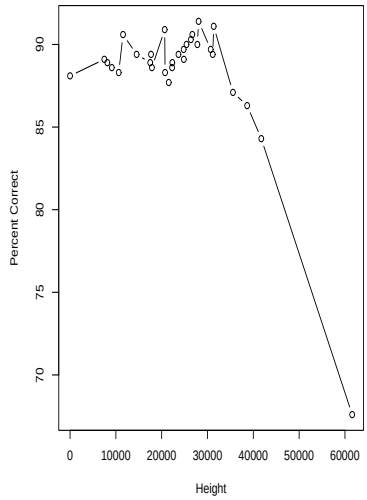
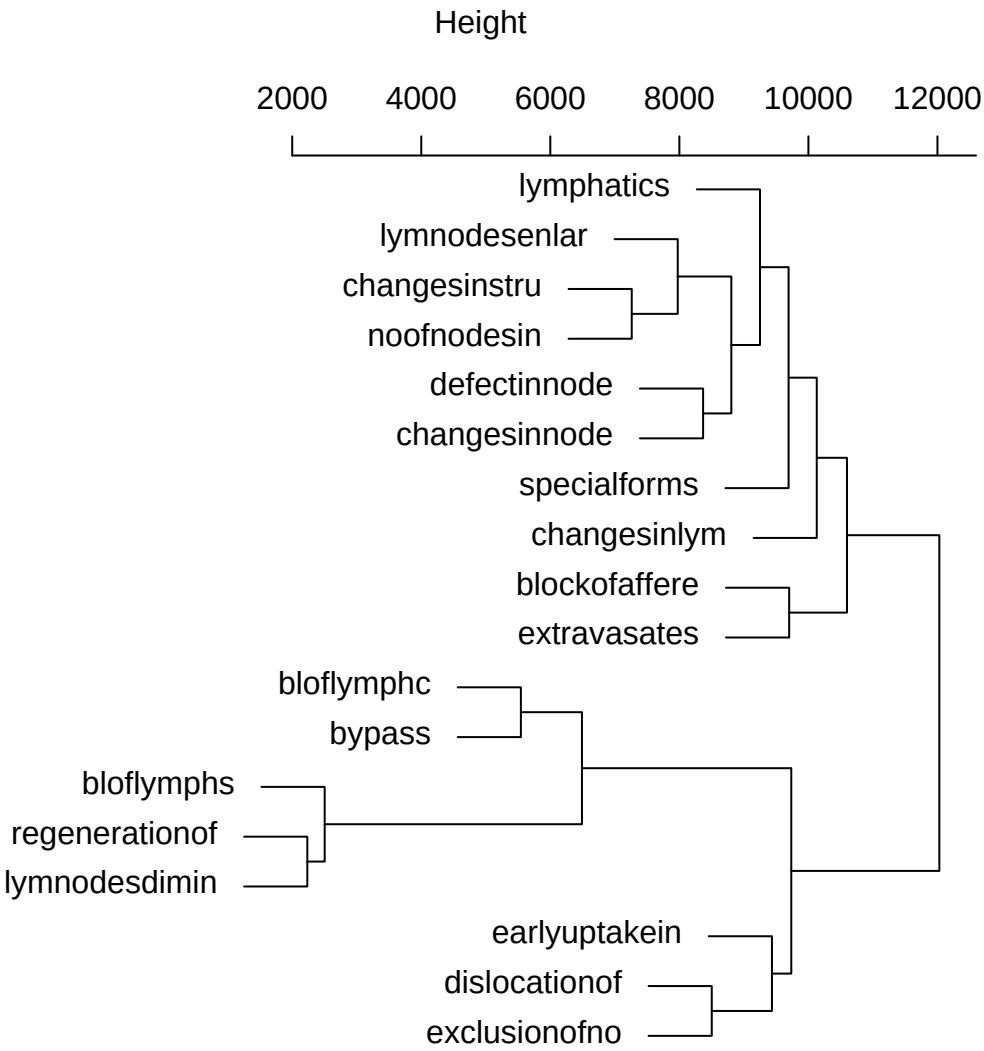


Figure 8.13: Lymph Data Set – Dendrogram



Lymph - NB & J48

Height	NB	J48	Attributes Used
0	85.7	80.3	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V14 V15 V16 V17 V18
2234	85.0	77.6	V01 V02 V03 V04 V05 V06 V08 V09 V10 V11 V12 V13 V14 V15 V16 V17 V18
2503	84.4	77.6	V01 V02 V03 V05 V06 v07 V08 V10 V11 V12 V13 V14 V15 V16 V17 V18
5544	86.4	76.9	V01 V02 V05 V06 V07 V08 V10 V11 V12 V13 V14 V15 V16 V17 V18
6491	84.4	77.6	V01 V02 V04 V06 V08 V10 V11 V12 V13 V14 V15 V16 V17 V18
7262	80.3	78.2	V01 V02 V04 V06 V08 V10 V11 V12 V13 V15 V16 V17 V18
7975	83.0	80.3	V01 V02 V04 V06 V08 V11 V12 V13 V15 V16 V17 V18
8368	83.7	78.9	V01 V02 V04 V06 V08 V11 V13 V15 V16 V17 V18
8502	84.4	81.0	V01 V02 V04 V06 V08 V11 V13 V15 v17 V18
8805	83.7	70.1	V01 V02 V04 V06 V08 V11 V15 v17 V18
9250	81.6	71.4	V02 V04 V06 V08 V11 V15 v17 V18
9435	76.2	71.4	V02 V04 V06 V11 V15 v17 V18
9693	70.7	70.1	V02 V04 V06 V11 V14 V17
9702	61.9	62.6	V04 V06 V11 V14 V17
9735	61.9	62.6	V04 V06 V11 V14
10,129	53.7	53.1	V04 V06 V14
10,598	57.8	50.3	V04 V14
12,029	55.1	55.1	V05

Table 8.12: Lymph Data Set – NB & J48

Code	Actual Name	Code	Actual Name
V1	LYMPHATICS	V10	LYMNODESENLAR
V2	BLOCKOFAFFERE	V11	CHANGESINLYM
V3	BLOFLYMPHC	V12	DEFECTINNODE
V4	BLOFLYMPHS	V13	CHANGESINNODE
V5	BYPASS	V14	CHANGESINSTRU
V6	EXTRAVASATES	V15	SPECIALFORMS
V7	REGENERATIONOF	V16	DISLOCATIONOF
V8	EARLYUPTAKEIN	V17	EXCLUSIONOFNO
V9	LYMNODESDIMIN	V18	NOOFNODESIN
		V19	class

Table 8.13: Lymph Names and Codes

Lymph - NB & J48

Attributes	lymphatics, blockofaffere, regenerationof, earlyuptakein, lymnondesdimin, changesinode, specialforms, noofnodesin	
Type	NB	J48
CSF	81.6	78.9
Wrapper	81.6	78.9

Table 8.14: Lymph Data Set – CSF & Wrapper

Figure 8.14: Lymph Data Set – NB

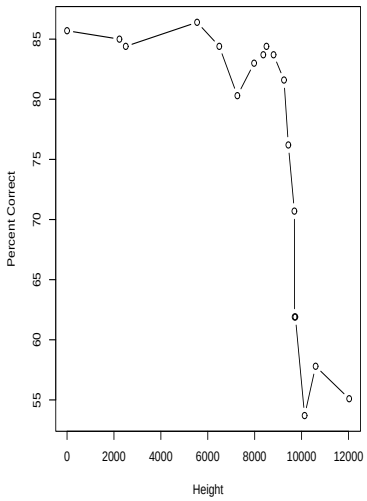


Figure 8.15: Lymph Data Set – J48

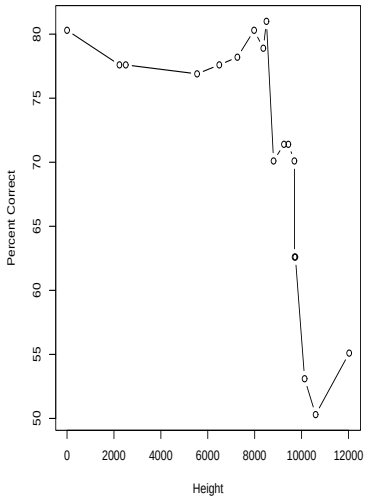
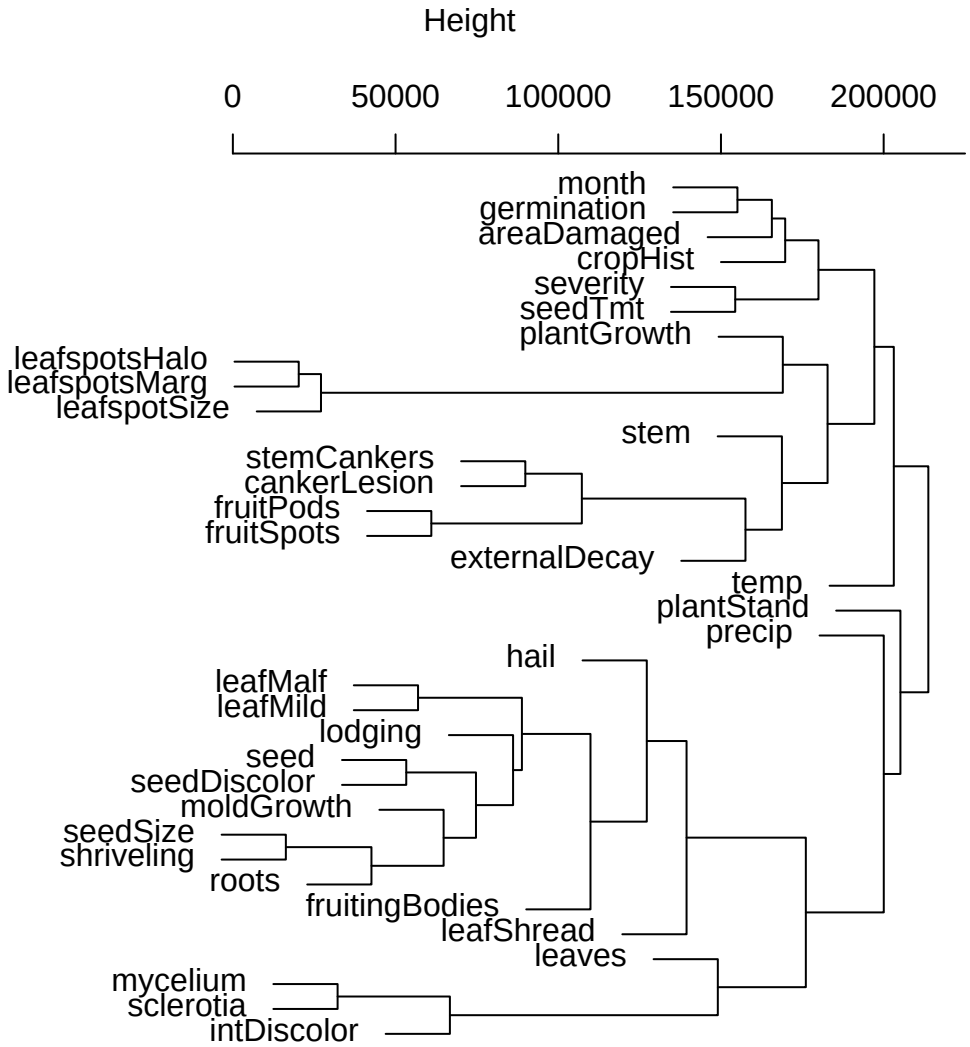


Figure 8.16: Soybean Data Set – Dendrogram



Soybean - NB & J48

Height	NB	J48	Attributes Used
0	92.4	91.6	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V14 V15 V16 V17 V18 V19 V20 V21 V22 V23 V24 V25 V26 V27 V28 V29 V30 V31 V32 V33 V34 V35
15,538	92.2	91.3	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V14 V15 V16 V17 V18 V19 V20 V21 V22 V23 V24 V25 V26 V27 V28 V29 V31 V32 V33 V34 V35
20,242	92.5	92.1	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V14 V15 V16 V17 V18 V19 V20 V21 V22 V23 V24 V25 V26 V27 V28 V29 V30 V31 V32 V34 V35
27,081	90.6	91.9	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V17 V18 V19 V20 V21 V22 V23 V24 V25 V26 V27 V28 V29 V31 V32 V33 V34 V35
31,408	90.4	92.1	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V17 V18 V19 V20 V21 V22 V23 V24 V26 V27 V28 V29 V31 V32 V33 V34 V35
41,109	89.8	89.4	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V17 V18 V19 V20 V21 V22 V23 V24 V26 V27 V28 V29 V31 V32 V33 V34
48,748	90.3	87.9	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V17 V18 V19 V20 V21 V22 V23 V24 V26 V27 V29 V31 V32 V33 V34
56,188	88.0	86.2	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V18 V19 V20 V21 V22 V23 V24 V26 V27 V29 V31 V32 V33 V34
58,864	87.9	85.9	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V18 V19 V20 V21 V22 V23 V24 V26 V27 V29 V31 V32 V34
63,179	87.9	87.1	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V18 V19 V20 V21 V22 V23 V24 V26 V27 V29 V32 V34
64,984	87.7	89.4	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V18 V19 V20 V21 V22 V23 V24 V27 V29 V32 V34

Soybean - NB & J48 (cont'd)

Height	NB	J48	Attributes Used
71,178	87.1	89.4	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V18 V19 V20 V21 V22 V23 V24 V27 V29 V34
81,533	85.9	86.2	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V18 V19 V21 V22 V23 V24 V27 V29 V34
85,362	84.3	82.8	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V19 V21 V22 V23 V24 V27 V29 V34
87,390	84.3	83.1	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V19 V22 V23 V24 V27 V29 V34
95,055	82.6	81.1	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V19 V23 V24 V27 V29 V34
105,706	81.9	81.7	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V16 V19 V24 V27 V29 V34
123,718	81.9	79.9	V01 V02 V03 V04 V06 V07 V08 V09 V10 V11 V12 V13 V16 V19 V24 V27 V29 V34
132,592	81.0	79.6	V01 V02 V03 V04 V06 V07 V08 V09 V10 V11 V13 V16 V19 V24 V27 V29 V34
136,612	79.9	78.4	V01 V02 V03 V04 V06 V07 V08 V09 V10 V11 V13 V19 V24 V27 V29 V34
146,972	79.2	79.0	V01 V02 V03 V04 V06 V07 V08 V09 V10 V11 V13 V22 V24 V27 V34
148,514	77.5	77.1	V02 V03 V04 V06 V07 V08 V09 V10 V11 V13 V22 V24 V27 V34
154,162	76.6	78.1	V02 V03 V04 V06 V07 V08 V09 V10 V14 V22 V24 V27 V34
154,388	73.8	76.2	V02 V03 V04 V06 V07 V09 V10 V14 V22 V24 V27 V34
156,382	72.0	70.2	V02 V03 V04 V06 V07 V09 V10 V14 V27 V29 V34
158,641	74.2	70.5	V01 V02 V03 V04 V06 V09 V14 V27 V29 V34
162,393	72.9	71.4	V01 V02 V03 V04 V09 V14 V27 V29 V34
162,604	72.3	72.0	V01 V02 V03 V04 V09 V14 V29 V34

Soybean - NB & J48 (cont'd)

Height	NB	J48	Attributes Used
173,288	68.2	68.4	V02 V03 V04 V10 V14 V29 V34
173,546	62.4	56.2	V02 V03 V04 V10 V29 V34
188,903	62.1	55.8	V02 V03 V04 V29 V34
189,156	56.8	57.3	V02 V04 V29 V34
195,223	44.4	40.2	V02 V29 V34
196,468	31.9	31.9	V29 V34
205,659	16.8	16.8	V34

Table 8.15: Soybean Data Set – NB & J48

Code	Actual Name	Code	Actual Name
V1	month	V19	stem
V2	plant stand	V20	lodging
V3	precip	V21	stemcankers
V4	temp	V22	cankerlesion
V5	hail	V23	fruiting bodies
V6	crop hist	V24	external decay
V7	area damaged	V25	mycelium
V8	severity	V26	int discolor
V9	seed tmt	V27	sclerotia
V10	germination	V28	fruitpods
V11	plant growth	V29	fruitspots
V12	leaves	V30	seed
V13	leaf spots halo	V31	mold growth
V14	leaf spots marg	V32	seed discolor
V15	leaf spot size	V33	seed size
V16	leaf shread	V34	shriveling
V17	leaf malf	V35	roots
V18	leaf mild	V36	class

Table 8.16: Soybean Names and Codes

Soybean - NB & J48

Attributes	leafspotsize, leafmalt, leafmild, stem, cankerlesion, fruittingbodies, externaldecay	
Type	NB	J48
CSF	92.1	90.4
Wrapper	92.4	90.3

Table 8.17: Soybean Data Set – CSF & Wrapper

Figure 8.17: Soybean Data Set – NB

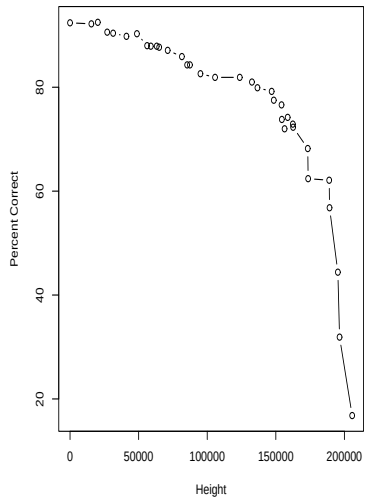
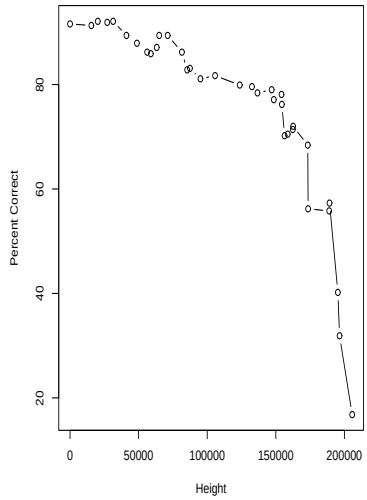


Figure 8.18: Soybean Data Set – J48



Splice - NB & J48

Height	NB	J48	Attributes Used
0	95.4	94.3	
3,554,098	93.8	90.6	
3,703,528	93.0	90.8	
3,724,774	93.0	91.1	
3,725,600	92.5	90.3	01 02 03 04 05 06 07 08 09 10 11 12 14 15 16 17 18 19 20 21 22 23 24 26 27 28 30 31 32 33 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60
3,726,910	92.7	90.3	
3,730,546	92.5	90.7	
3,732,684	93.0	90.6	
3,734,102	92.8	90.7	
3,734,330	93.1	90.6	01 02 03 04 05 06 07 08 09 10 11 12 14 15 16 17 18 19 20 21 22 23 24 26 27 28 30 31 32 33 35 36 37 38 39 41 42 43 44 46 48 49 50 51 52 53 54 55 57 59 60
3,734,904	92.9	90.1	
3,736,416	93.0	90.1	
3,736,510	92.9	90.7	
3,739,256	92.9	90.3	
3,740,794	93.1	90.2	02 03 04 06 07 08 09 10 11 12 14 15 16 17 18 19 20 22 23 24 26 27 28 30 31 32 33 35 36 37 39 41 42 43 44 46 48 49 51 52 53 54 55 57 59 60
3,744,322	93.0	90.1	
3,747,788	93.1	90.1	
3,750,280	92.8	89.9	
3,751,984	93.0	90.4	
3,755,524	92.8	90.1	02 04 06 07 09 11 12 14 15 16 17 18 19 20 22 24 26 27 28 30 31 32 33 35 36 37 39 41 42 44 46 48 49 51 52 53 54 55 57 59 60

Splice - NB & J48

Height	NB	J48	Attributes Used
3,757,816	92.7	90.1	
3,764,530	92.3	90.0	
3,764,930	92.6	90.4	
3,772,681	92.6	90.3	
3,774,922	92.4	90.2	02 04 06 07 09 11 12 14 15 16 17 19 20 22 24 26 27 28 30 31 32 33 35 37 39 41 42 44 46 48 49 51 54 56 59 60
3,778,748	92.4	90.6	
3,784,344	92.8	90.7	
3,785,188	92.6	90.7	
3,785,865	92.9	90.4	
3,788,298	92.9	90.6	02 04 06 09 11 14 16 17 19 20 22 24 26 27 28 30 31 32 33 35 37 39 41 43 46 48 49 51 54 56 59
3,792,236	91.2	88.5	
3,793,883	91.3	89.1	
3,795,345	91.4	88.8	
3,795,419	91.5	88.9	
3,797,620	91.3	89.2	02 04 06 09 11 14 17 19 22 24 26 27 28 30 31 32 34 37 40 43 46 48 51 54 56 59
3,800,189	91.2	89.2	
3,800,470	91.6	89.3	
3,802,684	91.5	88.4	
3,804,723	91.4	88.6	
3,806,902	91.4	89.1	02 06 10 14 17 19 22 24 26 28 30 31 32 34 39 40 45 48 51 56 59
3,808,732	91.3	88.8	
3,813,082	91.4	88.9	

Splice - NB & J48

Height	NB	J48	Attributes Used
3,814,390	91.3	88.9	
3,814,712	89.7	89.7	
3,814,724	90.8	88.9	02 13 17 19 22 26 28 30 31 32 34 37 44 48 51 56
3,815,043	91.2	88.8	
3,817,437	91.3	89.0	
3,818,000	91.1	89.1	
3,818,957	91.3	89.0	
3,822,777	91.0	88.8	13 18 22 26 28 30 31 32 34 37 50
3,823,863	90.9	88.6	
3,829,874	90.0	89.3	18 25 28 30 31 32 34 37 50
3,831,247	90.2	89.2	18 25 28 30 31 32 34 50
3,837,327	89.6	88.0	25 28 30 31 32 34 50
3,839,485	87.6	87.1	28 30 31 32 34 50
3,862,554	84.5	84.9	28 30 31 35 50
3,941,893	76.9	77.5	28 30 31 50
3,984,256	70.5	70.4	30 31 50
4.159,405	70.5	70.4	30 50
4,310,989	52.0	51.9	50

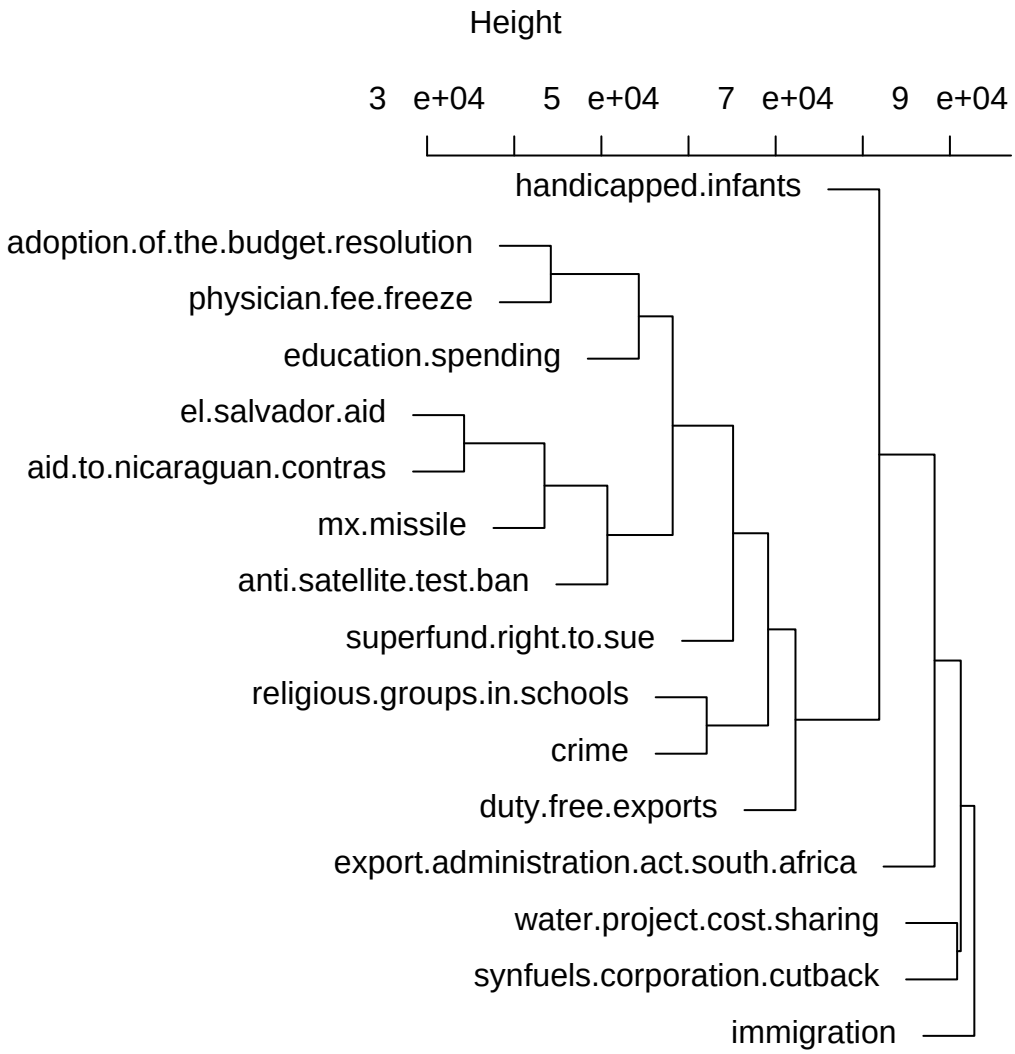
Table 8.18: Splice Data Set – NB & J48

Splice - NB & J48

Attributes	28, 29, 30, 31, 32, 35	
Type	NB	J48
CSF	93.6	93.3
Wrapper	93.6	93.3

Table 8.19: Splice Data Set – CSF & Wrapper

Figure 8.19: Vote Data Set – Dendrogram



Voting Data - NB & J48

Height	NB	J48	Attrs Offered
0	90.3	95.9	V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12 V13 V14 V15 V16
34,244	89.9	96.8	V01 V02 V03 V04 V06 V07 V08 V09 V10 V11 V12 V13 V14 V15 V16
43,472	91.0	96.1	V01 V02 V03 V04 V05 V06 V07 V10 V11 V12 V13 V14 V15 V16
44,196	91.7	94.9	V01 V02 V04 V05 V06 V07 V10 V11 V12 V13 V14 V15 V16
50,687	92.2	94.7	V01 V02 V04 V05 V06 V10 V11 V12 V13 V14 V15 V16
54,296	92.4	94.9	V01 V02 V04 V05 V06 V10 V11 V13 V14 V15 V16
58,182	84.6	86.9	V01 V02 V06 V08 V10 V11 V13 V14 V15 V16
62,084	85.7	87.8	V01 V02 V08 V10 V11 V13 V14 V15 V16
65,117	86.2	89.4	V01 V02 V05 V10 V11 V14 V15 V16
69,136	89.9	89.6	V01 V02 V05 V10 V11 V15 V16
72,276	86.6	88.7	V01 V02 V05 V10 V11 V16
81,889	88.5	88.9	V02 V05 V10 V11 V16
88,240	88.7	88.9	V02 V05 V10 V11
90,828	84.8	84.8	V05 V10 V11
91,244	84.8	84.8	V05 V10
92,803	84.8	84.8	V05

Table 8.20: Voting Data Set – NB & J48

Code	Actual Name	Code	Actual Name
V01	handicapped.infants	V09	mx.missile
V02	water.project.cost.sharing	V10	immigration
V03	adoption.of.the.budget.resolution	V11	synfuels.corporation.cutback
V04	physician.fee.freeze	V12	education.spending
V05	el.salvador.aid	V13	superfund.right.to.sue
V06	religious.groups.in.schools	V14	crime
V07	anti.satellite.test.ban	V15	duty.free.exports
V08	aid.to.nicaraguan.contras	V16	export.admin.act.south.africa

Table 8.21: Voting Names and Codes

Voting - NB & J48

Attributes	physician_fee_freeze	
Type	NB	J48
CSF	95.6	95.6
Wrapper	95.6	95.6

Table 8.22: Voting Data Set – CSF & Wrapper

Figure 8.20: Voting Data Set – NB

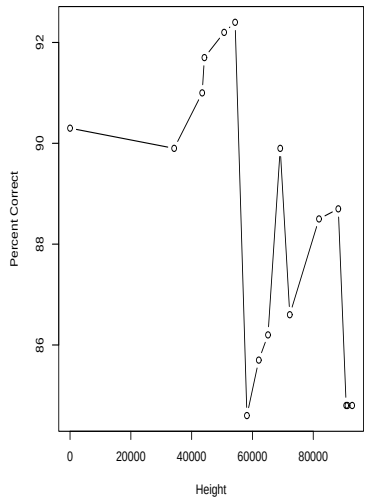


Figure 8.21: Voting Data Set – J48

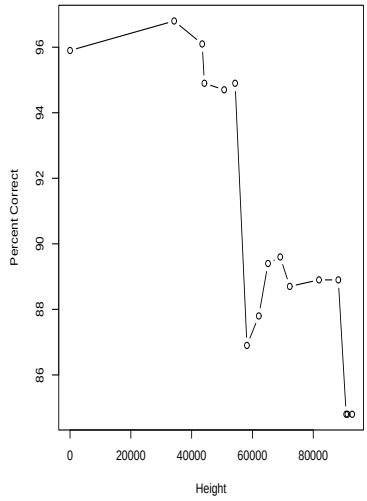
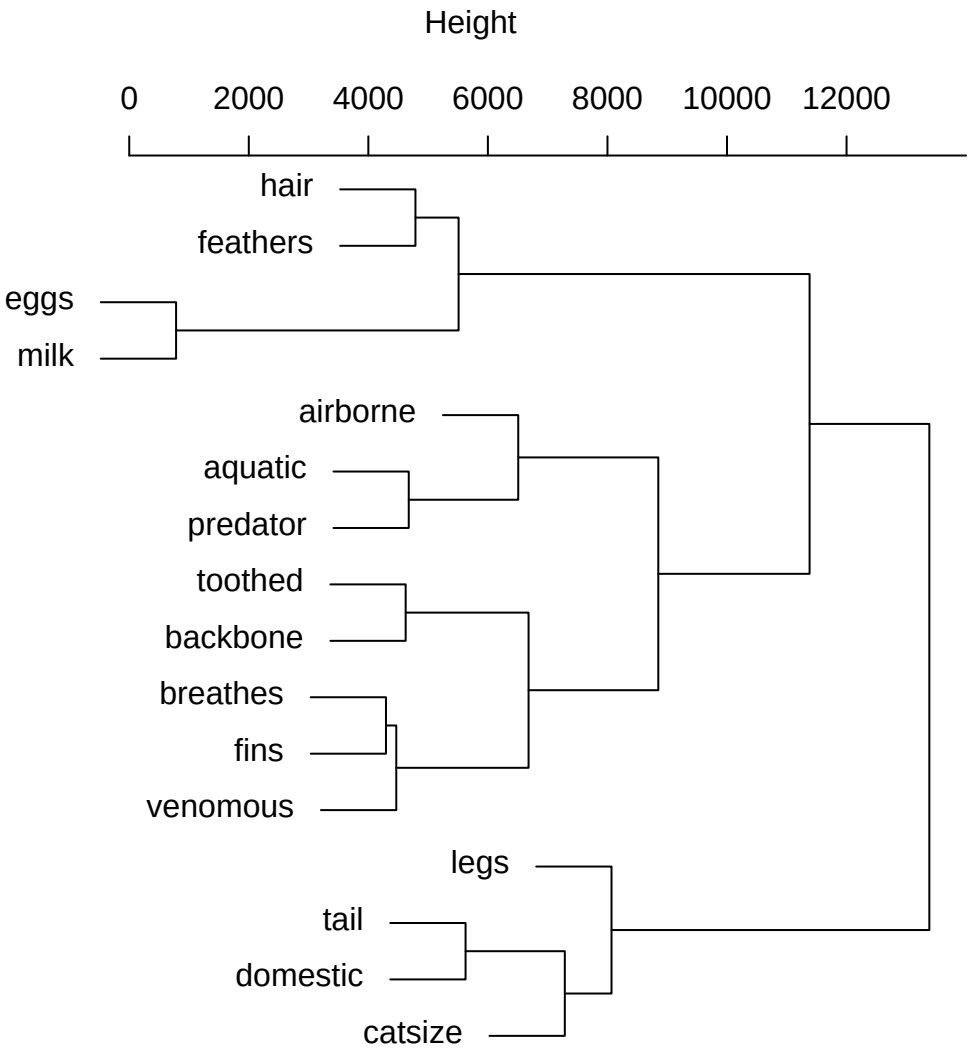


Figure 8.22: Zoo Data Set – Dendrogram



Zoo - Naive Bayes

Height	Naïve Bayes	J48	Attrs Used
0	93.1	92.1	V2 V3 V4 V5 V6 V7 V8 V9 V10 V11 V12 V13 V14 V15 V16 V17
588	93.1	93.1	V2 V3 V4 V6 V7 V8 V9 V10 V11 V12 V13 V14 V15 V16 V17
1398	93.1	93.1	V5 V3 V6 V7 V8 V9 V10 V11 V12 V13 V14 V15 V16 V17
1820	95.0	94.1	V5 V3 V6 V7 V8 V9 V10 V11 V12 V13 V14 V16 V17
2136	94.1	94.1	V5 V3 V7 V11 V8 V9 V10 V12 V14 V16 V17
2986	91.1	91.1	V5 V3 V11 V8 V9 V10 V12 V14 V16 V17
3116	91.1	91.1	V5 V3 V11 V8 V9 V10 V12 V14 V17
3240	93.1	91.1	V5 V3 V11 V8 V10 V12 V14 V17
3684	89.1	90.1	V5 V3 V11 V8 V10 V12 V14
3888	93.1	89.1	V5 V3 V11 V8 V10 V14
4296	89.1	90.1	V5 V3 V11 V8 V10
4421	72.3	87.1	V5 V10 V11 V8
4671	57.4	75.2	V5 V10 V8
4862	60.4	60.4	V5 V8
4948	60.4	60.4	V5

Table 8.23: Zoo Data Set – NB & J48

Code	Actual Name	Code	Actual Name
V02	hair	V10	backbone
V03	feathers	V11	breathes
V04	eggs	V12	venomous
V05	milk	V13	fins
V06	airborne	V14	legs
V07	aquatic	V15	tail
V08	predator	V16	domestic
V09	toothed	V17	catsize

Table 8.24: Zoo Names and Codes

Zoo - NB & J48

Attributes	hair, feathers, milk, toothed, backbone, breathes, fins, legs, tail	
Type	NB	J48
CSF	94	91
Wrapper	94	91

Table 8.25: Zoo Data Set – CSF & Wrapper

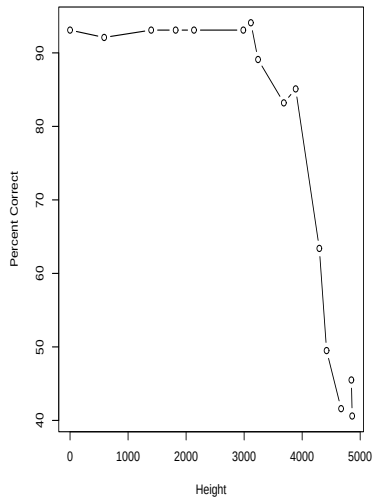


Figure 8.23: Zoo - NB

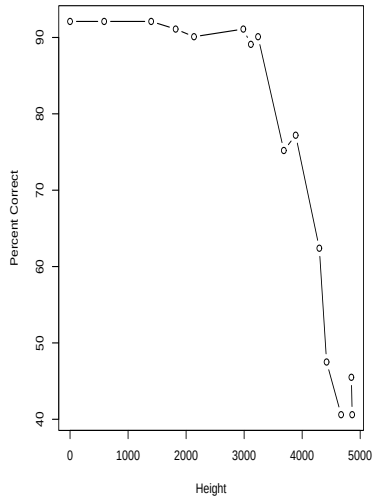


Figure 8.24: Zoo - J48

8.2 Results on Microarray Data Sets

As stated in [GE03], in early studies of relevance published in the late 90s [BL97, KJ97], few applications explored data with more than 40 attributes. With the increased interest of data miners in biocomputing in general, and in microarray data in particular, classification problems that involve thousands of features and relatively few examples came to the fore.

The the Barthélemy-Monjardet distance technique was applied to the Golub data set [GST99] that contains a training set of 38 instances and 7130 attributes. After discretization using the algorithm developed in [BFO98], there were 830 attributes that contained more than one value. These attributes were clustered using the k-means algorithm [HW79]. The Barthélemy-Monjardet distance was used to determine the central attribute within each cluster. Then, as before, hierarchical clustering was done. The results obtained on the training and the test set (that includes 34 instances) are shown in Tables 8.26 and 8.27, respectively.

No. of Clusters	Naïve Bayes	J48
all 829	100	89.5
128	100	84.2
64	100	89.5
32	100	94.7
16	97.4	89.5
8	97.4	86.8
4	97.4	92.1

Table 8.26: Golub Data Set – Training Set

No. of Clusters	Naïve Bayes	J48
128	97.1	94.1
64	97.1	97.1
32	100	70.6
16	94.1	85.3
8	94.1	94.1
4	91.2	91.2

Table 8.27: Golub Data Set – Test Set

8.3 Side Results

Not only were good results obtained using the "representative" attributes, the clusters themselves might be informative to users to examine to see which attributes clustered and which attribute was chosen as the "representative" one.

8.3.1 Hepatitis

In the hepatitis data set, SGOT was clearly a highly representative attribute, as evidenced by the results shown below in tables of the clustering at several different heights, 8.28 and 8.29.

Hepatitis Clustering

Cluster	Rep Att	Atts
1	STEROID	STEROID AGE
2	SGOT	HISTOLOGY PROTIME ALBUMIN SGOT ALKPHOSPHATE BILIRUBIN VARICES ASCITES SPIDERS SPLEENPALPABLE LIVERFIRM LIVERBIG ANOREXIA MALAISE FATIGUE ANTIVIRALS SEX

Table 8.28: Hepatitis Data Set – 14,000

Hepatitis Clustering

Cluster	Rep Att	Atts
1	AGE	
2	SGOT	PROTIME ALBUMIN SGOT ALKPHOSPHATE BILIRUBIN VARICES ASCITES SPIDERS SPLEENPALPABLE ANOREXIA MALAISE ANTIVIRALS SEX
3	STEROID	
5	FATIGUE	
8	LIVERFIRM	LIVERFIRM LIVERBIG
19	HISTOLOGY	

Table 8.29: Hepatitis Data Set – 11,000

8.3.2 Voting

Similarly, for Voting el_salvador_aid was a good indicator of many other attributes, see Tables 8.30, 8.31, and 8.32.

Voting Clustering

Cluster	Rep Att	Atts
1	EL_SALVADOR_AID	EXPORT_SOUTH_AFRICA DUTY_FREE_EXPORTS CRIME SUPERFUND_RIGHT_TO_SUE EDUCATION_SPENDING MX_MISSILE AID_TO_NICARAGUAN_CONTRAS ANTI_SATELLITE_TEST_BAN RELIGIOUS_GROUPS_IN_SCHOOLS EL_SALVADOR_AID PHYSICIAN_FEE_FREEZE ADOPTION_BUDGET_RESOLUTION HANDICAPPED_INFANTS
2	SYNFUELS_CORPORATION_CUTBACK	SYNFUELS_CORPORATION_CUTBACK WATER_PROJECT_COST_SHARING

Table 8.30: Voting Data Set – 90,830

Voting Clustering

Cluster	Rep Att	Atts
1	EL_SALVADOR_AID	EXPORT_SOUTH_AFRICA DUTY_FREE_EXPORTS CRIME SUPERFUND_RIGHT_TO_SUE EDUCATION_SPENDING MX_MISSILE AID_TO_NICARAGUAN_CONTRAS ANTI_SATELLITE_TEST_BAN RELIGIOUS_GROUPS_IN_SCHOOLS EL_SALVADOR_AID PHYSICIAN_FEE_FREEZE ADOPTION_BUDGET_RESOLUTION HANDICAPPED_INFANTS
2	WATER_PROJECT_COST_SHARING	
10	IMMIGRATION	
11	SYNFUELS_CORPORATION_CUTBACK	

Table 8.31: Voting Data Set – 88,250

Voting Clustering

Cluster	Rep Att	Atts
1	HANDICAPPED_INFANTS	
2	WATER_PROJECT_COST_SHARING	
3	EL_SALVADOR_AID	SUPERFUND_RIGHT_TO_SUE EDUCATION_SPENDING MX_MISSILE AID_TO_NICARAGUAN_CONTRAS ANTI_SATELLITE_TEST_BAN EL_SALVADOR_AID PHYSICIAN_FEE_FREEZE ADOPTION_BUDGET_RESOLUTION
6	CRIME	CRIME RELIGIOUS_GROUPS_IN_SCHOOLS
10	IMMIGRATION	
11	SYNFUELS_CORPORATION_CUTBACK	
15	DUTY_FREE_EXPORTS	
16	EXPORT_SOUTH_AFRICA	

Table 8.32: Voting Data Set – 65,200

CHAPTER 9

Clustering – Theory

Blore's Razor: Given a choice between two theories, take the one which is funnier.

– Anonymous

Mathematics consists of proving the most obvious thing in the least obvious way.

–GeorgePólya

9.1 Introduction

Here is another theorem showing a relationship between Generalized Barthélemy-Monjardet distance function and Pearson's coefficient.

9.2 Relationship Between The Distance Function and Pearson's Co-efficient

A proof that the distance function is related to Pearson's Co-efficient in that as the distance increases Pearson's Co-efficient decreases, as would be expected. Note, for simplicity, the investigation we will be limited to when $\beta = 2$.

Theorem 9.2.1 *Claim:* $\frac{1}{\pi_+\sigma_+} \leq \chi^2 \leq \frac{1}{\pi_-\sigma_-}\left[\|\pi\|^2 + \|\sigma\|^2 - d(\pi,\sigma)\right]$

Proof:

As a preliminary, note that:

$$\begin{aligned}
\chi^2 &= \sum_i \sum_j \frac{\left(\frac{n_{ij}}{N} - \frac{n_{i+}n_{+j}}{N^2}\right)^2}{\frac{n_{i+}n_{+j}}{N^2}} \\
&= \sum_i \sum_j \frac{\left(n_{ij} - \frac{n_{i+}n_{+j}}{N}\right)^2}{n_{i+}n_{+j}} \\
&= \sum_i \sum_j \frac{n_{ij}^2}{n_{i+}n_{+j}} + \sum_i \sum_j \frac{-2n_{ij}n_{i+}n_{+j}}{N \cdot n_{i+}n_{+j}} + \sum_i \sum_j \frac{n_{i+}^2 n_{+j}^2}{N^2 n_{i+}n_{+j}} \\
&= \sum_i \sum_j \frac{n_{ij}^2}{n_{i+}n_{+j}} - \frac{2}{N} \sum_i \sum_j n_{ij} + \frac{1}{N^2} \sum_i \sum_j n_{i+}n_{+j} \\
&= \sum_i \sum_j \frac{n_{ij}^2}{n_{i+}n_{+j}} - 1
\end{aligned}$$

, π_+ is the size of the largest block in the partition π and π_- is the size of the smallest block in the partition π

$$d_2(\pi, \sigma) = \frac{1}{N^2}\left[\sum n_{i+}^2 + \sum n_{+j}^2 - 2\sum\sum n_{ij}^2\right]$$

, where $i+$ is the sum of the tuples whose attribute is n_i, n_{ij} is the size of the intersection of the ith block of π and the jth block of σ, and N is the total number of tuples.

Now look at the right-hand inequality. The right-hand side is:

$$\frac{1}{\pi_-\sigma_-}\left[\|\pi\|^2 + \|\sigma\|^2 - N \cdot d(\pi, \sigma)\right] \leq \frac{1}{\pi_-\sigma_-}\left[\|\pi\|^2 + \|\sigma\|^2 - d(\pi, \sigma)\right]$$

$$= \frac{1}{\pi_- \sigma_-} \left[\sum n_{i+}^2 + \sum n_{+j}^2 - \sum n_{i+}^2 - \sum n_{+j}^2 + 2 \sum \sum n_{ij}^2 \right]$$

$$= \frac{2}{\pi_- \sigma_i} \sum \sum n_{ij}^2$$

$$\frac{1}{\pi_- \sigma_-} \sum \sum n_{ij}^2 \leq$$

$$\sum \sum \frac{n_{ij}^2}{n_{i+} n_{+j}} \leq$$

$$\sum \sum \frac{n_{ij}^2}{n_{i+} n_{+j}} - 1 \leq$$

$$\chi^2 =$$

The left-hand inequality:

$$\sum \sum \frac{n_{ij}^2}{n_{i+} n_{+j}} - 1 \;=\; \chi^2$$

$$\sum \sum \frac{n_{ij}^2}{n_{i+} n_{+j}} - \sum \sum \frac{1}{mn} \;=$$

, where m is the number of blocks in π and n is the number of blocks in σ

$$\sum \sum \left(\frac{n_{ij}^2}{n_{i+} n_{+j}} - \frac{1}{mn} \right) \;=$$

Now examine an individual element of the sum $\frac{n_{ij}^2}{n_{i+} n_{+j}}$:

$$\frac{mn \cdot n_{ij}^2 - n_{i+} n_{+j}}{mn \cdot n_{i+} n_{+j}} = \frac{n_{ij}^2}{n_{i+} n_{+j}} - \frac{1}{mn}$$

$$\frac{mn \cdot n_{ij}^2 - n_{ij}^2}{mn \cdot n_{i+} n_{+j}} \leq \qquad \text{, since } n_{i+} n_{+j} \geq n_{ij}^2$$

$$\frac{(mn-1) n_{ij}^2}{mn \cdot n_{i+} n_{+j}} =$$

$$\frac{mn - 1}{mn \cdot n_{i+} n_{+j}} \leq \qquad \text{, since } n_{ij}^2 \geq 1$$

$$\frac{1}{mn \cdot n_{i+} n_{+j}} \leq \qquad \text{, since } mn > 1$$

Thus,

$$\begin{aligned}
\sum\sum \frac{n_{ij}^2}{n_{i+}n_{+j}} - 1 &\leq \chi^2 \\
\sum\sum \frac{1}{mn \cdot n_{i+}n_{+j}} &\leq \\
\frac{1}{mn}\sum\sum \frac{1}{\pi_+\sigma_+} &\leq \\
\frac{1}{\pi_+\sigma_+} &=
\end{aligned}$$

∎

Thus, as the distance measure increases χ^2 decreases, i.e. the further apart the attributes are, the more independent are the attributes, which is what would expect.

CHAPTER 10

Cluster Algorithms – Coding

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

– Wernher von Braun

Mosher's Law of Software Engineering: Don't worry if it doesn't work right. If everything did, you'd be out of a job.

– Anonymous

10.1 Introduction

Clustering has been implemented with System R and using Java and Oracle. The reason for this is that R lends itself to prototyping while the Java/Oracle combination may be more usable by others. So the discussion will break down into two distinct parts: the R part and the Java part.

10.2 R Part

Implementing the cluster algorithm in R was quite straightforward, especially since there are a number of functions that will help. The functions from existing libraries

that were used were:

agnes (Agglomerative Nesting): computes the agglomerative hierarchical clustering of the dataset [Tea03].

pam (Partitioning Around Medoids): partitions (clusters) the data into **k** clusters "around medoids", a more robust version of K-means [Tea03].

rpart (Recursive Partitioning and Regression Trees: grows a tree by binary recursive partitioning [Tea03].

These were used with the distance function to see how it compared with other techniques for clustering and creating decision trees. The essential idea was to try to find 'representative' attributes which could be used to cluster the data effectively. The distance function was first used to cluster the data and then picked a representative attribute from each cluster of the pruned tree.

Here is the pseudo-code:

Input: `A table` T

Output: `A list of trees at all the heights where clustering occurs, showing the representative attribute and the other attributes for each cluster`

Method:
`if` T `is not discretized, discretize it;`
`find the distance between each attribute, store as a matrix;`
`Use` **agnes** `to compute an agglomerative hierarchical clustering of` T`;`
`use` **rpart** `to create a tree at each height wanted;`

A more detailed explanation of the using **rpart** in the implementation in R is:

- call **domyrpart** which will go through **getrepatts** a given number of times and return a list of the results which **plotrpartL** will use as input to plot the number of clusters versus cross-validation error
  - call **getrepatts** which returns a vector of the attributes' col number for a given number of clusters that are 'representative' of the attributes in the clusters.
  - call **myrpart** which takes the attributes from above converts them into a formula which it feeds to rpart
  - call **rpart** which returns an **rpart** object, essentially a tree
  - call **plotrpartL** which will return a plot of the results.

10.3 R Functions Written for Clustering

System R was used to create the following major functions:

finddist : finddist(a1, a2, exponent)

- *inputs:*

  a1: one attribute, its column number

  a2; the other attribute, its column number

  exponent: the exponent to use in the formula

- *output:* the distance between two attributes, using the distance function mentioned throughout this paper

getdistobj: getdistobj(mat, tar=0, exponent=2)

- *inputs:*

mat: a matrix of data, discretized

tar: the number of the column of the target attribute, the default is 0, i.e. the target attribute will not be left out. If you put in the target's column number, it will be left out in the calculations. You might want to leave the target attribute out of the calculations if you are looking at clustering of attributes.

exponent: the exponent to be used by our distance function, 2 is the default

- *output:* a dist class object, essentially a matrix with distances in lower left triangle, which are the distances between each attribute, using our distance function, needs finddist from disstuff2.

getrepatts: getrepatts(distobj, treeobj, k=NULL, h=NULL)

- *inputs:*

  distobj: a dist object

  treeobj: a tree object

  k: the number of clusters in the 'smaller' tree

  h: the minimum height where the tree will be cut

- *output:* a vector of the attributes' col number for a given number of clusters or height that are 'representative' of the attributes in the clusters, used to try to use fewer attributes to classify correctly

domyrpart: domyrpart(tar, dtab, distobj, treeobj, ..., maxclust=0, height=T)

- *inputs:*

  tar: target attribute

dtab: data - assumed to be nominal

distobj: dist obj of data (excluding target attr)

treeobj: data in tree form, e.g. agnes

maxclust: the max number of clusters if the user inputs nothing the function will find the number of clusters in the original tree and use that

height: boolean, to state whether the tree is being trimmed by number of clusters or pruned at a particular height

- *output: rbiglist* which is list(height, rcnt, rlist, attslist, attsusedlist)

  height: boolean indicating whether the rparts were found using min height or num of clusters

  rcnt: either a count of the num of clusters or the heights for each pruned tree found

  rlist: a list of rparts - the decision trees

  attslist: a list of the attributes 'offered' to rpart

  attsusedlist: a list of the attributes actually used by rpart for each rpart implementation

10.4 Java/Oracle

This implementation is quite different from the R implementation. While the R implementation built the dendrogram, this version viewed the clustering as a datatype and built it that way.[1]

The basic class is HCL which contains:

[1]Much of the inspiration for this code came from twins, FORTRAN code from the R project, and F. Murtagh (f.murtagh@qub.as.uk).

- Data:

  dissmat: dissimilarity matrix of the attributes

  attNames: attribute names, except the class attribute

  className: class attribute name

  heights: the heights where clustering occurs

  tableName: the name of the table in Oracle

- Methods:

  HCL: various constructors. The basic one of which will build the information needed from the table, e.g. the dissmat, the clustering.

  Output: various methods for writing out information, e.g. outputting an arff file, for use in Weka.

There is a wrapper class HCLWrapper which gives the user a reasonable interface to the HCL class. This class offers the user the following choices for massaging the table:

Info: gives a short description of what the program can do.

Go through: take a table and create arff files for each height there is clustering. The file consists of the representative attributes for that height.

Repeat: allows the user to go through a table repeatedly picking heights and seeing the output. It will output on the screen the heights as a range or all of them. Once the user has picked a height, it can output the clustering as a vector of numbers (telling the user which cluster each attribute is in

by number), the representative attributes, the clustering so the user can see which attribute is in each cluster, the average distance within and between the clusters, and will let the user decide whether to output an arff file for the height and whether to leave the table in Oracle with the new column (the column showing the clustering) or not.

Change table: go to a different table and go through it as the user wishes.

At present the Java implementation allows for clustering by the following methods: group, single, complete, or Ward. Since it uses the Lance-Williams formula, any other method that can be written using that formula can be added merely by changing a few places in the code.

The code mentioned in this dissertation may be found at cs.umb.edu/∼rickb.

CHAPTER 11

Conclusion

Vail's Second Axiom: The amount of work to be done increases in proportion to the amount of work already completed.

– Anonymous

11.1 Conclusion

This thesis has introduced the Generalized Barthélemy-Monjardet distance, used it to discretize data, and cluster data. In discetization, it tried to show that this is a reasonable extension of Fayyad's discretization strategy [Fay91] which often gives better results. In clustering, it showed a clustering method which returned good results and also returned dendrograms that gave the user an insight into which attributes clustered to which for further study of these attributes relationships.

11.2 Future Work

Obvious avenues of future research would be:

- to determine if there are any predictors for which is the best value for β when discretizing.

- to determine if there is a way to predict when clustering using the Barthélemy-Monjardet distance would be more or less effective than CSF or the Wrapper method in Weka.

REFERENCES

[Agr97] A. Agresti. *An Introduction to Categorical Data Analysis.* John Wiley, New York, 1997.

[Bar78] J.P. Barthélemy. "Remarques sur les propriétés metriques des ensembles ordonnés." *Math. Sci. hum.*, **61**:39–60, 1978.

[Bel34] E. T. Bell. "Exponential Numbers." *Amercian Mathematical Monthly*, **41**:411 – 419, 1934.

[BFO98] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees.* Chapman and Hall, Boca Raton, 1998.

[BGL00] M.P.S. Brown, W.N. Grundy, D. Lin, N. Cristiani, C. W. Sugnet, T.S. Furey, M. Ares, and D. Haussler. "Knowledge-based analysis of microarray gene expression data by using support vector machines." *PNAS*, **97**:262 – 267, 2000.

[BL95] J.P. Barthélemy and B. Leclerc. "The Median Procedure for Partitions." In *Partitioning Data Sets*, pp. 3–34, Providence, 1995. American Mathematical Society.

[BL97] A. Blum and P. Langley. "Selection of relevant features and examples in machine learning." *Artificial Intelligence*, pp. 245–271, 1997.

[BM98] C. L. Blake and C. J. Merz. *UCI Repository of machine learning databases.* University of California, Irvine, Dept. of Information and Computer Sciences, http://www.ics.uci.edu/∼mlearn/MLRepository.html, 1998.

[BPS05] Richard Butterworth, Gregory Piatetsky-Shapiro, and Dan A. Simovici. "On Feature Selection through Clustering." *ICDM*, 2005.

[BSS04] R. Butterworth, D. Simovici, G Santos, and L Ohno-Machado. "A Greedy Algorithm for Supervised Discretization." *Biomedical Informatics*, **37**:285 –292, Jun 2004.

[Dar70] Z. Daróczy. "Generalized Information Functions." *Information and Control*, **16**:36–51, 1970.

[Dev74] P. A. Devijer. "Entropie quadratique et reconnaisance des formes." In *Computer Oriented Learning Processes, Proceedings of the NATO Advanced Study Institute*, pp. 257–278, Château de Bonas, France, 1974.

[DHS01] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification.* John Wiley and sons, 2nd edition, 2001.

[Fay91] U. M. Fayyad. *On the Induction of Decision Trees for Multiple Concept Learning.* PhD thesis, University of Michigan, 1991.

[GE03] E. Guyon and A. Elisseeff. "An Introduction to Variable and Feature Selection." *J. of Machine Learning Research*, pp. 1157–1182, 2003.

[Gr91] George Grätzer. *General Lattice Theory.* Birkhäuser, 1991.

[GST99] T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, H. Coller, M. Loh, J.R. Downing, M.A. Caligiuri, C.D. Bloomfield, , and E.S. Lander. "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression." *Science*, pp. 531–537, 1999.

[Hal99] Mark A. Hall. *Correlation;based Feature Selection for Machine Selection.* PhD thesis, The University of New Zealand, 1999.

[HC67] J. H. Havrda and F. Charvat. "Quantification Methods of Classification Processes: Concepts of Structural α-Entropy." *Kybernetica*, **3**:30–35, 1967.

[HCB03] B. Hanczar, M. Courtine, A. Benis, C. Hannegar, K. Clement, and J.D. Zucker. "Improving Classification of Microarray Data Using Prototype-based Feature Selection." *SIGKDD Explorations*, pp. 23–28, 2003.

[HW79] J. A. Hartigan and M.A. Wong. "A K-means clustering algorithm." *Applied Statistics*, pp. 100 – 108, 1979.

[JKP94] G. H. John, R. Kohavi, and K. Pfleger. "Irrelevant features and the subset and the subset selection problem." *Proceedings of the Eleventh International Conference on Machine Learning*, pp. 121–129, 1994.

[KJ97] R. Kohavi and G. John. "Wrappers for feature selection." *Artificial Intelligence*, pp. 273–324, 1997.

[KR90] L. Kaufman and P. J. Rousseeuw. *Finding Groups in Data – An Introduction to Cluster Analysis.* Wiley Interscience, New York, 1990.

[KWR01] J. Khan, J.S Wei, M. Ringner, L. H. Saal, M. Ladanyi, F. Westerman, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, and

P.S. Meltzer. "Classification and Diagnostic Prediction of Cancers using Gene Expression Profiling and Artificial Neural Networks." *Nature Medicine*, **7**:673–679, 2001.

[Man91] R. López de Màntaras. "A Distance-Based Attribute Selection Measure for Decision Tree Induction." *Machine Learning*, **6**:81–92, 1991.

[Mon81] B. Monjardet. "Metrics on Parially Ordered Sets – A Survey." *Discrete Mathematics*, **35**:173–184, 1981.

[SB04] D. A. Simovici and R. Butterworth. "A Metric Approach to Supervised Discretization." *Revue des Nouvelles Technologies de l'Information*, **1**:197 – 202, Jan 2004.

[SJ02] D. A. Simovici and S. Jaroszewicz. "An Axiomatization of Partition Entropy." *IEEE Transactions on Information Theory*, **48**:2138–2142, 2002.

[SJ03] D. A. Simovici and S. Jaroszewicz. "Generalized Conditional Entropy and Decision Trees." In *Extraction et Gestion des connaissances - EGC 2003*, pp. 363–380, Paris, 2003. Lavoisier.

[SS04] D. Simovici and N. Singla. "Metric Incremental Clustering of Categorical Data." In *Proceedings of ICDM*, pp. 523–527, 2004.

[SW63] C Shannon and W. Weaver. *The Mathematical Theory of Communication*. University of Illinois Press, 1963.

[TA03] I. Tsamardinos and C. F. Aliferis. "Towards Principled Feature Selection: Relevancy, Filters and Wrappers." *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, Jan 2003.

[Tea03] The R Development Core Team. *The R Environment for Statistical Computing and Graphics*. Number Version 1.8.1. The R Development Core Team, 21 Nov 1999 - 2003.

[WF00] I. H. Witten and E. Frank. *Data Mining – Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann, San Francisco, 2000.

[WM97] D. Randall Wilson and Tony R. Martinez. "Improved Heterogeneous Distance Functions." *Journal of Artificial Intelligence Research 6*, pp. 1 – 34, January 1997.

[ZJ96] D. Zongker and A. Jain. "Algorithms for Feature Selection: An Evaluation." In *Proceedings of the International Conference on Pattern Recognition*, pp. 18–22, 1996.