COMPOSITE BOOLEAN SEPARATORS FOR DATA ANALYSIS WITH APPLICATIONS IN COMPUTED TOMOGRAPHY AND GENE EXPRESSION MICROARRAY DATA

by

IRINA I. LOZINA

A Dissertation submitted to the

Graduate School-New Brunswick

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

Graduate Program in Operations Research

written under the direction of

DR. PETER L. HAMMER and DR. ALEXANDER KOGAN

and approved by

_____

_____

_____

_____

_____

New Brunswick, New Jersey

May 2007

ABSTRACT OF THE DISSERTATION

COMPOSITE BOOLEAN SEPARATORS FOR DATA ANALYSIS WITH APPLICATIONS IN COMPUTED TOMOGRAPHY AND GENE EXPRESSION MICROARRAY DATA

By IRINA I. LOZINA

Dissertation Directors:

DR. PETER L. HAMMER

and

DR. ALEXANDER KOGAN

An important topic in machine-learning / data-mining is that of analyzing binary datasets. A binary dataset consists of a subset of *n*-vectors (observations) with binary components, each of which has an associated binary outcome (the class of the observation). Clearly, the set of *n*-vectors and their outcomes represent a partially defined Boolean function. The central problem of machine-learning / data-mining, the so-called *classification problem*, consists in finding an "extension" of the partially defined Boolean function closely approximating a hidden ("target") function. Various methods have been developed to solve this and related problems, such as identifying misclassified observations, revealing irrelevant and/or redundant variables, etc.

In this thesis, we propose a new approach to analyzing different problems in machine-learning / data-mining. First, we define a simple procedure for generating artificial Boolean variables, called *Composite Boolean Features*, and describe an iterative algorithm for generating Boolean functions which agree with the outcomes in a large proportion of the observations in the dataset. We call these functions *Composite Boolean Separators* (*CBSes* for short). We then use the idea of CBSes in several ways. In particular,

- we demonstrate the usefulness of these concepts by showing how the introduction of CBSes can enhance the accuracy of classification systems;
- we employ CBSes for identifying misclassified observations and examine how deletion of such observations and reversal of their class influence the classification accuracy;
- we apply the new variables to the attribute selection problem, i.e., to the problem of finding "good" (*informative*) subsets of the original attributes, or equivalently, identifying "bad" (irrelevant and/or redundant) attributes in the given datasets.

All the results have been tested on eight publicly available datasets and validated by five well-known machine-learning / data-mining techniques. Also, we applied CBSes, along with other techniques, to the analysis of two real-life medical datasets: computed tomography data and breast cancer gene expression microarray data.

The results presented in this thesis demonstrate that for many real-life datasets, the application of CBSes increases the classification accuracy significantly. CBSes also prove useful in the missclassification and attribute selection problems.

Acknowledgements

I am greatly indebted to Dr. Peter L. Hammer without whom this dissertation would not be possible. I cannot overestimate the importance of his involvement in my graduate study. He provided encouragement, inspiration, excellent teaching, lots of good ideas and a pleasant company. I am sorry to see the thesis finished without him[1].

Special thanks go to Dr. Alexander Kogan for his agreement to advise me in the final stage of this work. His guidance has also been abundantly helpful.

I also would like to thank Professors Endre Boros, Andras Prekopa, David Shanno and Gabriela Alexe for serving on my dissertation committee and for the interesting courses they taught.

Finally, I am grateful to my husband, Vadim Lozin, and to my daughter, Yana, for their love, understanding, endless patience, support, encouragement and for believing in me.

[1] Peter L. Hammer (1936 – 2006) passed away in a tragic car accident on December 27, 2006.

Table of Contents

List of Tables

List of Illustrations

1. INTRODUCTION

An important topic in machine-learning / data-mining is that of analyzing binary datasets. A binary dataset Ω consists of a subset of n-vectors with binary $\{0,1\}$ components, each of which has an associated binary outcome. The n-vectors of Ω are called observations, while those whose outcome is 1 (respectively 0) are called positive (respectively negative) observations. We shall denote the sets of all positive and negative observations in Ω by Ω^+ and Ω^-, respectively. The i-th components of all the vectors in Ω will be viewed as the values of a variable x_i; frequently variables are also called attributes or features.

Clearly, the set of n-vectors in Ω and their outcomes represent a partially defined Boolean function. The central problem of machine-learning / data-mining, the so-called *classification problem*, consists in finding an "extension" of the partially defined Boolean function (i.e., a Boolean function which is defined in every binary n-vector, and which agrees in Ω with the given values) closely approximating a hidden ("target") function.

Since the number of variables present in datasets appearing in real-life problems is usually very large, an important aspect of the classification problem is *attribute selection*, i.e., a set of methods and techniques for identifying and eliminating unnecessary variables included in datasets. There is a rich literature in machine-learning / data-mining dedicated to attribute selection (see [39], [40], [65], and for survey see [15] and [29]). This is an extremely important area of research, since the number of irrelevant and/or

redundant variables is often very large, and their presence does not only slow down the computational aspects of data analysis, but can also introduce inaccuracies and errors.

An exactly opposite approach, which one can term "attribute construction", has been adopted in several publications ([2], [28], [47], [55], [66], [67], [72], [75]), where beside the given variables, additional "artificial" variables have been introduced and added to the given ones, in order to increase the accuracy of classification. For example, the artificial variables proposed in [28] were associated to pairs of given variables using simple arithmetic operations; to a pair of binary variables x, y it was suggested to associate new variables of the form $a+bx+cy$, where a, b and c were real numbers, chosen in such a way that the artificial variable contributed to the separation of positive observations from negative ones. It was shown in the same paper that the introduction of artificial variables can enhance the accuracy of classifications.

In this thesis, we contribute to both problems: attribute construction and attribute selection. In particular, we develop a procedure for creating new variables that represent logical functions of the given variables. We then add the new variables to the original set and study the effect of this addition on the accuracy of classification. We also apply the new variables to the attribute selection problem, i.e., to the problem of finding "good" (*informative*) subsets of the original attributes, or equivalently, identifying "bad" (irrelevant and/or redundant) attributes in the given datasets. Moreover, the artificial variables turn out to be a useful tool in one more aspect of data cleaning, i.e., identifying "bad" observations.

Usually, real-word datasets contain noise which can be introduced in different ways. For example, errors can be made at the time of sampling, i.e., incorrect data was collected for some observations. We refer to the problem of identifying such observations as the *attribute noise problem*. Another example deals with the situation when an operator, who creates a dataset electronically, inputs a wrong class to some observations. Such errors are called *classification noise.* Wrongly classified observations may appear in a different way, for instance, when a medical doctor makes an incorrect diagnosis. We refer to the problem of identifying misclassified observations as the *classification noise problem*, or simply *misclassification problem*. Identifying observations containing noise is very important, since their presence may result in incorrect classification models. Different methods for identifying suspicious observations were discussed in many papers ([13], [21], [22], [24], [69], [71], [93], [94], [97], [100], [101]). In our study, we concentrate on the classification noise problem and present two new techniques for finding subsets of suspicious observations. One of them is a new approach for data cleaning: it uses synthetic variables to eliminate noise. The second one is a development of the approach proposed by Brodley and Friedl in [21], [22]. This method uses *Simulated Error Rate* (*SER*) for identifying suspicious sets. We examine how deletion of suspicious observations and reversal of their class influence the classification accuracy.

All our results are experimental, which is in accordance with the following observation by Thomas G. Dietterich [31]:

> **"**Fundamental research in machine-learning is inherently empirical, because the performance of machine-learning algorithms is determined by how well their underlying assumptions match the structure of the world. Hence, no amount of

mathematical analysis can determine whether a machine-learning algorithm will work well. Experimental studies are required".

In our computational experiments we use five well-known machine-learning / data-mining methods. A short description of each of these methods is presented below.

1.1. Classification Methods

- **Support Vector Machines** ([33], [81], [82], [83])

  This method is founded on Vapnik's Statistical Learning Theory [92]. A support vector classifier has the form

$$y = \begin{cases} 1 & when \quad \sum_j a_j y_j K(x_j, x) + b \geq 0 \\ 0 & otherwise \end{cases}$$

  where the coefficients α_j and b are learned parameters (they are learned by solving a convex optimization problem) and the function $K(x_j, x)$ is a kernel function that in some sense measures the similarity between the test observation x and the training observation x_j.

- **Simple Logistic Regression** ([26], [52], [62], [89])

  This is a classifier for building linear logistic regression models. Logistic regression allows one to predict a discrete outcome. What we want to predict from knowledge of relevant independent variables is not a precise numerical value of a dependent variable, but rather the probability (p) that it is 1 rather than 0. Since probabilities can only take values between 0 and 1, a logistic transformation of p is made, which is defined as:

$$\text{logit}(p)=\log(p/(1-p)) .$$

Logistic regression involves fitting to the data an equation of the form:

$$\text{logit}(p)= a + b_1x_1 + b_2x_2 + b_3x_3 + ...$$

Just like linear regression, logistic regression estimates for each regressor x_i a coefficient b_i which measures the regressor's independent contribution to variations in the dependent variable and also estimates the constant a of the equation.

- **Multilayer Perceptron** ([68], [70], [78], [79], [80])

  Multilayer Perseptron (MP) is a network of simple *neurons* called *perceptrons*. The basic concept of a single perceptron was introduced by Rosenblatt in 1958. MP is composed of more than one layer of neurons (artificial neural network), with some or all of the outputs of each layer connected to one or more of the inputs of another layer. The first layer is called the input layer, the last one is the output layer, and in between there may be one or more hidden layers. The principle of the network is that when data from an input pattern is presented at the input layer the network nodes perform calculations in the successive layers until an output value is computed at each of the output nodes. This output signal should indicate the appropriate class for the input data.

- **Decision trees (C4.5)** ([73], [74])

  J. Ross Quinlan’s algorithm C4.5 builds decision trees top-down and prunes them. The procedure starts from the root node and greedily chooses a split of the

data that maximizes some objective function. After choosing a split, the sub-samples are then mapped to the two children nodes. This procedure is then recursively applied to the children, and the tree is growing until each subset in the partition contains cases of a single class, or until no test offers any improvement. The result is often a very complex tree that "overfits the data". The tree is then used as a starting point for a bottom up search, performing a pruning of the tree. This eliminates nodes that are redundant or are unable to "pay for themselves" in terms of the objective function.

- **Logical Analysis of Data (LAD)** ([18], [27], [41], [42])

  The idea of LAD for analysis of binary data was proposed by P.L.Hammer in the middle of 1980s. Later it was developed for analysis of data with numerical values of attributes. LAD is a combinatorics, optimization, and logic based methodology for the analysis of data. The basic concepts used in LAD are described below.

  Cut points and binarization.

  One of the underlying principles of LAD is to disregard the exact values of a variable, specifying for each observation only whether the corresponding value of this variable is sufficiently 'large' or 'small'. The binarization procedure proposed in [17] consists in associating to each numerical variable x one or more *cutpoints* c', c'', …, and then associating to each of these cutpoints a binary variable x', x'', …, defined by

$$x' = \begin{cases} 1 & if \quad x > c' \\ 0 & otherwise \end{cases}, \qquad x'' = \begin{cases} 1 & if \quad x > c'' \\ 0 & otherwise \end{cases}, \quad \ldots\ldots\ldots$$

The binarization process is correct if and only if the binary (0, 1) vector representing the image of any positive observation is different from the binary image of any negative observation. It has been shown in [17] that the minimization of the number of binary variables allowing the correct binarization of a given dataset can be accomplished by solving a set-covering problem.

Logical patterns

The central concept of LAD is a pattern. A 'conjunction' is a set of conditions that require that the binary variables take specific (0 or 1) values. A conjunction is called a positive (or negative) pattern if its conditions are satisfied simultaneously by 'sufficiently many' of the positive (or negative) cases, and by 'sufficiently few' of the negative (or positive) cases. If an observation satisfies all the conditions describing a pattern, then we say that the observation is *covered* by this pattern. Three of the most important characteristics of a pattern are its *degree*, its *prevalence* and *homogeneity*.

- The *degree* of a pattern is the number of its defining conditions.
- The *prevalence* of a positive (or negative) pattern is the proportion of positive (or negative) cases covered by it.
- The *homogeneity* of a positive (respectively negative) pattern is the proportion of positive (respectively negative) cases among all the cases covered by the

pattern. Patterns which cover only positive or only negative cases (i.e., have the homogeneity 100%) are called *pure patterns*.

Pandect and Theory

The *pandect* (i.e., the collection of all of the positive and negative patterns corresponding to a dataset) is an important concept of LAD which is used in the construction of diagnostic and prognostic systems, analysis of the importance and role of variables, and identification of new classes of observations, among other factors. In view of the enormous number of patterns found in a typical dataset, the construction of the entire pandect is not realistic. The set of all positive (or negative) patterns of degree at most d^+ (or d^-) and prevalence at least p^+ (or p^-) is called the (d^+, p^+) *positive pandect* (or the (d^-, p^-) *negative pandect*). Clearly, the pandect is not a minimal system because it may contain many redundant patterns, without which the system can still remain accurate. LAD uses the set-covering formulation to find a subset of patterns which cover all the observations. This subset is called a *theory* (or *model*). *Models* provide classification of observations in the dataset as well as of new observations. The way in which the model provides the classification of a new observation is the following.

- ✓ If an observation is covered by positive (negative) patterns, but is not covered by any one of the negative (positive) patterns, then the observation is classified as positive (negative).

- ✓ If an observation is not covered by any positive or negative pattern, then it remains "unclassified".
- ✓ If an observation is covered by some positive and also some negative patterns in the model, then a weighting process is applied to decide on the appropriate classification.

Pattern space

Pattern-based representation of the observations is constructed by associating to each observation and to each pattern in the pandect an indicator variable that shows whether the observation satisfies (indicator = 1) or does not satisfy (indicator = 0) the conditions that define that pattern. In this way, each observation is characterized by a sequence of 0-1 values of the indicator variables associated with the positive and negative patterns in the pandect.

1.2. Structure of Thesis

In Chapter 2, we define a simple procedure for generating artificial Boolean variables, called *Composite Boolean Features*, and describe an iterative algorithm for generating Boolean functions which agree with the outcomes in a large proportion of the observations in the dataset. We call these functions *Composite Boolean Separators* (*CBSes* for short).

In Chapter 3, we use the idea of composite Boolean separators to study various problems

of machine-learning / data-mining. In particular, in Section 3.1 we demonstrate the usefulness of these concepts by showing on a number of publicly available datasets how the introduction of CBSes can enhance the accuracy of classifications. Sections 3.2 and 3.3 are devoted to data cleaning problems. More specifically, in Section 3.2 we analyze the problem of identifying misclassified observations and develop two new approaches to solve it. One of them is a *CBS* based method, while the other one is a modification of the idea of *Simulated Error Rate* (*SER*) introduced by Brodley and Friedl [21], [22]. Both approaches demonstrate robustness in eliminating class noise. In Section 3.3, composite Boolean separators are applied to the attribute selection problem, i.e., the problem of identifying informative subsets of variables. All the results presented in Chapter 3 are tested on eight datasets available on the Web in the *Repository of the University of California at Irvine*.

In Chapter 4 we present case-study results based on two real-life medical datasets: computed tomography data and breast cancer gene expression microarray data.

2. GENERATION OF COMPOSITE BOOLEAN SEPARATORS

The results presented in this chapter refer to binary datasets only. In order to apply our techniques to a dataset which is generally non-binary, we have first to "binarize" the data, i.e., to replace each variable taking numerical values by one or more binary variables, following the procedure in [17]. After converting the dataset to an equivalent binary form, we view the new data as representing a partially defined Boolean function (pdBf). The classification problem requires the determination of an "extension" of this pdBf, i.e., of a Boolean function which agrees with the values of the pdBf in the points of the dataset. To determine such an extension, we develop a heuristic method for constructing new Boolean variables, called *Composite Boolean Feature*s (*CBF*s). After a number of iterations, this method produces a set of *Composite Boolean Separators* (*CBSe*s), each of which is a Boolean function agreeing with the given outcomes in a large proportion of the observations in the dataset.

2.1. Concept of Composite Boolean Separators

In this chapter, we describe an approach for creating a set of artificial Boolean variables (to be called *Composite Boolean Features*). For this purpose, to *every* pair of binary (0,1) variables x and y we associate 12 new binary variables $f_1(x,y)$, …, $f_{12}(x,y)$, where $\{f_i(x,y), i = 1,\ldots,12\}$ is the set of *all* Boolean functions depending on two variables, except for two constant functions (0 and 1) and two functions representing the original variables (x and y). At this point the newly created binary variables are added to the dataset, and

the unknown Boolean function which determines the positive or negative nature of each observation (frequently called the "hidden function" or, the "target function") is considered as depending both on the original and on the added variables. In order to keep at a reasonable level the size of the problem obtained by introducing the composite Boolean features as additional variables in the dataset, a filtering mechanism is used to retain only those CBFs which have a sufficiently high classification power.

The basic idea of using Boolean functions for producing artificial variables has already been considered in the literature (see e.g. [16], [47], [66]). Among the methods based on an iterative generation of artificial Boolean variables we mention GALA [55], CITRE [67], FRINGE [72], and LFC [75]. In several studies (IB3-CI [2], GALA [55], CITRE [67], FRINGE [72], LFC [75]) special attention is given to the use of minimal sets of logical operators (e.g. negation and conjunction) to express existing Boolean relations between data attributes. In some algorithms (e.g. LFC and FRINGE) feature construction is accomplished parallelly with the construction of decision tree classifiers. In most of these methods the new features are constructed based on previously generated hypotheses. Another common feature of most of these methods is the confinement of the construction process to a restricted set of Boolean expressions.

The method proposed in this thesis differs from the above mentioned ones in two essential ways. On the one hand, the composite Boolean features generated at each step of the proposed iterative process are associated to *every* pair of existing variables, i.e., no pre-selection is made to identify the "most promising" pairs. On the other hand, the

Boolean operations used for generating new CBFs are not restricted in any way, i.e., the outputs of ***all*** the Boolean functions of two variables (with the obvious exception of constant functions) are evaluated as candidates for new composite Boolean features.

2.2. Composite Boolean Features as Artificial Variables

In order to present in detail the procedure of generating composite Boolean features we shall define the *negation* $\bar{x}$ of a binary $\{0,1\}$ variable x as $1 - x$, and define for any two binary variables x_i and x_j, their *disjunction* $x_i \vee x_j = x_i + x_j - x_i\, x_j$, their *conjunction* $x_i \,\&\, x_j$, defined as their product $x_i\, x_j$ (and denoted simply as $x_i\, x_j$), and their *sum modulo 2* as $x_i \oplus x_j = x_i + x_j - 2\, x_i\, x_j$. Note that treating the 0,1 values of Boolean variables as the numbers 0,1 (i.e., not as symbols) allows the definition of arithmetic operations with them, and does not lead to any confusion.

Given a Boolean function y depending on a subset of the Boolean variables $x_1, x_2, \ldots, x_n$ in the dataset, the *classification power of* y, $CP(y)$, is defined in the following way: if $\pi(y)$ denotes the number of positive observations for which the value of y is 1, and $\upsilon(y)$ denotes the number of negative observations for which the value of y is 0, then

$$CP(y) = \frac{1}{2}\left(\frac{\pi(y)}{|\Omega^+|} + \frac{\upsilon(y)}{|\Omega^-|}\right).$$

In order to construct the composite Boolean features associated to a dataset we shall associate to every pair of Boolean variables x_i, x_j all the Boolean functions $y_k(x_i, x_j)$

depending on them. In total, there are 16 Boolean functions of two variables:

$$1,\ 0,\ x_i,\ \bar{x}_i,\ x_j,\ \bar{x}_j,\ x_i \vee x_j,\ x_i x_j,\ x_i \vee \bar{x}_j,\ x_i \bar{x}_j,\ \bar{x}_i \vee x_j,\ \bar{x}_i x_j,\ \bar{x}_i \vee \bar{x}_j,\ \bar{x}_i\ \bar{x}_j,\ x_i \oplus x_j,\ \overline{x_i \oplus x_j}$$

As we mentioned before, we exclude from our consideration the two constant functions 0 and 1. Also, we do not generate the two functions x_i and x_j as they are present in the dataset. Therefore, to every pair of Boolean variables we shall associate 12 Boolean functions. In order to reduce the number of Boolean functions generated in this way, we shall calculate the *CP* of each $y_k(x_i, x_j)$ and retain only those functions whose *CP* exceeds a certain threshold. In this thesis, we take as the threshold the maximum of $CP(x_i)$ for all $i = 1,\ldots, n$; clearly, choosing a higher (lower) threshold would lead to the retention of a smaller (larger) set of CBFs.

Before we proceed to a formal description of the procedure of generating composite Boolean features, let us consider a simple example illustrating the main steps of the procedure.

Example. Let us consider a dataset containing three negative observations (A,B,C) (the "class" of these is labeled 0) and three positive observations (D, E, F) (the "class" of these is labeled 1), described in terms of four binary variables (x_1, x_2, x_3, x_4):

Obs.	x_1	x_2	x_3	x_4	class
A	0	1	0	0	0
B	1	1	1	0	0
C	0	0	0	1	0
D	1	0	1	0	1
E	1	0	0	0	1
F	0	0	1	1	1

We shall examine now the CBFs depending on the $\binom{4}{2} = 6$ possible pairs of original variables. As mentioned above, for each pair of variables we shall list 12 Boolean functions depending on these two variables. For example, for the pair x_1, x_2 we shall construct the following functions:

Obs.	$\bar{x}_1$	$x_1 x_2$	$x_1 \vee x_2$	$x_1 \bar{x}_2$	$x_1 \vee \bar{x}_2$	$x_1 \oplus x_2$
A	1	0	1	0	0	1
B	0	1	1	0	1	0
C	1	0	0	0	1	0
D	0	0	1	1	1	1
E	0	0	1	1	1	1
F	1	0	0	0	1	0
CP	2/6	2/6	1/2	5/6	4/6	4/6

Obs.	$\bar{x}_2$	$\bar{x}_1 \vee \bar{x}_2$	$\bar{x}_1 \bar{x}_2$	$\bar{x}_1 \vee x_2$	$\bar{x}_1 x_2$	$\overline{x_1 \oplus x_2}$
A	0	1	0	1	1	0
B	0	0	0	1	0	1
C	1	1	1	1	0	1
D	1	1	0	0	0	0
E	1	1	0	0	0	0
F	1	1	1	1	0	1
CP	5/6	4/6	1/2	1/6	2/6	2/6

In the line called *CP* we indicate the classification power of each of the 12 functions above. For example, the *CP* of the function $x_1 x_2$ is 2/6 (since this function agrees with the

outcome in the observations A and C), and the *CP* of its compliment is 4/6 (since the complement agrees with the outcome in the observations B, D, E and F). Similar tables can be constructed for all the other pairs of variables.

Since the largest value of *CP* corresponding to the variables $x_1,\ldots, x_4$ is 4/6 (achieved on x_1 and x_3), we shall retain only those CBFs which have a *CP* of 5/6 or higher; the retained columns are the following:

$$\bar{x}_2,\ x_1\bar{x}_2,\ \ x_1 \vee x_3,\ x_1 \oplus x_3,\ \ \bar{x}_2 x_3,\ \ \bar{x}_2\bar{x}_4,\ \ \bar{x}_2 \vee x_4,\ \ \overline{x_2 \oplus x_4}\ .$$

Obs.	x_1	x_2	x_3	x_4	$\bar{x}_2$	$x_1\bar{x}_2$	$x_1 \vee x_3$	$x_1 \oplus x_3$	$\bar{x}_2 x_3$	$\bar{x}_2\bar{x}_4$	$\bar{x}_2 \vee x_4$	$\overline{x_2 \oplus x_4}$
A	0	1	0	0	0	0	0	0	0	0	0	0
B	1	1	1	0	0	0	1	0	0	0	0	0
C	0	0	0	1	1	0	0	0	0	0	1	0
D	1	0	1	0	1	1	1	0	1	1	1	1
E	1	0	0	0	1	1	1	1	0	1	1	1
F	0	0	1	1	1	0	1	1	1	0	1	0
CP	4/6	1/6	4/6	1/2	5/6	5/6	5/6	5/6	5/6	5/6	5/6	5/6

It can be seen that the CBF $\bar{x}_2 \vee x_4$ takes the same values as $\bar{x}_2$ in each of the 6 observations. Therefore this feature can be eliminated from the table. Similarly, both $\bar{x}_2\bar{x}_4$ and $\overline{x_2 \oplus x_4}$ take the same values as $x_1\bar{x}_2$, and therefore it is enough to retain one (say, $x_1\bar{x}_2$) of these three composite Boolean features. The set of original variables and retained CBFs (to be denoted by x_5, x_6, x_7, x_8 and x_9) becomes

Obs.	x_1	x_2	x_3	x_4	$x_5 = \bar{x}_2$	$x_6 = x_1 \bar{x}_2$	$x_7 = x_1 \vee x_3$	$x_8 = x_1 \oplus x_3$	$x_9 = \bar{x}_2 x_3$	class
A	0	1	0	0	0	0	0	0	0	0
B	1	1	1	0	0	0	1	0	0	0
C	0	0	0	1	1	0	0	0	0	0
D	1	0	1	0	1	1	1	0	1	1
E	1	0	0	0	1	1	1	1	0	1
F	0	0	1	1	1	0	1	1	1	1
CP	4/6	1/6	4/6	1/2	5/6	5/6	5/6	5/6	5/6	

Now let us summarize the above discussion in the following procedure for generating composite Boolean features. We view the computation of the classification power of a variable x as a single call of a subroutine *CP(x)*.

Algorithm *CBF*

Input: a pdBf $F(x_1,x_2,\ldots,x_n)$

Output: a set B of composite Boolean features

M:=max{ $CP(x_1),\ldots, CP(x_n)$}

p:=n

B:=$\varnothing$

For each i=1,…,n-1,

For each j=i+1,…,n,

For each k=1,…,12,

Compute Boolean function $f_k(x_i,x_j)$

If $CP(f_k(x_i,x_j)) > M$ and $f_k(x_i,x_j) \neq x_l$, for each l=1,…,p,

then p:=p+1, x_p:= $f_k(x_i,x_j)$, B:=$B \cup \{ x_p \}$

Return B

As we mentioned before, the choice of the threshold *M* proposed in the algorithm is not the only possible way to define it; choosing a higher (lower) threshold would lead to the retention of a smaller (larger) set of composite Boolean features.

2.3. Iterative Procedure for Finding Composite Boolean Separators (CBSes)

The CBFs identified in the process described in Section 2.2 can be regarded as synthetic variables associated to the dataset. As such, they can be simply added to the original data, and the process described in Section 2.2 can now be repeated on the augmented dataset. Moreover, the resulting CBFs can again be added to the new dataset, and the process can be repeated again. If in a certain step *Algorithm CBF* produces no new composite Boolean features, we terminate the process and call the CBFs found in the previous step the *Composite Boolean Separators* (*CBSe*s) of the original dataset.

It is important to note that it is not necessarily true that there exists a CBS whose values coincide with the correct classification of all the observations in the dataset. Our experience shows however that in every example we have studied, several separators were found which took the same values as the outcome of "almost all" observations.

***Example* (*continued*)**. Let us repeat now the procedure for generating CBFs, with x_1, ..., x_9 playing the role of original variables. Applying *Algorithm CBF* to the extended table, we find the four new composite Boolean features, x_5x_7, $x_6 \vee x_8$, $x_6 \vee x_9$, $x_8 \vee x_9$ having *CP* values exceeding 5/6.

Obs.	$f_1 = x_5 x_7$	$f_2 = x_6 \vee x_8$	$f_3 = x_6 \vee x_9$	$f_4 = x_8 \vee x_9$	class
A	0	0	0	0	0
B	0	0	0	0	0
C	0	0	0	0	0
D	1	1	1	1	1
E	1	1	1	1	1
F	1	1	1	1	1
CP	1	1	1	1	

In conclusion, we have found four functions (f_1, f_2, f_3 and f_4) which take exactly the same values as the class; clearly these functions are CBSes. Substituting in the expressions of these separators, the expressions of $x_5, \ldots, x_9$ as functions of the original variables $x_1, \ldots, x_4$, we find that:

$$f_1 = \bar{x}_2 (x_1 \vee x_3),$$

$$f_2 = (x_1\ \bar{x}_2) \vee (x_1 \oplus x_3),$$

$$f_3 = (x_1\ \bar{x}_2) \vee (\bar{x}_2 x_3),$$

$$f_4 = (x_1 \oplus x_3) \vee (\bar{x}_2 x_3).$$

We conclude this section with a formal description of the procedure for generating composite Boolean separators. This procedure uses *Algorithm CBF* as a subroutine. For the conceptual clarity, we purposely omit some implementation details that are used to improve its efficiency.

Algorithm *CBS* (*Composite Boolean Separators*)

Input: a pdBf *F*

Output: a set *B* of *CBSe*s

While *CBF* (*F*) ≠ ∅

Do *B*:= *CBF* (*F*), Augment *F* by adding to it composite Boolean features from *B*

Return *B*

3. CBS BASED DATA ANALYSIS METHODS

In this chapter, we apply composite Boolean separators to three major problems of machine-learning / data-mining: classification, misclassification and attribute selection. In our experiments, we use several datasets available on the web in the *Repository of the University of California at Irvine* [53] and several frequently used classification methods.

a) Datasets. The datasets examined in this study along with their main characteristics are described in Table 1. The list of cutpoints and binarized variables for these datasets are presented in Table A, …, Table H in the Appendix. Each table provides the definition of the binary variables used in this study. For example, the first line of Table A indicates that

$$a_1 = \begin{cases} 1 & if \quad mcv > 87 \\ 0 & otherwise \end{cases} .$$

Note that we have eliminated from the study the observations which include missing data, and in the case of the **bcw** dataset which contains many repetitions of some of the observations, we have retained only one copy of each observation.

b) *Classification methods***.** In the computational experiments aimed at evaluating the usefulness of composite Boolean separators, we used the *Logical Analysis of Data* (*LAD*) methodology (see [27], [41], or the surveys [18] or [42]), and four of the most frequently

applied machine-learning / data-mining procedures:

- support vector machines (*SMO*),
- artificial neural networks (*MP*),
- linear logistic regression (*SL*),
- decision trees (*C4.5*).

The software used for *LAD* was Datascope [54], while for the other four methods we used the WEKA package [96]. For all these methods we used the default values of the control parameters, as given in the WEKA package.

Table 1

BENCHMARK DATASETS

Name of dataset	Abbreviation	Number of observations		Number of attributes	
		Positive	Negative	Given	Binarized
BUPA liver-disorders	**bld**	200	145	6	29
German credit	**ger**	700	300	24	57
Pima Indians Diabetes	**pid**	130	262	8	23
Cleveland heart disease	**hea**	137	160	13	17
Australian credit	**aus**	307	383	14	45
Ionosphere	**ion**	225	126	33	71
Wisconsin breast cancer	**bcw**	236	213	9	20
Congressional voting records	**vot**	124	108	16	16

3.1. Classification

Classification is an important problem in such fields as artificial intelligence, data mining, machine learning, and so on. In terminology of Boolean functions this problem consists in finding an extension of a given partially defined Boolean function. Such an extension is also called a classifier or a model. In order to establish the reliability of classifiers they have to be validated. For this purpose, we apply a cross-validation technique called *k-folding*.

K-folding involves a random partitioning of the dataset into k (e.g. 2, 5, or 10) approximately equal-size subsets, using k -1 of these subsets as the training set and the remaining one as the test set, then repeating this experiment k times, while using in each experiment another one of the k subsets as the test set, and calculating the average accuracy on the test sets over the k folds.

In Table 2 we report the average classification accuracies obtained by applying the five machine-learning / data-mining methods to the eight datasets in Table 1, using the original variables. The averages refer to the results of twenty 10-folding cross-validation experiments, each of them based on a different random 10-partitioning.

Table 2

CLASSIFICATION ACCURACIES FOR BENCHMARK DATASETS

	SMO	MP	SL	C4.5	LAD	Average
bld	50.03%	67.63%	66.24%	63.65%	69.29%	63.37%
ger	69.39%	65.50%	68.56%	66.18%	72.21%	68.37%
pid	72.31%	73.81%	72.07%	76.13%	74.60%	73.78%
hea	83.05%	78.70%	82.48%	77.16%	82.35%	80.75%
aus	86.47%	82.98%	86.66%	84.93%	85.57%	85.32%
ion	91.10%	88.78%	85.08%	88.05%	91.58%	88.92%
bcw	95.28%	94.39%	94.86%	92.78%	94.44%	94.35%
vot	97.05%	94.42%	96.49%	96.61%	97.14%	96.34%

3.1.1. Composite Boolean Separators for Benchmark Datasets

Applying the iterative algorithm described in Section 2.3 for the generation of composite Boolean separators to the eight datasets described in Table 1 produces new variables, which – at least in some of the cases – have extremely simple expressions in terms of the original binary variables. The simplest example is that of the dataset **vot**

vot: $\bar{a}_4$

which is produced in the very first iteration, and whose *CP* (in percentages) is $\frac{1}{2}\left(\frac{118}{124}+\frac{107}{108}\right)*100 = 97.12\%$. It is interesting to note that this separator *depends only on 1 of the 16 original variables*, but provides the correct classification of 225 of the 232 observations in the original dataset.

Following the same procedure of step-by-step substitutions, we can generate CBSes for

other datasets. For each of the sets we present one of the separators with highest *CP*:

$$\textbf{brw}: \; a_5 \vee a_8 \vee (a_7\, a_{14}) \vee (a_{14}\, a_{15}) \vee (a_9\, a_{15})$$

bld: $(\overline{a}_5\ \overline{a}_{12}\ \overline{a}_{29}) \vee (\overline{a}_5\, a_{18}\ \overline{a}_{29}) \vee (\overline{a}_9\ \overline{a}_{12}\ \overline{a}_{29}) \vee (\overline{a}_{12}\, a_{23}\ \overline{a}_{29}) \vee (\overline{a}_9\, a_{18}\ a_{20}\ \overline{a}_{29}) \vee (a_{18}\ a_{20}\ a_{23}\ \overline{a}_{29}) \vee (\overline{a}_1\ \overline{a}_{13}\ \overline{a}_{16}\ \overline{a}_{26}\ \overline{a}_{29}) \vee (\overline{a}_{13}\ \overline{a}_{16}\, a_{24}\ \overline{a}_{26}\ \overline{a}_{29}) \vee (\overline{a}_1\ \overline{a}_{16}\, a_{25}\, \overline{a}_{26}\ \overline{a}_{29}) \vee (\overline{a}_{16}\, a_{24}\, a_{25}\, \overline{a}_{26}\ \overline{a}_{29}) \vee (\overline{a}_9\ \overline{a}_{11}\ \overline{a}_{16}\ \overline{a}_{26}\ \overline{a}_{29}) \vee (\overline{a}_{11}\ \overline{a}_{16}\, a_{19}\ \overline{a}_{26}\ \overline{a}_{29}) \vee (\overline{a}_9\ \overline{a}_{16}\, a_{23}\ \overline{a}_{26}\ \overline{a}_{29}) \vee (\overline{a}_{16}\, a_{19}\ a_{23}\ \overline{a}_{26}\ \overline{a}_{29})$.

The *CPs* of these two separators are 96.12% and 74.16% respectively. Let us note that since the expression of the separator found for the dataset **bld** is rather complex, it may make sense to use instead of it one of the CBFs found at the previous step of the iterative process

$$\textbf{bld}: (\overline{a}_{11}\, a_{22}) \vee (a_{22}\, a_{25}) \vee (\overline{a}_9\ \overline{a}_{12}) \vee (\overline{a}_{12}\, a_{23}) \vee (\overline{a}_9\, a_{19}) \vee (a_{19}\, a_{23})$$

which has a much simpler expression and has a reasonably high *CP* of 71.37%.

A composite Boolean separator with the highest *CP* generated for the dataset **aus** is

aus: $(a_7 a_{27} a_{28}\ a_{35}) \vee (a_7\, a_{27}\ a_{28}\, \overline{a}_{40}) \vee (a_7\, a_{12}\ a_{27}\, \overline{a}_{40}) \vee (a_7\, a_{14}\, a_{27}\, \overline{a}_{40}) \vee (a_7\, a_{12}\ a_{13} a_{27} a_{35}) \vee (a_7\, a_{13}\ a_{14}\, a_{27}\ a_{35}) \vee (a_{12}\, a_{18}\, a_{27}\ a_{35}\ \overline{a}_{40}) \vee (a_{12}\, a_{14}\, a_{27}\ a_{35}\ \overline{a}_{40}) \vee (a_{12}\, a_{18}\ a_{27}\, a_{28}\ a_{35}) \vee (a_{12}\, a_{14}\ a_{27}\, a_{28}\ a_{35}) \vee (a_{12}\, a_{13}\ a_{14}\, a_{27}\ a_{35}) \vee (a_{13} a_{18}\ a_{27}\, a_{28}\ a_{35}) \vee (a_{13} a_{14}\ a_{27}\, a_{28}\ a_{35})$,

having a *CP* of 88.56%. It should be remarked that – similarly to the case of **bld** – we have found a much simpler CBF for **aus:**

$$\textbf{aus:} \; a_{27}(a_{22} \vee a_{35})\ (a_{33} \vee a_{12} \vee a_{13}),$$

having a *CP* of 87.05%.

Examples of CBSes with highest *CP* for the remaining four datasets are:

$$\textbf{ion :} (a_1\ a_2\ a_6\ a_{11}\ a_{34}\ \overline{a}_5\ a_{48}) \vee (a_4\ a_8\ a_{26})$$

$$\textbf{pid:} (a_4\ a_{22}) \vee (a_3\ a_{14}\ a_{22}) \vee (a_{14}\ a_{18}\ a_{22}) \vee (a_3\ a_5) \vee (a_3\ a_{15}\ a_{22}) \vee (a_4\ a_5) \vee (a_5\ a_9\ a_{23}) \vee (a_9\ a_{15}\ a_{22}\ a_{23})$$

$$\textbf{hea:} (a_4\ a_{12}\ a_{16}) \vee (a_4\ a_5\ a_7\ a_{12}) \vee (a_4\ a_{17}) \vee (a_5\ a_{12}\ a_{17}) \vee (a_5\ a_{16}a_{17}) \vee (a_4\ a_5\ a_{16}) \vee (a_5\ a_7\ \overline{a}_{11}\ a_{12}) \vee (a_4\ \overline{a}_{11}\ a_{16}) \vee (a_5\ a_7\ \overline{a}_{11}\ a_{16})$$

$$\textbf{ger}: (a_3\ a_{20}) \vee (\overline{a}_4\ a_{20}) \vee (a_1\ a_{20}\ a_{22}) \vee (\overline{a}_4\ a_{12}) \vee (a_3\ \overline{a}_{51}) \vee (a_3 a_{12}) \vee (\overline{a}_4\ \overline{a}_{51}) \vee (a_{22}\ a_{48}) \vee a_2 \vee (a_{19}\ \overline{a}_{33}).$$

The *CP*s of these four separators are respectively 93.24%, 82.56%, 86.29%, 74.10%.

The construction of the above listed composite Boolean separators required respectively 1, 4, 7, 8, 5, 5, 6, 10 iterations for the eight datasets considered, i.e., an average of 5.75 iterative steps.

3.1.2. Classification with Composite Boolean Separators

It has been seen before that the values of the different separators coincide among themselves in a (usually) very high proportion of the observations given in a dataset. Moreover, the values of the CBSes are very frequently equal to 1 (respectively to 0) in the positive (respectively negative) observations in the dataset, a property which makes the CBSes a promising tool for classification.

To evaluate the quality of the constructed separators and show their usefulness for classification purposes we perform experiments in two different ways. First, we interpret composite Boolean separators as artificial variables. Second, we view each separator as a classification system.

Composite Boolean Separators as Artificial Variables

In the computational experiments aimed at comparing the results of various classification systems we had always to clarify the extent of the collection of CBSes to be used. Since the number of separators can be large and addition of all of them can introduce extra noise, we have retained in the experiments only the set of *best* CBSes defined as those separators whose *CP*s are within 1% of the highest *CP* of all the CBSes constructed. In some experiments we used only one CBS with the highest *CP*.

In Table 3 reported below we present the results of applying the five classification methods to the eight datasets described in Table 1, using

- the original variables;
- the original variables along with the *best* CBSes found;
- the original variables together with one separator with the highest *CP*;
- the *best* CBSes only.

The results in the table represent averages obtained in twenty 10-folding experiments using five different classification methods (i.e., every entry in the table represents the average accuracy found in 1,000 experiments).

Table 3

AVERAGE CLASSIFICATION ACCURACY ON DATASETS WITH (AND WITHOUT) ORIGINAL VARIABLES AND CERTAIN BEST CBSES

Dataset	Average accuracy of 5 classification methods (SMO, MP, SL, C4.5, LAD) using			
	Original variables	Original variables and best CBSes	Original variables and one CBS with highest CP	Best CBSes
bld	63.37%	73.20%	73.04%	73.42%
ger	68.37%	69.27%	69.03%	73.46%
pid	73.78%	79.47%	79.62%	81.93%
hea	80.75%	83.86%	84.29%	84.80%
aus	85.32%	86.74%	86.75%	88.32%
ion	88.92%	91.90%	91.70%	93.02%
bcw	94.35%	95.42%	95.49%	95.55%
vot	96.34%	96.26%	96.31%	96.79%

It is interesting to note that for the datasets considered,

- the use of the original variables jointly either with all the *best* CBSes, or just with one separator with the highest *CP*, gives higher average accuracy than the use of only the original variables (except for **vot**);
- the use of the *best* CBSes without original variables gives higher average accuracy than the use of the original variables jointly either with all the *best* separators, or just with one separator with the highest *CP*.

The results in Table 3 show high quality of CBSes as artificial variables.

In Table 4 we compare the average accuracies of various classification methods obtained on the original variables with the accuracies obtained on the set of *best* CBSes applied to the eight benchmark datasets, and also we show average error rate reduction if only *best* CBSes are used for classification.

Table 4

COMPARISON OF CLASSIFICATION ACCURACIES ON ORIGINAL VARIABLES AND BEST CBSES

Dataset	Average accuracy of 5 classification methods	Maximum accuracy of 5 classification methods	Average accuracy obtained with best CBSes	Improvement over **average** accuracy of 5 classification methods	Improvement over **maximum** accuracy of 5 classification methods	Average error rate reduction
bld	63.37%	69.29%	73.42%	+10.05%	+4.13%	27.44%
ger	68.37%	72.21%	73.46%	+5.09%	+1.25%	16.09%
pid	73.78%	76.13%	81.93%	+8.15%	+5.80%	31.08%
hea	80.75%	83.05%	84.80%	+4.05%	+1.75%	21.04%
aus	85.32%	86.66%	88.32%	+3.00%	+1.66%	20.44%
ion	88.92%	91.58%	93.02%	+4.10%	+1.44%	37.00%
bcw	94.35%	95.28%	95.55%	+1.20%	+0.27%	21.24%
vot	96.34%	97.14%	96.79%	+0.45%	-0.35%	12.30%
			Average	+4.51%	+1.99%	23.33%

It can be seen that the average accuracy improvement provided by the CBSes

- *in comparison with the **average** accuracy* of the five classification methods applied to the original datasets, is 4.51%;
- *in comparison with the **maximum** accuracy* among the five classification methods applied to the original datasets, is 1.99%.

Moreover, it can be seen that for every dataset (with the possible exception of **vot –** a dataset which from the beginning allows an exceptionally accurate classification) the accuracy obtained on the set of *best* CBSes does not only exceed the *average* accuracy of the five examined classification methods, but surpasses even the accuracy given by the very best of these five methods. Also the table shows that *the average error rate is reduced by 23.33*% if only *best* CBSes are used for classification.

Composite Boolean Separators as Classification Systems

To evaluate a CBS as a classification system we have compared the accuracy of this system with that of several of the most frequently used machine-learning / data-mining methods.

Table 5 shows the average accuracies of various classification methods applied to the datasets in 2-folding experiments. These experiments were performed in the following way. The dataset was divided into two parts. One of them was used as a training set and remaining one as a test set. On the training set we constructed CBSes and chose one with the highest *CP*. The quality of this separator was checked on the test set. Then we exchanged training and test sets and repeated experiments. The same observations (before binarization) in training sets were used for construction of classifiers by other five classification methods and the same observations (before binarization) in test sets were used for validation of these classifiers. To compare the results we performed the paired two sample for means one-tail *t* test.

Table 5

RESULTS OF 2-FOLDING EXPERIMENTS USING SIX CLASSIFICATION METHODS

	bld	ger	pid	hea	aus	ion	bcw	vot	AVERAGE		t Stat	P(T<=t) one-tail
									Method	CBS		
SMO	50.35%	67.30%	73.10%	82.80%	86.20%	81.95%	95.33%	94.53%	78.95%		-1.55	0.08
CBS	65.29%	70.26%	79.11%	81.08%	85.79%	88.77%	92.96%	94.06%		82.16%		
MP	67.98%	64.33%	71.63%	77.50%	81.35%	83.23%	94.53%	94.50%	79.38%		-2.06	0.04
CBS	65.29%	70.26%	79.11%	81.08%	85.79%	88.77%	92.96%	94.06%		82.16%		
SL	64.40%	68.68%	72.73%	81.20%	86.20%	83.75%	94.45%	96.25%	80.96%		-1.12	0.15
CBS	65.29%	70.26%	79.11%	81.08%	85.79%	88.77%	92.96%	94.06%		82.16%		
C4.5	63.55%	66.18%	72.38%	73.98%	83.30%	87.38%	93.03%	97.13%	79.62%		-2.12	0.04
CBS	65.29%	70.26%	79.11%	81.08%	85.79%	88.77%	92.96%	94.06%		82.16%		
LAD	67.44%	71.97%	74.68%	82.19%	85.57%	91.15%	94.73%	96.51%	83.03%		1.06	0.16
CBS	65.29%	70.26%	79.11%	81.08%	85.79%	88.77%	92.96%	94.06%		82.16%		

Conclusion 1. *The results of t-tests applied to the average accuracies show that CBSes seem to be statistically better than multilayer perceptron, decision trees, support vector machines considered at the confidence level of at least 90%, seem somewhat better than simple logistic regression, and seem to be somewhat weaker than LAD. All in all, the method is definitely comparable with the other methods considered.*

3.2. Identification of Misclassified Observations

The quality of real-world datasets is usually not perfect. The presence of noise usually leads to negative effects such as decrease of the classification accuracy, increase of the size of the model, incorrect decisions, and many others. To enhance the quality of data, we propose here two techniques for identifying suspicious observations, i.e., those which were supposedly misclassified.

3.2.1. CBS Technique

3.2.1.1. Consistent Composite Boolean Separators and Suspicious Observations

A phenomenon observed in many datasets is that each observation in the dataset is classified in the same way by all (or almost all) the composite Boolean separators, i.e., if an observation is classified as positive or negative by one of the separators, then all (or almost all) the other separators classify it in the same way. This phenomenon is present in

particular in all the eight datasets examined above, as well as in both real-life medical datasets to be analyzed in Chapter 4. This motivates the following definitions.

Definition 1: *An observation will be called* strongly reliable *if it was classified correctly by all the CBSes.*

Definition 2: *An observation will be called* strongly suspicious *if it was classified erroneously by all the CBSes.*

We shall denote the set of all strongly reliable observations by R and the set of all strongly suspicious observations by S. Also, let T denote the "residual" set, i.e., the set of those observations in the dataset which do not belong to R or S.

In what follows, we examine the question of whether the classes to which the observations in S are assigned in the dataset are correct, i.e., whether their classifications by the CBSes are credible. In order to derive some useful conclusions about the partitioning of the dataset into the subsets R, S, and T, we have carried out a large number of computational experiments meant to clarify the characteristics of these subsets.

In the first experiment, to be called *strong deletion*, the accuracy of the five classification methods described in the introduction applied to all the observations in the dataset ($R \bigcup S \bigcup T$) was compared to that of the same methods applied to the observations in the set $R \bigcup T$ only, i.e., those remaining in the dataset after the deletion of the strongly

suspicious observations. The average accuracies obtained in twenty 10-folding cross-validation experiments carried out with the five methods on each of the eight datasets are shown in Table 6.

Table 6

RESULTS ON THE ORIGINAL DATASETS AND ON THE DATASETS OBTAINED AFTER DELETION OF STRONGLY SUSPICIOUS OBSERVATIONS

Dataset	Average accuracy of 5 classification methods		Average accuracy increase	Average error rate reduction	Size of dataset $R \bigcup T$
	Original dataset $R \bigcup S \bigcup T$	Dataset $R \bigcup T$ after deletion of strongly suspicious observations			
bld	63.37%	78.91%	15.54%	42.42%	69.60%
ger	68.37%	93.61%	25.24%	79.80%	74.00%
pid	73.78%	87.26%	13.48%	51.41%	84.90%
hea	80.75%	90.28%	9.54%	49.56%	90.90%
aus	85.32%	95.26%	9.93%	67.64%	89.90%
ion	88.92%	89.62%	0.70%	6.32%	95.50%
bcw	94.35%	97.11%	2.76%	48.85%	97.80%
vot	96.34%	99.50%	3.16%	86.34%	97.40%
Average	**81.40%**	**91.44%**	**10.04%**	**54.04%**	**87.50%**

An examination of the table above leads us to the following statement.

Conclusion 2. *By deleting the set S of strongly suspicious observations, we obtain a new dataset which includes on the average almost 90% of the observations, and on which the examined machine-learning / data-mining methods have on average a 10% higher accuracy and a 54% less error rate than on the original datasets.*

While the role of the first experiment was to demonstrate the predictability of the subset $R \cup T$ remaining after the deletion of the strongly suspicious observations, the role of the second experiment is to demonstrate the suspiciousness of the strongly suspicious subset *S*. For this purpose, we shall compare the average accuracies obtained in twenty 10-folding cross-validation experiments carried out on the original dataset $R \cup S \cup T$, with the average accuracies obtained by training on the set $R \cup T$ and testing on the strongly suspicious set *S*. After randomly partitioning in 20 different ways each of the datasets $R \cup T$ into 10 subsets, we have used in 20×10 experiments 9 of these subsets for training and tested the results on *S*. The average accuracies obtained in this way are shown in Table 7.

Table 7

RESULTS ON THE ORIGINAL DATASETS AND ON THE STRONGLY SUSPICIOUS SUBSETS S

Dataset	Average accuracy of 5 classification methods		Average accuracy decrease	Average error rate increase	Size of strongly suspicious subset
	Original dataset $R \cup S \cup T$	Strongly suspicious subset S			
bld	63.37%	27.26%	36.11%	49.64%	30.40%
ger	68.37%	8.30%	60.07%	65.51%	26.00%
pid	73.78%	12.12%	61.66%	70.16%	15.10%
hea	80.75%	21.38%	59.37%	75.52%	9.10%
aus	85.32%	3.03%	82.29%	84.86%	10.10%
ion	88.92%	39.20%	49.72%	81.78%	4.50%
bcw	94.35%	1.00%	93.35%	94.29%	2.20%
vot	96.34%	0.01%	96.33%	96.34%	2.60%
Average	**81.40%**	**14.04%**	**67.36%**	**77.26%**	**12.50%**

An examination of the table above leads us to the following statement.

Conclusion 3. *On the set S of strongly suspicious observations, which includes on the average 12.50% of the observations in the examined datasets, the average accuracy of the examined machine-learning / data-mining methods decreases by almost 70% and the average error rate increases by almost 80% compared to the original dataset.*

In light of the above conclusion it is natural to wonder whether the very low accuracy (or very high error rate) of classification methods on the strongly suspicious set *S* is due to

- errors in the given descriptions of attribute values in the dataset, or
- a difference in the nature of the observations in *S* compared to those in $R \bigcup T$, or
- errors in the given classifications of the observations in *S*.

The results of the second experiment can be presented in a different way by showing how the models learned on the set $R \bigcup T$ classify the "reversed" set $\overline{S}$ consisting of the observations in the set *S*, having reversed classifications (i.e., reversing the classification of a positive observation to negative, and of a negative one to positive). These results are shown in Table 8.

It can be seen that reversing the classification of the observations in the strongly suspicious set *S*, the accuracies on $\overline{S}$ become comparable to those on $R \bigcup S \bigcup T$.

Table 8

RESULTS ON THE ORIGINAL DATASETS AND ON THE REVERSED STRONGLY SUSPICIOUS SUBSETS $\overline{S}$

Dataset	Average accuracy of 5 classification methods		Average change in accuracy	Average error rate change
	Original dataset $R \cup S \cup T$	Reversed strongly suspicious subset $\overline{S}$		
bld	63.37%	72.74%	+9.37%	-25.58%
ger	68.37%	91.70%	+23.33%	-73.76%
pid	73.78%	87.88%	+14.10%	-53.78%
hea	80.75%	78.62%	-2.13%	+9.96%
aus	85.32%	96.97%	+11.65%	-79.36%
ion	88.92%	60.80%	-28.12%	+71.73%
bcw	94.35%	99.00%	+4.65%	-82.30%
vot	96.34%	99.99%	+3.65%	-99.73%
Average	**81.40%**	**85.96%**	**+4.56%**	**-41.60%**

Moreover, it is interesting to notice that in six of the eight datasets the accuracy on $\overline{S}$ is actually higher than that on $R \cup S \cup T$, the increase averaging at almost 5%. The only dataset on which the reversal produces a sizeable decrease in accuracy is **ion;** the other dataset on which there is a small (approx. 2%) decrease of accuracy is **hea,** on which however the accuracy found on $R \cup S \cup T$ and on $\overline{S}$ remain comparable. It also can be noticed that the average error rate was reduced by more than 40%. These observations lead to the following statement.

Conclusion 4. *The reversal of the given classifications of the strongly suspicious observations produces a set* $\overline{S}$ *of observations on which the machine-learning / data-mining methods examined in this study provide accuracies comparable with and usually higher than on the original dataset.*

In view of the above three conclusions, it is natural to ask which one of the two methods presented above, *deletion* or *reversal*, can produce better results. In order to answer this question we have compared the accuracies of the five machine-learning / data-mining methods on the eight datasets; the original dataset $R \cup S \cup T$, the dataset $R \cup T$ obtained by deletion (i.e., by the deletion of *S*), and the dataset $R \cup \overline{S} \cup T$ obtained by the reversal of the classifications of the strongly suspicious observations. The average results of twenty 10-folding cross-validation experiments are presented in Table 9. These results lead to the following statement.

Conclusion 5. *Regardless of the classification methods used, deletion and reversal improve the accuracy of classification, the average improvements in accuracy being of approximately 10% and 11% respectively; the improvement obtained by reversal is slightly higher in most cases than that obtained by deletion. Both deletion and reversal cut the error rate more than in half.*

Table 9

Dataset		SMO	MP	SL	C4.5	LAD	Average	Average increase in accuracy	Average error rate reduction
bld	Original	50.03%	67.63%	66.24%	63.65%	69.29%	63.37%		
	Deletion	50.65%	84.42%	78.67%	97.87%	82.94%	78.91%	15.54%	42.42%
	Reversal	69.56%	87.88%	70.98%	98.64%	84.11%	82.23%	18.87%	51.51%
ger	Original	69.39%	65.50%	68.56%	66.18%	72.21%	68.37%		
	Deletion	89.56%	95.41%	89.97%	97.37%	95.75%	93.61%	25.24%	79.79%
	Reversal	90.45%	95.73%	91.25%	98.57%	97.69%	94.74%	26.37%	83.36%
pid	Original	72.31%	73.81%	72.07%	76.13%	74.60%	73.78%		
	Deletion	84.13%	86.51%	85.81%	93.09%	86.77%	87.26%	13.48%	51.42%
	Reversal	85.95%	88.37%	87.84%	94.26%	87.61%	88.81%	15.03%	57.33%
hea	Original	83.05%	78.70%	82.48%	77.16%	82.35%	80.75%		
	Deletion	90.22%	90.73%	89.85%	91.44%	89.17%	90.28%	9.54%	49.55%
	Reversal	89.86%	89.89%	89.57%	92.72%	89.95%	90.40%	9.65%	50.12%
aus	Original	86.47%	82.98%	86.66%	84.93%	85.57%	85.32%		
	Deletion	95.26%	94.93%	95.44%	95.39%	95.26%	95.26%	9.93%	67.65%
	Reversal	95.34%	95.23%	95.77%	96.20%	95.30%	95.57%	10.25%	69.83%
ion	Original	91.10%	88.78%	85.08%	88.05%	91.58%	88.92%		
	Deletion	86.13%	89.93%	85.91%	92.13%	93.99%	89.62%	0.70%	6.32%
	Reversal	84.90%	89.09%	84.83%	93.59%	91.76%	88.83%	-0.08%	-0.72%
bcw	Original	95.28%	94.39%	94.86%	92.78%	94.44%	94.35%		
	Deletion	97.92%	96.90%	97.61%	95.78%	97.33%	97.11%	2.76%	48.85%
	Reversal	97.97%	96.93%	97.45%	95.65%	97.28%	97.06%	2.71%	47.96%
vot	Original	97.05%	94.42%	96.49%	96.61%	97.14%	96.34%		
	Deletion	99.58%	99.16%	99.58%	99.58%	99.60%	99.50%	3.16%	86.39%
	Reversal	99.58%	99.17%	99.58%	99.58%	99.59%	99.50%	3.16%	86.39%
						Average	**81.40%**		
						Average	**91.44%**	**10.04%**	**54.04%**
						Average	**92.14%**	**10.75%**	**55.71%**

Before concluding this section we shall return to the question of whether the observations in S have been misclassified in the original dataset. We have seen before that the models learned on $R \cup T$ and tested on S have very low accuracies (Table 7). Also, we have seen in Table 8 that the models learned on $R \cup T$ and tested on $\overline{S}$ have very high accuracies. In order to understand the structure of the strongly suspicious sets S and complete the tests we have developed and cross-validated a series of models on these sets. Since in some of the datasets the size of the sets S was too small to carry out the experiments, we have only performed it for the datasets **pid**, **bld**, **aus**, **ger** – whose strongly suspicious sets are sufficiently large. In Table 10 and Table 11 below we present the results of these experiments. These results are averages of twenty 10-folding experiments performed in the following way. The set S (respectively $\overline{S}$) is randomly partitioned into 10 approximately equally sized parts, 9 of which are used as a training set, and the resulting model is tested on $R \cup T$. In each of the 10 tests in a 10-folding experiment another part of S is removed.

Table 10

AVERAGE ACCURACY OF MODELS LEARNED ON S BY 5 CLASSIFICATION METHODS

D a t a s e t	Cross-validation on S	Testing on $R \cup T$
bld	82.11%	25.17%
pid	93.32%	15.93%
ger	89.30%	10.64%
aus	95.72%	5.28%

Table 11

AVERAGE ACCURACY OF MODELS LEARNED ON $\overline{S}$ BY 5 CLASSIFICATION METHODS

Dataset	Cross-validation on $\overline{S}$	Testing on $R \cup T$
bld	81.41%	74.86%
pid	93.18%	84.02%
ger	91.70%	91.65%
aus	95.80%	94.78%

The results shown in Table 10 indicate clearly that the sets *S* in these four datasets, considered in isolation, allow a very accurate classification. The same conclusion is also true for the sets $\overline{S}$ (see Table 11).

All in all, it is clear that the models built on *S*, respectively on $\overline{S}$, have high accuracy, and that $R \cup T$ is "inconsistent" with *S*, but it is perfectly consistent with $\overline{S}$.

Therefore it seems that *all the strongly suspicious observations in all the eight datasets examined were simply misclassified in their original version.*

3.2.1.2. Expanding the Suspicious Set

In the previous section we have defined as strongly reliable (respectively, as strongly suspicious) those observations for which (*i*) *all* the CBSes gave the same classification, and (*ii*) that classification coincided with (respectively, differed from) the classification given in the original dataset. In this section we shall relax the requirements of this

definition in order to identify more of the observations whose classifications given in the original dataset may be questionable.

Let c be the number of CBSes constructed as in Section 2.3, and let p be an arbitrary number in [0,1]. Let us give first an intuitive definition of a natural partition of Ω, which reflects the classifications given by the CBSes. Let us define the *p-reliable* subset R_p of the dataset Ω as the subset which consists of those observations for which the outcomes (1 or 0) of at least pc CBSes agree both among themselves and with their classifications (positive or negative) given in the dataset. Similarly, the *p-suspicious* subset S_p of Ω is defined as consisting of those observations for which the outcomes (1 or 0) of at least pc CBSes agree among themselves, but disagree with their classifications (positive or negative) given in the dataset. The remaining "divergent" subset T_p consists of the observations for which the CBS outcomes are split in such a way that both the number of 1's and that of 0's is less than pc; clearly, $T_p = \Omega \setminus (R_p \cup S_p)$. The observations in R_p and S_p will be called *p-reliable,* respectively *p-suspicious*. Clearly, the strongly reliable and strongly suspicious observations represent the special case of p-reliable, respectively p-suspicious, observations corresponding to $p = 1$. We shall examine later in this section a way of determining a good value of p.

In the computational experiments reported below the value of p was chosen to be 0.75. The following Table 12 presents the average accuracies obtained in twenty 10-folding experiments using five machine-learning / data-mining methods, as well as the sizes of the corresponding sets of p-suspicious and strongly suspicious observations.

Table 12

AVERAGE ACCURACY OF 5 CLASSIFICATION METHODS ON THE ORIGINAL DATASETS AND ON THE DATASETS OBTAINED AFTER DELETION AND REVERSAL (P=1 AND P=0.75)

Dataset	Average accuracy of 5 classification methods					Size	
	Original dataset	Strong deletion	Deletion for ***p*=0.75**	Strong reversal	Reversal for ***p*=0.75**	Strongly suspicious set	*p*-suspicious set for ***p*=0.75**
bld	63.37%	78.91%	78.91%	82.23%	82.23%	30.43%	30.43%
ger	68.37%	93.61%	94.57%	94.74%	95.11%	26.00%	26.50%
pid	73.78%	87.26%	88.81%	88.81%	89.96%	15.10%	16.07%
hea	80.75%	90.28%	93.53%	90.40%	93.61%	9.10%	12.46%
aus	85.32%	95.26%	96.82%	95.57%	96.74%	10.10%	11.88%
ion	88.92%	89.62%	89.74%	88.83%	89.13%	4.50%	5.11%
bcw	94.35%	97.11%	97.61%	97.06%	97.65%	2.20%	2.90%
vot	96.34%	99.50%	99.48%	99.50%	97.81%	2.60%	3.88%
Average	81.40%	91.44%	92.43%	92.14%	92.78%	12.50%	13.65%

The following conclusions can be drawn from these results:

- the average number of observations in the *p*-suspicious sets exceeds the average number of observations in the strongly suspicious sets by about 1% of the size of the datasets;
- on every dataset studied, the accuracy of classification after deletion for $p = 0.75$ is higher than classification after strong deletion; the average increase in accuracy being of approximately 1%;
- on every dataset studied, with the exception of **vot,** the accuracy of classification after reversal for $p = 0.75$ is higher than classification

after strong reversal; the average increase in accuracy being of approximately 0.6%;

- the accuracy of classification after reversal for $p = 0.75$ is moderately increased in most datasets compared to classification after deletion for $p = 0.75$, the average increase being of approximately 0.4%.

In order to illustrate the influence of various values of p on the accuracy of deletion and reversal we shall consider the dataset **hea**, for which the number of CBSes is 12. In Table 13 we show the effects of deleting and of reversing all the suspicious observations, for different values of the parameter p between 0.75 and 1. Since in this dataset there are no observations in which the values of *exactly* 9 (= 0.75 x 12) CBSes coincide, the reliable and suspicious sets are defined by the observations in which the values of at least 10 separators coincide; these sets of observations corresponds to p less than 0.917. When p exceeds 0.917, at least 11 separators have to agree in the observations defining R_p and S_p, and for $p = 1.0$ this number is 12.

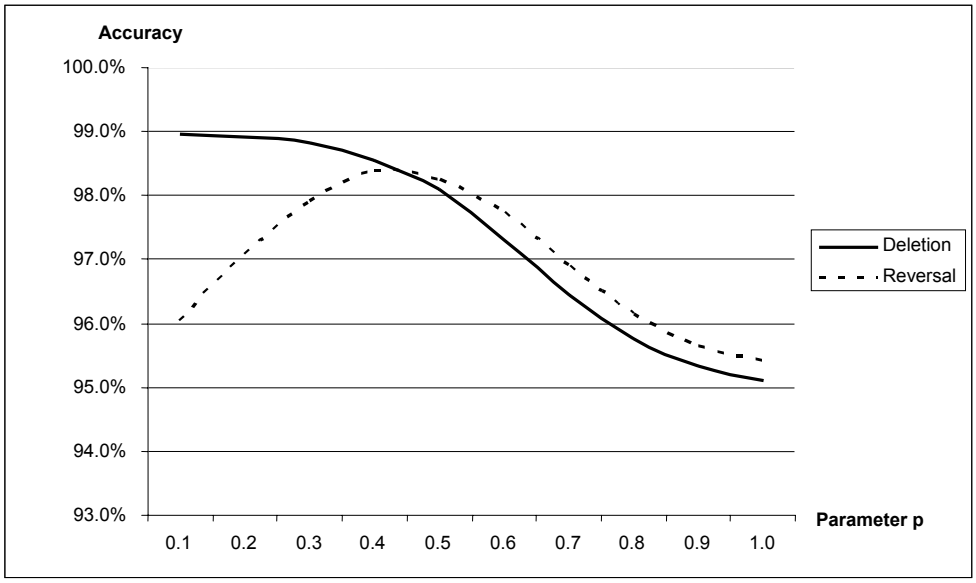
Table 13

AVERAGE ACCURACY OF 5 CLASSIFICATION METHODS FOR THE DATASET **HEA** ($0.75 \leq p \leq 1$)

Value of p	Average accuracy of 5 classification methods		Size of $\lvert S_p \rvert$
	Deletion	Reversal	
$0.75 \leq p < 0.917$	93.53%	93.61%	12.46%
$0.917 \leq p < 1.0$	92.91%	92.92%	11.78%
$p = 1.0$	90.28%	90.40%	9.09%

In order to identify a good value of the parameter p, let us make some observations, based on the accumulated experimental evidence. First, we have noticed in our experiments that for large values of $p \in [0,1]$ the accuracy of classification after reversal is generally higher than the accuracy of classification after deletion. Second, it was remarked that the accuracy of classification after deletion increases monotonically when p decreases. Third, it was also observed that the accuracy of classification after reversal increases with decreasing p until it reaches a peak, after which it starts decreasing. The second and third remarks indicate that – if we disregard small irregularities – the accuracy of deletion is a monotonically non-increasing function of p, while the accuracy of reversal is a unimodal function of p. The dependence of the accuracy of deletion and of reversal on p is illustrated in Figure 3. It is important to remember however that this picture provides only an approximate description of the real phenomenon.

Figure 3. **Finding a good value of *p***



Based on the above, it is natural to assume that for high values of p the suspicious set S includes only a part of those observations whose classification is perhaps erroneous, while for low values of p too many observations are included in S. Our objective is to find a true set of misclassified observations. Therefore, we do not want to leave out those observations which are really misclassified or to include those which are not. In this respect, the following hypothesis seems reasonable.

Hypothesis: *the optimal value of the parameter p is that one for which the accuracy of deletion is closest to that of reversal*.

In the computational experiments reported in this thesis, we have used a simple heuristic for finding a relatively good value p^* of the parameter p. For evaluating the accuracies of classifying the various sets of observations in this process we have always used twenty 10-folding cross-validation experiments with each of the five machine-learning / data-mining methods listed in the introduction, and reported the average accuracy of these 1,000 experiments. For every $t = 1, 2,\ldots$ let the suspicious set S_t consist of those observations in which at least c_t of the total number c of CBSes have the same value, and this value differs from the given classification of the corresponding observations. Let us define c_t to be $c - t + 1$, and p_t to be $\frac{c_t}{c}$. We shall denote by $\delta(t)$ and $\rho(t)$ the accuracy of classification on the dataset $\Omega \setminus S_t$ obtained by deletion, respectively the accuracy of classification on the dataset $(\Omega \setminus S_t) \cup \overline{S}_t$ obtained by reversal, and we define the *tolerance* ε to be an arbitrary small nonnegative number. Let $\hat{t}$ be the first index t

for which $|\rho(t) - \delta(t)| \leq \varepsilon$, and let $t^* = \min\{\hat{t}, \lfloor c/2 \rfloor + 1\}$. We shall take $p^* = p_{t^*} = 1 - \frac{(t^*-1)}{c}$. This value was chosen so as to satisfy the property of p^* stated by the hypothesis given above, while making sure that the set of CBSes used in the definition of the p-suspicious set S_{p^*} includes at least half of all CBSes.

In the next table we shall show the influence of the choice of the value of p on the accuracy of deletion and reversal, as well as on the size of the suspicious set. It can be seen that while strong deletion as well as strong reversal can improve the average accuracy by more than 10%, deletion or reversal using the value p^* of the parameter p can add to this a further improvement of 1-2%. Also, while in strong deletion or strong reversal the size of the suspicious set averages at 12.5% of Ω, the size of S_{p^*} averages at 13.7%.

Table 14

RESULTS ON THE DATASETS OBTAINED AFTER DELETION AND REVERSAL (P=1, P=0.75, P=P*)

Dataset	p^*	Average accuracy of 5 classification methods							Size of suspicious set for		
		Original dataset	Deletion for			Reversal for					
			p=1	p=0.75	$p=p^*$	p=1	p=0.75	$p=p^*$	p=1	p=0.75	$p=p^*$
bld	0.53	63.37%	78.91%	78.91%	79.40%	82.23%	82.23%	82.32%	30.4%	30.4%	31.3%
ger	0.5	68.37%	93.61%	94.57%	95.02%	94.74%	95.11%	95.39%	26.0%	26.5%	27.0%
pid	0.53	73.78%	87.26%	88.81%	89.32%	88.81%	89.96%	90.42%	15.1%	16.1%	17.6%
hea	0.83	80.75%	90.28%	93.53%	93.53%	90.40%	93.61%	93.61%	9.1%	12.5%	12.5%
aus	0.91	85.32%	95.26%	96.82%	96.77%	95.57%	96.94%	96.84%	10.1%	11.9%	11.6%
ion	1.00	88.92%	89.62%	89.74%	89.62%	88.83%	89.13%	88.83%	4.5%	5.1%	4.5%
bcw	1.00	94.35%	97.11%	97.61%	97.11%	97.06%	97.65%	97.06%	2.2%	2.9%	2.2%
vot	1.00	96.34%	99.50%	99.48%	99.50%	99.50%	97.81%	99.50%	2.6%	3.9%	2.6%

An interesting question concerning the suspicious sets is to know whether there is a clear relationship between their sizes and the improvement of accuracy by deletion or reversal. Table 15 provides an affirmative answer to this question. It shows that, when $p \geq p^*$ the correlation between $|S|$ and the possible accuracy improvements is of 0.88 for deletion and 0.92 for reversal. Moreover, it can also be seen from the table that there is a strong negative correlation between average accuracy on the original data and its possible improvement by deletion or reversal; not surprisingly there is a -0.98 correlation between the average accuracy on the original set and the size of the suspicious set.

Table 15

CORRELATIONS BETWEEN ACCURACY ON ORIGINAL DATA, IMPROVEMENTS BY DELETION AND REVERSAL, AND SIZE OF SUSPICIOUS SET FOR $P \geq P^*$

		Average accuracy of 5 classification methods on original dataset	Improvement of average accuracy of 5 classification methods by		Size of suspicious set
			Deletion	Reversal	
Average accuracy of 5 machine classification methods on original dataset			-0.86	-0.89	-0.98
Improvement of average accuracy of 5 classification methods by	Deletion			0.99	0.88
	Reversal				0.92
Size of suspicious set					

3.2.2. Best SER (Simulated Error Rate) Technique

In this section, we analyze one more method for identifying suspicious observations. This method is based on the notion of simulated error rate.

Definition 3. *The Simulated Error Rate (SER) of an observation w in N k-folding experiments is the average number of times the observation, while being in the test set, was wrongly classified by a method C, i.e.,* $\sigma_C(w) = \frac{e_w}{N}$, *where* e_w *is the number of erroneous classifications of observation* $w \in \Omega$ *by the classification method C in the N experiments.*

The procedure for identifying suspicious observations based on the notion of *SER* can be described as follows.

Procedure *Best_SER*

1. repeat *k*-folding *N* times for each classification method C_i (i = 1, 2, 3, …..);
2. compare average classification accuracies of the methods C_i for i = 1, 2, 3, … obtained on test sets;
3. choose a method *C* which gives the *best* accuracy (if there is a tie, i.e., several methods give the same best accuracy, then pick any of those);
4. using the chosen method, for each observation in the dataset evaluate *SER* on the test sets;
5. create the subset of suspicious observations by including in it the observations *w* with $\sigma_C(w) = 1$.

Since the condition in step 5 of this procedure is quite strong, it may lead to omitting some misclassified observations from the suspicious set. To overcome this difficulty we at times relax this condition by introducing a numerical parameter $\alpha \in [0,1]$ and replacing the requirement $\sigma_C(w) = 1$ with $\alpha \leq \sigma_C(w) \leq 1$. With this modification, the above procedure will be referred to as *Best_SER*(α). The set of observations found by *Best_SER*(α) will be denoted S_α.

To identify the optimal value of the parameter α we adapt the ***hypothesis*** proposed in Section 3.2.1.2 in the following way: it seems reasonable to expect that the *optimal value of* α *is the one for which the accuracy obtained by deletion is closest to that obtained by reversal*.

Let δ_α be the accuracy of classification on the dataset $\Omega \setminus S_\alpha$ obtained by deletion of the set S_α, and let ρ_α be the accuracy of classification on the dataset $(\Omega \setminus S_\alpha) \cup \overline{S}_\alpha$ obtained by reversal of the class of the observations in the set S_α. According to the above hypothesis, we want the value of $|\rho_\alpha - \delta_\alpha|$ to be within some *tolerance* ε which is a small nonnegative number. If $\rho_\alpha - \delta_\alpha > \varepsilon$, we gradually increase the parameter α until $|\rho_\alpha - \delta_\alpha| \leq \varepsilon$. If $\rho_\alpha - \delta_\alpha < -\varepsilon$, we gradually decrease α until $|\rho_\alpha - \delta_\alpha| \leq \varepsilon$.

The idea of using *SER* for filtering data is not new. For example, Brodley and Friedl (see [21], [22]) employed this idea in a two-step procedure for identifying and eliminating mislabeled training instances. The difference between the technique proposed by Brodley and Friedl and the technique proposed in this section is the following. The authors of

[21], [22] use several machine-learning / data-mining methods to compute *SER* for each observation, and qualify an observation as suspicious if most of the methods (*Majority*) or all of them (*Consensus*) indicate that this is an error. In our approach, out of several methods we choose the one which gives the *best* average accuracy on the test sets and calculate *SER* of each observation for this method only. We call this approach the *Best SER* technique.

To justify the proposed modification we implement two series of experiments. In the first series, we use four classification methods and classify an observation as suspicious if two methods having the highest accuracies indicate that the observation is misclassified in every one of the twenty 10-folding experiments. We call this approach *2-consensus*. In the second series, we implement the *Best SER* technique with the same four classification methods. The results reported in Table 16 refer to the eight datasets described in Table 1 with the following modifications: we do not delete missing data, but we delete repetitions in all these datasets.

Our experiments show that the first approach is more conservative than the *Best SER,* since it eliminates fewer instances from the data. The drawback of a conservative approach is the risk of retaining "bad" data. It can be seen from Table 16 that after deletion/reversal for each machine-learning / data-mining method and for each dataset the accuracy increases for both techniques. However for the *Best SER* technique the average increase is almost twice as much as that obtained by the first approach. The minimum

Table 16

COMPARISON OF 2-CONSENSUS WITH BEST SER

Dataset		SMO		MP		SL		C4.5		Average		Average increase in accuracy		Average error rate reduction	
		2-consensus	*Best SER*	*2-consensus*	*Best SER*	*2-consensus*	*Best SER*	*2-consensus*	*Best SER*	*2-consensus*	*Best SER*	*2-consensus*	*Best SER*	*2-consensus*	*Best SER*
bld	Original	52.39%	52.39%	67.21%	67.21%	66.46%	66.46%	63.05%	63.05%	62.28%	62.28%				
	Deletion	54.53%	54.60%	74.45%	76.55%	72.09%	72.42%	69.18%	69.79%	67.56%	68.34%	5.28%	6.07%	14.00%	16.09%
	Reversal	55.83%	56.42%	77.34%	78.81%	73.31%	74.46%	67.33%	68.55%	68.45%	69.56%	6.17%	7.28%	16.36%	19.30%
ger	Original	62.44%	62.44%	64.67%	64.67%	68.71%	68.71%	63.61%	63.61%	64.86%	64.86%				
	Deletion	66.51%	86.84%	68.82%	88.06%	70.39%	89.26%	66.61%	82.34%	68.08%	86.63%	3.22%	21.77%	9.16%	61.95%
	Reversal	67.60%	88.64%	69.37%	90.49%	70.96%	91.82%	66.47%	81.41%	68.60%	88.09%	3.74%	23.23%	10.64%	66.11%
pid	Original	70.29%	70.29%	72.24%	72.24%	71.18%	71.18%	70.17%	70.17%	70.97%	70.97%				
	Deletion	75.69%	75.75%	79.00%	80.18%	76.70%	76.73%	75.49%	76.03%	76.72%	77.17%	5.75%	6.20%	19.81%	21.36%
	Reversal	77.40%	77.04%	81.41%	81.48%	78.23%	77.98%	77.85%	77.61%	78.72%	78.53%	7.75%	7.56%	26.70%	26.04%
hea	Original	78.81%	78.81%	78.52%	78.52%	82.93%	82.93%	75.51%	75.51%	78.94%	78.94%				
	Deletion	85.91%	94.39%	85.63%	93.59%	87.30%	94.96%	83.30%	89.93%	85.54%	93.22%	6.59%	14.28%	31.29%	67.81%
	Reversal	86.86%	95.28%	86.64%	94.63%	88.54%	95.54%	83.00%	90.77%	86.26%	94.06%	7.32%	15.11%	34.76%	71.75%
aus	Original	84.82%	84.82%	82.75%	82.75%	86.73%	86.73%	81.79%	81.79%	84.02%	84.02%				
	Deletion	92.29%	93.60%	91.35%	93.65%	93.44%	95.03%	91.20%	92.80%	92.07%	93.77%	8.05%	9.75%	50.38%	61.01%
	Reversal	92.82%	94.31%	91.99%	94.09%	93.79%	95.35%	92.17%	93.24%	92.69%	94.25%	8.67%	10.23%	54.26%	64.02%
ion	Original	85.33%	85.33%	88.02%	88.02%	84.41%	84.41%	88.02%	88.02%	86.44%	86.44%				
	Deletion	86.85%	88.87%	88.41%	89.02%	84.25%	87.57%	88.95%	91.12%	87.12%	89.15%	0.67%	2.70%	4.94%	19.91%
	Reversal	86.59%	88.27%	88.59%	88.26%	84.61%	87.58%	88.88%	91.37%	87.17%	88.87%	0.72%	2.42%	5.31%	17.85%
bcw	Original	93.66%	93.66%	93.37%	93.37%	94.47%	94.47%	91.43%	91.43%	93.23%	93.23%				
	Deletion	96.97%	99.09%	96.15%	98.85%	96.68%	98.56%	94.70%	96.25%	96.13%	98.19%	2.89%	4.96%	42.69%	73.26%
	Reversal	96.93%	99.18%	96.28%	99.08%	96.68%	98.72%	94.81%	95.60%	96.18%	98.15%	2.94%	4.91%	43.43%	72.53%
vot	Original	92.18%	92.18%	93.08%	93.08%	94.62%	94.62%	92.72%	92.72%	93.15%	93.15%				
	Deletion	97.22%	99.48%	97.56%	99.06%	97.09%	98.88%	98.11%	99.58%	97.50%	99.25%	4.35%	6.10%	63.50%	89.05%
	Reversal	96.29%	98.56%	97.64%	99.15%	97.01%	98.32%	96.86%	98.39%	96.95%	98.61%	3.80%	5.46%	55.47%	79.71%
									Average	79.24%	79.24%				
									Average	83.84%	88.21%	4.60%	8.98%	29.47%	51.31%
									Average	84.38%	88.76%	5.14%	9.53%	30.87%	52.16%

increase in accuracy for the eight datasets is 2.42% for *Best SER* and 0.67% for *2-consensus*; the maximum increase in accuracy for the eight datasets is 23.23% for *Best SER* and 8.67% for *2-consensus.* It also can be seen that deletion/reversal reduce the average error rate by more than half if *Best SER* is used and by only 30% if *2-consensus* is used. This discussion shows that the *Best SER* technique is a promising modification of the Brodley-Friedl's idea.

Now let us present a detailed study of the *Best SER* technique in which we use five machine-learning / data-mining methods described in the introduction. For each dataset presented in Table 1 and each method we run twenty 10-folding experiments ($k = 10$ and $N = 20$). Table 17 presents the average accuracies obtained in these experiments. For each dataset we emphasize in bold the best accuracy.

Table 17

BEST CLASSIFICATION ACCURACIES FOR BENCHMARK DATASETS

	SMO	MP	SL	C4.5	LAD	Average
bld	50.03%	67.63%	66.24%	63.65%	**69.29%**	63.37%
ger	69.39%	65.50%	68.56%	66.18%	**72.21%**	68.37%
pid	72.31%	73.81%	72.07%	**76.13%**	74.60%	73.78%
hea	**83.05%**	78.70%	82.48%	77.16%	82.35%	80.75%
aus	86.47%	82.98%	**86.66%**	84.93%	85.57%	85.32%
ion	91.10%	88.78%	85.08%	88.05%	**91.58%**	88.92%
bcw	**95.28%**	94.39%	94.86%	92.78%	94.44%	94.35%
vot	97.05%	94.42%	96.49%	96.61%	**97.14%**	96.34%

It can be seen that LAD provides the maximum accuracy for the **bld**, **ger**, **ion**, and **vot** datasets. For **hea** and **bcw**, the best results were obtained by using the SMO method. Simple Logistic and C4.5 give the best accuracies for the **aus** and **pid** datasets, respectively.

For each dataset we use the method that gives the best accuracy in order to calculate *SER* of each observation in the set and construct the set S_α with $\alpha = 1$, i.e., we include in S_α those observations that have been misclassified 20 times. Next, we use the obtained subsets of suspicious observations for data cleaning in two ways: we either delete this subset from the dataset or reverse the class of observations in this subset. In Table 18, we present the original results as well as the results of deletion and reversal. These results lead us to the following conclusion.

Conclusion 6. *Both deletion and reversal improve the average accuracy of classification, the improvement being approximately 7.5%, and the difference between these two improvements in favor of reversal is only 0.21%. Both deletion and reversal cut the error rate almost in half. The accuracy increases (the error rate decreases) for each classification method regardless of the method used for the construction of the suspicious set.*

Table 18

AVERAGE ACCURACY ON THE ORIGINAL DATASETS AND ON THE DATASETS OBTAINED AFTER DELETION AND REVERSAL

($\alpha = 1$)

Dataset		SMO	MP	SL	C4.5	LAD	Average	Average increase in accuracy	Average error rate reduction
bld	Original	50.03%	67.63%	66.24%	63.65%	69.29%	63.37%		
	Deletion	54.23%	75.85%	75.85%	76.42%	78.55%	72.18%	8.81%	24.05%
	Reversal	50.99%	76.30%	73.00%	73.80%	80.36%	70.89%	7.52%	20.53%
ger	Original	69.39%	65.50%	68.56%	66.18%	72.21%	68.37%		
	Deletion	86.39%	85.77%	86.77%	83.49%	85.16%	85.52%	17.15%	54.22%
	Reversal	88.28%	86.21%	88.47%	85.43%	86.63%	87.00%	18.63%	58.90%
pid	Original	72.31%	73.81%	72.07%	76.13%	74.60%	73.78%		
	Deletion	75.41%	76.91%	77.42%	83.95%	80.15%	78.77%	4.99%	19.03%
	Reversal	76.65%	77.05%	78.15%	85.16%	81.37%	79.68%	5.90%	22.50%
hea	Original	83.05%	78.70%	82.48%	77.16%	82.35%	80.75%		
	Deletion	95.33%	94.26%	95.05%	88.54%	92.28%	93.09%	12.34%	64.10%
	Reversal	95.89%	95.73%	96.39%	89.53%	92.06%	93.92%	13.17%	68.42%
aus	Original	86.47%	82.98%	86.66%	84.93%	85.57%	85.32%		
	Deletion	94.22%	93.77%	94.99%	94.31%	93.61%	94.18%	8.86%	60.35%
	Reversal	95.03%	94.29%	95.30%	95.23%	93.80%	94.73%	9.41%	64.10%
ion	Original	91.10%	88.78%	85.08%	88.05%	91.58%	88.92%		
	Deletion	88.36%	90.69%	89.55%	91.07%	96.70%	91.27%	2.35%	21.21%
	Reversal	88.20%	89.01%	89.92%	91.32%	96.21%	90.93%	2.01%	18.14%
bcw	Original	95.28%	94.39%	94.86%	92.78%	94.44%	94.35%		
	Deletion	98.55%	98.00%	97.28%	93.23%	96.97%	96.80%	2.45%	43.36%
	Reversal	98.06%	97.73%	97.23%	94.22%	96.68%	96.79%	2.44%	43.19%
vot	Original	97.05%	94.42%	96.49%	96.61%	97.14%	96.34%		
	Deletion	99.58%	99.56%	99.58%	99.58%	99.61%	99.58%	3.24%	88.52%
	Reversal	99.04%	99.87%	99.17%	98.59%	99.20%	99.18%	2.84%	77.60%
						Average	**88.93%**	**7.53%**	**46.86%**
							89.14%	**7.74%**	**46.67%**

It can be seen from Table 18 that the largest accuracy difference between reversal and deletion was obtained on the dataset **ger** and it is equal to 1.49%. This suggests that perhaps the value $\alpha = 1$ is not optimal for the set **ger**. We define ε to be equal to 1% and

decrease α to 0.95, i.e., the observations which were erroneously classified 19 or 20 times are considered to be suspicious. With this relaxation of α, the size of the suspicious set increases from 17.10% to 18.80% of the total number of the observations in the dataset **ger**. The results obtained for deletion and reversal after the relaxation are shown in Table 19.

Table 19

AVERAGE ACCURACY OF DELETION AND REVERSAL ON THE DATASET **GER**

(α = 0.95)

Dataset		SMO	MP	SL	C4.5	LAD	Average	Average increase in accuracy	Average error rate reduction
ger	Deletion	88.14%	86.48%	88.20%	85.78%	86.24%	86.97%	18.60%	58.80%
	Reversal	88.94%	86.84%	88.99%	88.22%	86.35%	87.87%	19.50%	61.65%

After the relaxation the difference between the average accuracy obtained after reversal and the average accuracy obtained after deletion becomes less than 1% and further relaxation is not needed.

In order to verify the correctness of the construction of the set S_α, we run one more series of experiments. The goal of these experiments is to compare the average accuracies obtained on the original dataset with the average accuracies obtained by training on the set $\Omega \setminus S_\alpha$ and testing on the suspicious set S_α. After randomly partitioning in 20 different ways each of the datasets $\Omega \setminus S_\alpha$ into 10 subsets, we use in 20×10 experiments 9 of these subsets for training and test the results on S_α. The obtained results are presented in Table 20.

Table 20

AVERAGE ACCURACY OF 5 CLASSIFICATION METHODS ON THE ORIGINAL DATASETS AND SUSPICIOUS SUBSETS

($\alpha = \alpha^*$)

($\alpha = \alpha^*$) Dataset	Average accuracy of 5 classification methods		Average accuracy decrease	Average error rate increase	Size of suspicious subset S_α
	Original dataset	Suspicious subset S_α			
bld	63.37%	29.34%	34.03%	48.16%	13.62%
ger	68.37%	14.59%	53.78%	62.97%	18.80%
pid	73.78%	14.97%	58.81%	69.16%	6.12%
hea	80.75%	5.90%	74.85%	79.54%	12.46%
aus	85.32%	2.93%	82.39%	84.88%	8.41%
ion	88.92%	25.88%	63.04%	85.05%	5.11%
bcw	94.35%	5.57%	88.78%	94.02%	3.12%
vot	96.34%	11.64%	84.70%	95.86%	3.02%
Average	**81.40%**	**13.85%**	**67.55%**	**77.45%**	**8.83%**

The following conclusion can be made from the above table.

Conclusion 7*. The average accuracy on the suspicious subset S_α is very low and is almost 70% less than on the original dataset. The average error rate on this subset is 77.5% more than on the original dataset.*

We conclude this section by reporting results that describe the correlation between possible accuracy improvement obtained after deletion and reversal and the size of the

suspicious subset S_α. These results are valid only for values of $\alpha \geq \alpha^*$, where α^* stands for the optimal value of α.

Table 21

CORRELATIONS BETWEEN IMPROVEMENTS BY DELETION AND REVERSAL AND SIZE OF SUSPICIOUS SET FOR

$\alpha \geq \alpha^*$

$\alpha \geq \alpha^*$		Improvement of average accuracy of 5 classification methods by		Size of suspicious set
		Deletion	Reversal	
Improvement of average accuracy of 5 classification methods by	Deletion		0.99	0.94
	Reversal			0.9
Size of strongly suspicious set				

3.2.3. Comparison of the Results Obtained by CBS and Best SER Techniques

In the two previous sections we described two different techniques for identifying suspicious observations. Table 22 summarizes the results obtained by means of these techniques in terms of average increase in accuracy for deletion/reversal, in terms of average error rate reduction for deletion/reversal, and in terms of the size of suspicious sets. The average increase in accuracies presented in the table was obtained in the following way. 1,000 experiments (twenty 10-folding cross-validation experiments for

five machine-learning / data-mining methods) were performed for each original dataset described in Table 1. The same experiments were performed for the data obtained after deletion of the suspicious observations from the original datasets and for the data obtained after reversal of the class of the suspicious observations. The average increase in accuracy for deletion/reversal is the difference between the average accuracy of 1,000 experiments obtained after deletion/reversal and the average accuracy of 1,000 experiments obtained on the original data.

Table 22

COMPARISON OF CBS TECHNIQUE WITH BEST SER TECHNIQUE

Dataset	Average increase in accuracy if suspicious set was obtained by using		Average error rate reduction if suspicious set was obtained by using		Size of suspicious set obtained by using	
	CBS technique (p=p*)	*Best SER* technique (α=α*)	CBS technique (p=p*)	*Best SER* technique (α=α*)	CBS technique (p=p*)	*Best SER* technique (α=α*)
	Deletion Reversal	Deletion Reversal	Deletion Reversal	Deletion Reversal		
bld	16.03% 18.95%	8.81% 7.52%	43.76% 51.73%	24.05% 20.53%	31.30%	13.62%
ger	26.65% 27.02%	18.60% 19.50%	84.26% 85.43%	58.80% 61.65%	27.00%	18.80%
pid	15.54% 16.64%	4.99% 5.90%	59.27% 63.46%	19.03% 22.50%	17.60%	6.12%
hea	12.78% 12.86%	12.34% 13.17%	66.39% 66.81%	64.10% 68.42%	12.46%	12.46%
aus	11.45% 11.52%	8.86% 9.41%	78.00% 78.47%	60.35% 64.10%	11.60%	8.41%
ion	0.70% -0.09%	2.35% 2.01%	6.32% -0.81%	21.21% 18.14%	4.50%	5.11%
bcw	2.76% 2.71%	2.45% 2.44%	48.85% 47.96%	43.36% 43.19%	2.20%	3.12%
vot	3.16% 3.16%	3.24% 2.84%	86.34% 86.34%	88.52% 77.60%	2.60%	3.02%
Average	11.13% 11.60%	7.71% 7.85%	59.15% 59.92%	47.43% 47.02%	13.66%	8.83%

It can be seen that the average size of the suspicious subsets obtained by *Best SER* is less than that obtained with CBSes by 4.83%. It also can be seen that the average increase in accuracy obtained for deletion by CBSes is greater than that obtained by *Best SER* by 3.43%. Also, the average increase in accuracy obtained for reversal by CBSes is greater than that obtained by *Best SER* by 3.73%. Moreover, it can be seen that the average error rate for deletion (reversal) is reduced by almost 12% (13%) more if the CBS technique is applied. The results in Table 22 show that for the dataset **hea** the size of the suspicious subsets obtained by different techniques is the same and the results for deletion/reversal are close to each other (the average increase in accuracy is approximately 12.5%). The *Best SER* technique gives better results for the **ion** dataset (the average improvement is approximately 2%)**.** This is the only dataset for which CBSes do not work properly (deletion and reversal of the set of suspicious observations do not improve the average result obtained on the original data). It also can be noted that the size of the suspicious subset obtained by *Best SER* for **bcw** is greater (by 0.92%), but the average accuracy for deletion/reversal is slightly greater (by 0.31% / 0.27%) if the *CBS* technique is used. Based on the above, the following conclusion can be made.

Conclusion 8. *Each of the two techniques under the study has its own advantages and disadvantages. In other words, these techniques are generally incomparable. However, in most datasets that have been analyzed the CBS technique achieves better results, i.e., the average increase in accuracy and the average error rate reduction obtained by CBSes is lager than that obtained by Best SER. The average accuracy decrease on the suspicious set for both techniques is almost 70%.*

Now let us compare the suspicious sets obtained by the two different methods. Table 23 shows the sizes of the suspicious sets (the numbers of observations) for each dataset obtained by means of the CBS and *Best SER* techniques and the number of the suspicious observations belonging to the intersection of these two subsets.

Table 23

COMPARISON OF SUSPICIOUS SUBSETS OBTAINED BY CBSES AND BEST SER

Dataset	Size of suspicious subset obtained using		Size of the intersection of 2 suspicious subsets
	CBSes	*Best SER*	
bld	108	47	29
ger	270	188	162
pid	69	24	22
hea	37	37	22
aus	80	58	53
ion	16	18	7
bcw	10	14	7
vot	6	7	6

It is interesting to note that all suspicious observations of **vot** found by one of the techniques have been also found by the other one. For the other datasets the situation is somewhat similar: most of the suspicious observations found by one of the techniques have been also found by the other one. To make this statement more formal, let us calculate the ratio of the size of the intersection of the two subsets to the smaller one of them. For instance, for the dataset **vot** this ratio is equal to 100%. The average of this ratio computed over all datasets is more than 75%. This suggests the following conclusion:

Conclusion 9. *The subsets of the suspicious observations obtained by using the CBS and Best SER techniques have a large intersection.*

3.2.4. The Results for Deletion/Reversal on the Intersection and on the Union of Two Suspicious Sets

We have seen before that the suspicious subsets obtained by means of the CBS and *Best SER* techniques have a large intersection. Let us analyze the results for deletion/reversal on this intersection. It is obvious that the size of this set is less than or equal to the size of the smaller subset of suspicious observations found by the two techniques. The observations belonging to the intersection are "more suspicious" than others, since they were identified by both methods. It is interesting to observe the change in accuracy if we delete or reverse only these observations. In Table 24 we present the average results of twenty 10-folding cross-validation experiments of five machine-learning / data-mining methods obtained for the eight datasets.

Table 24 shows that the average size of the suspicious sets for the given datasets is 6.43%; the average increases in accuracy for deletion and for reversal are 5.65% and 6.41%, respectively. It also can be seen from the table that the error rate is reduced by 34.33% for deletion and by 37.44% for reversal. It indicates that *some misclassified observations were not included in the suspicious subsets*. The table also shows that the original result for the **ion** dataset is not improved. The correlation between possible accuracy improvement of deletion and of reversal and the size of the suspicious set obtained by intersection is 0.97 and 0.98, respectively.

Table 24

RESULTS FOR DELETION AND REVERSAL ON THE INTERSECTION

Dataset		SMO	MP	SL	C4.5	LAD	Average	Average increase in accuracy	Average error rate reduction	Size of suspicious set
bld	Original	50.03%	67.63%	66.24%	63.65%	69.29%	63.37%			8.41%
	Deletion	50.00%	74.52%	71.64%	75.50%	74.90%	69.31%	5.94%	16.22%	
	Reversal	50.00%	75.62%	73.29%	78.73%	76.60%	70.85%	7.48%	20.42%	
ger	Original	69.39%	65.50%	68.56%	66.18%	72.21%	68.37%			16.20%
	Deletion	84.88%	82.49%	84.28%	82.76%	84.78%	83.84%	15.47%	48.91%	
	Reversal	86.61%	84.45%	86.31%	86.52%	86.85%	86.15%	17.78%	56.21%	
pid	Original	72.31%	73.81%	72.07%	76.13%	74.60%	73.78%			5.61%
	Deletion	75.19%	74.49%	77.49%	82.90%	79.31%	77.88%	4.10%	15.64%	
	Reversal	75.81%	75.07%	78.64%	84.06%	80.12%	78.74%	4.96%	18.92%	
hea	Original	83.05%	78.70%	82.48%	77.16%	82.35%	80.75%			7.41%
	Deletion	90.61%	89.07%	89.83%	86.76%	87.97%	88.85%	8.10%	42.08%	
	Reversal	91.13%	89.98%	90.59%	88.47%	88.78%	89.79%	9.04%	46.96%	
aus	Original	86.47%	82.98%	86.66%	84.93%	85.57%	85.32%			7.68%
	Deletion	93.44%	92.45%	94.15%	93.04%	93.16%	93.25%	7.93%	54.02%	
	Reversal	93.82%	93.17%	94.79%	94.08%	93.45%	93.86%	8.54%	58.17%	
ion	Original	91.10%	88.78%	85.08%	88.05%	91.58%	88.92%			1.99%
	Deletion	85.35%	89.42%	85.04%	89.26%	93.73%	88.56%	-0.36%	-3.25%	
	Reversal	84.76%	88.42%	84.59%	89.05%	94.22%	88.21%	-0.71%	-6.41%	
bcw	Original	95.28%	94.39%	94.86%	92.78%	94.44%	94.35%			1.56%
	Deletion	95.56%	95.92%	95.72%	92.66%	96.02%	95.18%	0.83%	14.69%	
	Reversal	95.40%	95.98%	95.80%	93.55%	96.35%	95.42%	1.07%	18.94%	
vot	Original	97.05%	94.42%	96.49%	96.61%	97.14%	96.34%			2.59%
	Deletion	99.58%	99.16%	99.58%	99.58%	99.60%	99.50%	3.16%	86.34%	
	Reversal	99.58%	99.17%	99.58%	99.58%	99.59%	99.50%	3.16%	86.34%	
						Average	**87.05%**	**5.65%**	**34.33%**	**6.43%**
							87.81%	**6.41%**	**37.44%**	

Now let us analyze the results for deletion/reversal in case when the suspicious set is the union of two suspicious sets obtained by the *CBS* and *Best SER* techniques. The results are presented in Table 25.

Table 25

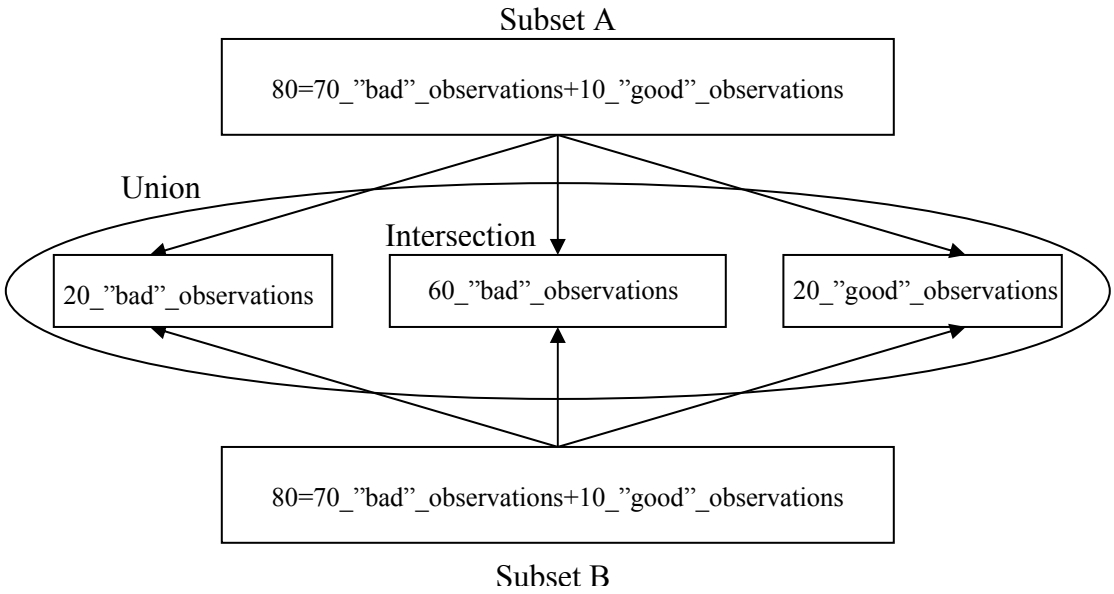
RESULTS FOR DELETION AND REVERSAL ON THE UNION

Dataset		SMO	MP	SL	C4.5	LAD	Average	Average increase in accuracy	Average error rate reduction	Size of suspicious set
bld	Original	50.03%	67.63%	66.24%	63.65%	69.29%	63.37%			36.52%
	Deletion	52.98%	88.54%	80.90%	99.02%	84.51%	81.19%	17.82%	48.65%	
	Reversal	65.58%	86.67%	68.76%	93.82%	82.35%	79.43%	16.07%	43.87%	
ger	Original	69.39%	65.50%	68.56%	66.18%	72.21%	68.37%			29.60%
	Deletion	95.16%	96.38%	94.93%	99.41%	97.33%	96.64%	28.28%	89.41%	
	Reversal	91.25%	92.10%	90.79%	97.47%	94.55%	93.23%	24.86%	78.60%	
pid	Original	72.31%	73.81%	72.07%	76.13%	74.60%	73.78%			18.11%
	Deletion	85.23%	88.65%	88.44%	94.55%	90.45%	89.46%	15.68%	59.80%	
	Reversal	86.54%	86.96%	89.26%	95.75%	90.14%	89.73%	15.95%	60.83%	
hea	Original	83.05%	78.70%	82.48%	77.16%	82.35%	80.75%			17.51%
	Deletion	98.47%	97.48%	97.80%	97.08%	93.99%	96.96%	16.22%	84.26%	
	Reversal	96.23%	93.91%	95.17%	91.86%	92.73%	93.98%	13.23%	68.73%	
aus	Original	86.47%	82.98%	86.66%	84.93%	85.57%	85.32%			12.32%
	Deletion	97.08%	97.42%	97.46%	97.66%	97.04%	97.33%	12.01%	81.81%	
	Reversal	96.73%	97.49%	97.59%	97.49%	96.73%	97.21%	11.89%	80.99%	
ion	Original	91.10%	88.78%	85.08%	88.05%	91.58%	88.92%			7.69%
	Deletion	89.09%	90.69%	89.96%	92.99%	96.24%	91.79%	2.88%	25.99%	
	Reversal	87.63%	89.13%	89.03%	94.10%	93.94%	90.77%	1.85%	16.70%	
bcw	Original	95.28%	94.39%	94.86%	92.78%	94.44%	94.35%			3.79%
	Deletion	98.78%	98.69%	97.63%	94.67%	97.49%	97.45%	3.10%	54.87%	
	Reversal	97.56%	97.75%	96.77%	95.06%	96.85%	96.80%	2.45%	43.36%	
vot	Original	97.05%	94.42%	96.49%	96.61%	97.14%	96.34%			3.02%
	Deletion	99.58%	99.56%	99.58%	99.58%	99.61%	99.58%	3.24%	88.52%	
	Reversal	99.04%	99.87%	99.17%	98.59%	99.20%	99.18%	2.83%	77.32%	
						Average	**93.80%**	**12.40%**	**66.66%**	**16.07%**
							92.54%	**11.14%**	**58.80%**	

Table 25 shows that the average size of the suspicious sets is 16.07%; the average increases in accuracy for deletion and for reversal are 12.40% and 11.14%, respectively; the average error rate reduction is 66.66% for deletion and 58.80% for reversal. The average results for reversal are less than that for deletion. This means that *too many observations were included in the suspicious subset; the percent of correctly classified*

observations in this set became too high. To better understand this phenomenon, let us consider an example shown in the figure below.

Figure 4. **Example: union of two suspicious subsets**



Assume we have two suspicious subsets A (e.g., obtained by using CBSes) and B (e.g, obtained by using *Best SER*). Each of the subsets includes 80 suspicious observations. We do not claim that each observation which is included in the subsets is "bad", i.e., it has wrong class. Let us suppose that in each of the subsets 70 of 80 observations are really "bad", and 10 of 80 observations (1/8 = 12.5%) are included in the subset by error, i.e., they are correctly classified in the original dataset. It was shown above that subsets of the suspicious observations obtained by using the CBS and Best *SER* techniques have a large intersection. Let us suppose that the intersection of A and B includes 60 observations. These observations, as we noted before, are more suspicious than others, so we can assume that all these observation are really "bad". It is clear that the union of

these subsets consists of 100 observations. These 100 observations include 80 "bad" observations and 20 "good" ones (see Figure 4). It implies that the percent of correctly classified observation in the newly created suspicious subset is 20/100 = 20%. This number is 7.5% higher than the number of the observations that were included in the sets A and B. So, the accuracy after reversal decreases, since we reverse the class of too many correctly classified observations.

Table 26 summarizes the average results from the tables above.

Table 26

		Suspicious set was obtained by using			
		CBS technique (p=p*)	Best SER technique (α= α*)	Intersection	Union
Average increase in accuracy	Deletion	11.13%	7.71%	5.65%	12.40%
	Reversal	11.60%	7.85%	6.41%	11.14%
Average error rate reduction	Deletion	59.15%	47.43%	34.33%	66.66%
	Reversal	59.92%	47.02%	37.44%	58.80%
Average size of suspicious set		13.66%	8.83%	6.43%	16.07%

These tables show that the best average increase in accuracy (best average error rate reduction) for deletion/reversal was obtained by the CBS technique and by the union; the size of the suspicious set obtained by the union is 2.41% more than that obtained by the CBS technique.

Conclusion 10. *The average increase in accuracy (the average error rate reduction) obtained on the intersection of two suspicious subsets is less than the average increase in accuracy (the average error rate reduction) on each of the subsets; the average increase*

in accuracy (the average error rate reduction) obtained on the union of two suspicious sets is comparable with the one obtained by the CBS technique, but the suspicious sets obtained by the union may contain many observations with the correct class.

3.3. Attribute Selection

As we mentioned before, attribute selection is the process of identifying and removing as many of the irrelevant and redundant attributes as possible. Alternatively, we want to find minimum sets of attributes that provide as much information for determining the class of the observations in the dataset as the original set of attributes. We shall refer to such subsets of attributes as *informative* subsets. In this section, we apply the CBS technique to the problem of identifying informative subsets and compare this approach with two other methods, C*FS* and *Consistency*, which are standard procedures of the WEKA package [96].

3.3.1. Attribute Selection Using Composite Boolean Separators

Let us repeat that we work with binary data to obtain CBSes. If the dataset is not binary, then the *Binarization* procedure (see [17]) is applied. We call the variables in the original dataset before binarization the *original variables*, and the variables obtained after this procedure the *original binary variables*. First, our attribute selection technique finds an informative subset of the *original binary variables*. Then using thus identified binary attributes and their relationship with the original ones (see Appendix) we identify the respective informative subset of the *original variables*.

In the attempt to reveal informative subsets of variables by utilizing the obtained CBSes, we consider two different approaches. The ***first approach*** (to be called *All_CBSe*s) consists of the following steps:

- for each CBS, find the subset of all *original binary variables* which this CBS depends on;
- take the union of all the subsets found in the previous step.

To illustrate this approach, let us return to the example in Sections 2.2 and 2.3. In this example, variables x_1, x_2, x_3, and x_4 are *original binary variables*; f_1, f_2, f_3, and f_4 are obtained CBSes. It can be seen that x_1, x_2 and x_3 are included in the formulas for the CBSes, so these variables form the set we are looking for. It is *not necessary* that the formula of each CBS should include each variable from the constructed set of variables, but it is *necessary* that each variable from this set should be in the formula for at least one separator.

Example

Obs.	x_1	x_2	x_3	x_4	$f_1 = \bar{x}_2 (x_1 \vee x_3)$	$f_2 = (x_1\ \bar{x}_2) \vee (x_1 \oplus x_3)$	$f_3 = (x_1\ \bar{x}_2) \vee (\bar{x}_2\ x_3)$	$f_4 = (x_1 \oplus x_3) \vee (\bar{x}_2\ x_3)$	class
A	0	1	0	0	0	0	0	0	0
B	1	1	1	0	0	0	0	0	0
C	0	0	0	1	0	0	0	0	0
D	1	0	1	0	1	1	1	1	1
E	1	0	0	0	1	1	1	1	1
F	0	0	1	1	1	1	1	1	1
CP	4/6	1/6	4/6	1/2	1	1	1	1	

The ***second approach*** (to be called *One_CBS*) uses one CBS with the highest CP. In this case, the informative set of attributes consists of those *original binary variables* which this CBS depends on. If there is a tie, i.e., there are several CBSes with the same highest *CP*, then we have several informative subsets of attributes. In this case, only additional experiments (for example, cross-validation) can show which subset is better. In the above example all CBSes have the highest *CP*. Since all these separators depend on the same *original binary variables*, only one subset $\{ x_1, x_2, x_3 \}$ is defined as an informative subset of attributes in the example.

In the next four tables, we present the results of application of the above two approaches to the eight datasets listed in Table 1. Each entry in the tables is the average result of twenty 10-folding cross-validation experiments. First, we present the results for the *original binary variables*.

From Table 27 and Table 28 we can conclude that the subsets of attributes obtained by using *One_CBS* on average are better than the ones obtained by *All_CBSe*s. First of all, these subsets contain fewer variables (on average 7 versus 9). Second of all, on average the *One_CBS* approach leads to slightly better classification results. It also can be seen that the average accuracy obtained on the original binary datasets is very close to that obtained on the informative subsets. The accuracy goes down for four datasets and it goes up also for four datasets. The highest loss of accuracy is for the **bld** dataset. On average the number of the original binary variables is 35. This number decreases to nine variables in case of the first approach and to seven variables in case of the second approach.

Table 27

RESULTS FOR ATTRIBUTE SELECTION WITH ALL_CBSES (ORIGINAL BINARY VARIABLES)

Dataset	# of original binary variables #of variables in informative subset	Average accuracy obtained by					Average accuracy of 5 methods	Difference between average accuracy of informative subset and original binary dataset
		SMO	MP	SL	C4.5	LAD		
bld	29	76.89%	72.51%	75.50%	70.44%	69.29%	72.93%	
	8	69.05%	67.34%	69.15%	68.85%	68.51%	68.58%	-4.35%
ger	57	68.22%	65.63%	68.31%	64.58%	72.21%	67.79%	
	12	61.72%	67.60%	65.20%	66.47%	72.00%	66.60%	-1.19%
pid	23	75.56%	73.20%	75.54%	76.49%	74.60%	75.08%	
	9	70.22%	75.12%	77.17%	75.89%	74.17%	74.51%	-0.57%
hea	17	83.77%	80.18%	83.21%	81.49%	82.35%	82.20%	
	8	83.26%	80.74%	83.63%	82.10%	84.51%	82.85%	0.65%
aus	45	86.12%	83.53%	85.95%	84.65%	85.57%	85.16%	
	9	86.17%	86.09%	86.59%	85.28%	86.14%	86.05%	0.89%
ion	71	87.69%	86.59%	89.02%	88.13%	91.58%	88.60%	
	14	89.20%	88.02%	89.29%	88.69%	83.88%	87.82%	-0.79%
bcw	20	95.19%	94.51%	94.72%	93.81%	94.44%	94.53%	
	11	95.39%	94.80%	95.46%	94.94%	93.71%	94.86%	0.32%
vot	16	97.05%	94.42%	96.49%	96.61%	97.14%	96.34%	
	1	97.12%	97.12%	97.12%	97.12%	97.12%	97.12%	0.77%
						Average	82.83%	
						Average	82.30%	-0.53%

Table 28

RESULTS FOR ATTRIBUTE SELECTION WITH ONE_CBS (ORIGINAL BINARY VARIABLES)

Dataset	# of original binary variables / #of variables in informative subset	Average accuracy obtained by					Average accuracy of 5 methods	Difference between average accuracy of informative subset and original binary dataset
		SMO	MP	SL	C4.5	LAD		
bld	29	76.89%	72.51%	75.50%	70.44%	69.29%	72.93%	
	7	66.25%	70.17%	69.44%	70.17%	68.87%	68.98%	-3.95%
ger	57	68.22%	65.63%	68.31%	64.58%	72.21%	67.79%	
	11	61.93%	65.22%	67.63%	68.67%	70.97%	66.88%	-0.90%
pid	23	75.56%	73.20%	75.54%	76.49%	74.60%	75.08%	
	9	70.22%	75.12%	77.17%	75.89%	74.17%	74.51%	-0.57%
hea	17	83.77%	80.18%	83.21%	81.49%	82.35%	82.20%	
	7	83.89%	84.08%	82.86%	82.71%	84.31%	83.57%	1.37%
aus	45	86.12%	83.53%	85.95%	84.65%	85.57%	85.16%	
	9	86.17%	86.09%	86.59%	85.28%	86.14%	86.05%	0.89%
ion	71	87.69%	86.59%	89.02%	88.13%	91.58%	88.60%	
	9	88.78%	89.59%	89.83%	88.37%	81.65%	87.64%	-0.96%
bcw	20	95.19%	94.51%	94.72%	93.81%	94.44%	94.53%	
	6	95.09%	95.12%	95.38%	95.30%	93.85%	94.95%	0.41%
vot	16	97.05%	94.42%	96.49%	96.61%	97.14%	96.34%	
	1	97.12%	97.12%	97.12%	97.12%	97.12%	97.12%	0.77%
						Average	82.83%	
						Average	82.46%	-0.37%

The results for the binary datasets are good, but we are interested in obtaining an informative subset of the *original variables*. To this end, we restore the original variables from binary ones. Let us illustrate the restoration procedure with an example.

Example. Consider the subset of attributes for **bld** obtained by using *One_CBS*. This subset consists of seven *original binary variables* (9, 11, 12, 19, 22, 23, 25). Table A in the Appendix shows that binary variables 9, 11, and 12 are obtained by binarization of the original variable "sgpt", the binary variable 19 is the result of binarization of the original variable "sgot", and binarization of the original variable "gammagt" gives the binary variables 22, 23 and 25. Therefore, the three original variables "sgpt", "sgot" and "gammagt" form the informative subset chosen by the attribute selection procedure.

Table 29 and Table 30 present the results obtained on the informative subsets of the *original variables*. These tables show that the results obtained on the original variables are similar to those obtained on the *original binary* datasets. The average number of variables in the original datasets is 15. This number reduces to 7 if *All_CBSe*s is used and to 6 in case if only *One_CBS* is used. The average accuracy on the original datasets is very close to the average accuracy obtained on the informative subsets and is better only by 0.43% for the first approach and by 0.30% for the second one. The informative subsets win in accuracy for four out of eight datasets and lose for the remaining four. On the basis of the above discussion the following conclusion can be made.

Table 29

RESULTS FOR ATTRIBUTE SELECTION WITH ALL_CBSES (ORIGINAL VARIABLES)

Dataset	# of original variables	Average accuracy obtained by					Average accuracy of 5 methods	Difference between average accuracy of informative subset and original dataset
	# of variables in informative subset	SMO	MP	SL	C4.5	LAD		
bld	6	50.03%	67.63%	66.24%	63.65%	69.29%	63.37%	-0.74%
	4	50.00%	68.28%	64.85%	64.27%	65.75%	62.63%	
ger	24	69.39%	65.50%	68.56%	66.18%	72.21%	68.37%	-3.06%
	8	61.80%	64.72%	64.71%	65.59%	69.71%	65.31%	
pid	8	72.31%	73.81%	72.07%	76.13%	74.60%	73.78%	0.49%
	6	71.90%	74.74%	72.79%	75.93%	76.02%	74.28%	
hea	13	83.05%	78.70%	82.48%	77.16%	82.35%	80.75%	0.29%
	7	83.27%	78.86%	82.52%	78.49%	82.04%	81.04%	
aus	14	86.47%	82.98%	86.66%	84.93%	85.57%	85.32%	0.81%
	8	86.21%	85.85%	87.05%	85.26%	86.31%	86.13%	
ion	33	91.10%	88.78%	85.08%	88.05%	91.58%	88.92%	-1.90%
	12	84.05%	89.92%	83.48%	88.72%	88.94%	87.02%	
bcw	9	95.28%	94.39%	94.86%	92.78%	94.44%	94.35%	-0.08%
	7	95.23%	94.53%	94.76%	93.18%	93.66%	94.27%	
vot	16	97.05%	94.42%	96.49%	96.61%	97.14%	96.34%	0.77%
	1	97.12%	97.12%	97.12%	97.12%	97.10%	97.11%	
Average	15						81.40%	
	7						80.97%	-0.43%

Table 30

RESULTS FOR ATTRIBUTE SELECTION WITH ONE_CBS (ORIGINAL VARIABLES)

Dataset	# of original variables / # of variables in informative subset	Average accuracy obtained by: SMO	MP	SL	C4.5	LAD	Average accuracy of 5 methods	Difference between average accuracy of informative subset and original dataset
bld	6	50.03%	67.63%	66.24%	63.65%	69.29%	63.37%	
	3	50.00%	66.81%	64.81%	61.73%	64.00%	61.47%	-1.90%
ger	24	69.39%	65.50%	68.56%	66.18%	72.21%	68.37%	
	7	63.66%	68.22%	65.98%	66.43%	71.29%	67.12%	-1.25%
pid	8	72.31%	73.81%	72.07%	76.13%	74.60%	73.78%	
	6	71.90%	74.74%	72.79%	75.93%	76.02%	74.28%	0.49%
hea	13	83.05%	78.70%	82.48%	77.16%	82.35%	80.75%	
	6	83.10%	79.47%	82.33%	80.58%	82.56%	81.61%	0.86%
aus	14	86.47%	82.98%	86.66%	84.93%	85.57%	85.32%	
	8	86.21%	85.85%	87.05%	85.26%	86.31%	86.13%	0.81%
ion	33	91.10%	88.78%	85.08%	88.05%	91.58%	88.92%	
	8	83.98%	90.38%	83.89%	89.64%	87.58%	87.09%	-1.83%
bcw	9	95.28%	94.39%	94.86%	92.78%	94.44%	94.35%	
	5	94.62%	94.13%	94.32%	93.58%	93.31%	93.99%	-0.36%
vot	16	97.05%	94.42%	96.49%	96.61%	97.14%	96.34%	
	1	97.12%	97.12%	97.12%	97.12%	97.10%	97.11%	0.77%
Average	15						81.40%	
	6						81.10%	-0.30%

Conclusion 11. *The informative subsets of variables chosen by using All_CBSes and by using One_CBS have fairly small size, since their number of variables is about half of the number of the original ones. The second approach produces subsets of slightly smaller size. Both methods also perform well in terms of classification accuracy regardless of classification methods applied to the chosen subsets. The difference between the average accuracy on the informative subsets and on the original data is small. It is difficult to compare these two methods with respect to accuracy, since for some datasets the first method is better (for example, for* ***bld****) and for other datasets the second one is better (for example, for* ***ger****).*

3.3.2. Comparison of Attribute Selection Results Obtained with CBS and with WEKA Approaches

In this section, we report attribute selection results obtained with two WEKA methods (*CFS* and *Consistency*) and compare them with the results obtained in the previous section.

CFS [39], [40] (Correlation-based Feature Selection) is an attribute selection technique which builds an informative subset of attributes so that any two attributes in the subset have a low correlation with each other, while each of them has a high correlation with the class.

Consistency [40] (Consistency-Based Subset Evaluation) builds a combination of the attributes whose values divide the data into subsets containing a strong single class

majority. This method looks for the smallest subset with consistency equal to that of the full set of attributes.

In Table 31 we present the percentage of correct classifications averaged over twenty 10-foding experiments obtained by the two methods described above.

Table 31

RESULTS OF ATTRIBUTE SELECTION OBTAINED WITH TWO WEKA METHODS (CFS AND CONSISTENCY)

Dataset	# of variables in original dataset	# of variables in informative set obtained by WEKA using	Average accuracy obtained by					Average accuracy of 5 methods
		CFS Consistency	SMO	MP	SL	C4.5	LAD	
bld	6	1 1	50.00% 50.00%	56.94% 56.94%	50.21% 50.21%	60.50% 60.50%	52.93% 52.93%	54.12% 54.12%
ger	24	4 19	60.48% 68.01%	66.20% 65.08%	65.01% 68.15%	66.28% 65.32%	69.76% 72.54%	65.55% 67.82%
pid	8	2 7	73.24% 71.03%	72.40% 70.14%	73.12% 72.11%	77.84% 71.76%	73.18% 74.52%	73.95% 71.91%
hea	13	5 12	83.74% 83.44%	81.52% 78.50%	82.69% 83.01%	82.24% 77.36%	83.56% 81.94%	82.75% 80.85%
aus	14	1 14	86.21% 86.47%	86.21% 82.98%	86.21% 86.66%	86.21% 84.93%	86.21% 85.57%	86.21% 85.32%
ion	33	9 10	83.92% 84.51%	90.71% 89.50%	83.67% 83.86%	89.53% 89.49%	88.67% 88.94%	87.30% 87.26%
bcw	9	8 4	93.96% 94.85%	93.51% 93.67%	93.18% 94.18%	91.54% 91.73%	93.88% 94.25%	93.21% 93.73%
vot	16	1 15	97.12% 97.12%	97.12% 94.23%	97.12% 96.57%	97.12% 96.61%	97.10% 97.01%	97.11% 96.31%
	Average	4 10					Average	80.03% 79.67%

It can be seen from the above table that the average size of the chosen subsets of variables is four for *CFS* and 10 for *Consistency*. Thus, we conclude that the size of the

subsets obtained by *Consistency* is maximum among the two CBS methods and the two WEKA methods. We also see that the average accuracy obtained by *Consistency* is minimum among the four mentioned methods.

In Table 32, we compare the two CBS based methods with the two WEKA approaches. To make the comparison we performed the paired two sample for means two-tail *t* test using the confidence level of 95%. We use the sign **+** in favor of the CBS based methods and the sign **-** in favor of the WEKA approaches. For example, the sign **+** located at the intersection of the *CFS* row and the *One_CBS* column for the **ger** dataset indicates that the *One_CBS* method statistically performs better than *CFS* on the given dataset. The sign **-** located at the intersection of the *CFS* row and the *One_CBS* column for the **hea** dataset shows that *One_CBS* performs statistically worse than *CFS*.

Table 32

COMPARISON OF CBS BASED METHODS WITH CFS AND CONSISTENCY

Dataset	Method	All_CBSes	One_CBS
bld	CFS	+	
	Consistency	+	
ger	CFS		+
	Consistency		
pid	CFS		
	Consistency	+	+
hea	CFS		-
	Consistency		
aus	CFS		
	Consistency		
ion	CFS		
	Consistency		
bcw	CFS	+	
	Consistency		
vot	CFS		
	Consistency		

The above table shows that the *All_CBSes* method has four wins; *One_CBS* has two wins and one loss. *CFS* has one win and three losses, and *Consistency* has three losses.

Conclusion 12. *None of the four analyzed methods provides the best results uniformly for all datasets. However, in most cases the CBS based approaches show better results than the CFS and Consistency methods.*

4. ANALYSIS OF TWO REAL-LIFE MEDICAL DATASETS

This chapter is devoted to the analysis of two real-life datasets: computed tomography data and breast cancer gene expression microarray data. In Sections 4.1, 4.2 we report the results obtained for these datasets with LAD, and in Section 4.3 we compare them with the results obtained by the CBS technique.

4.1. Logical Analysis of Computed Tomography Data to Differentiate Entities of Idiopathic Interstitial Pneumonias[2]

The *Idiopathic Interstitial Pneumonias* (IIPs) are a heterogeneous group of nonneoplastic disorders resulting from damage to the lung parenchyma by varying patterns of inflammation and fibrosis. A new classification of IIPs was established in 2001 by an International Consensus Statement defining the clinical manifestations, pathology and radiological features of patients with IIPs [12]. Various forms of IIP differ both in their prognoses and their therapies, but are not easily distinguishable using clinical, biological and radiological data, and therefore frequently requiring pulmonary biopsies to establish the diagnosis. The aim of this study is to analyze computed tomography (CT) data by techniques of biomedical informatics to distinguish between three types of IIPs:

- *Idiopathic Pulmonary Fibrosis* (IPF)
- *Non Specific Interstitial Pneumonia* (NSIP)

[2] The results presented in this section are based on joint work with M.W. Brauner, N. Brauner, P.L. Hammer, and D. Valeyre, published in [20].

- *Desquamative Interstitial Pneumonia* (DIP)

4.1.1. Patients and Methods

This is a study of the CT scans in patients with IIPs referred to the Department of Respiratory Medecine, Avicenne Hospital, Bobigny, France, for medical advice on diagnosis and therapy. The diagnosis was established on clinical, radiographic and pathologic data. The 56 patients included 34 IPFs, 15 NSIPs, and 7 DIPs.

We reviewed the CT examination of the chest from these patients. CT scans were evaluated for the presence of signs and a score was established for the two main lesions, ground-glass attenuation and reticulation. Pulmonary disease severity on thin section CT scans was scored semiquantitatively in upper, middle and lower lung zones. The six areas of the lung were defined as follows: the upper zones are above the level of the carina; the middle zones, between the level of the carina and the level of the inferior pulmonary veins; and the lower zones, under the level of the inferior pulmonary veins. The profusion of opacities was recorded separately in the six areas of the lung to yield a total score of parenchymal opacities. The severity was scored in each area according to four basic categories : 0 = normal, 1 = slight, 2 = moderate, 3 = advanced (total : 0-18).

The data consisting of the binary attributes 1, 2, …, 10, and the numerical attributes 11, 12, and 13 are listed bellow:

1. IIT	intralobular interstitial thickening
2. HC	honeycombing
3. TB	traction bronchiectasis
4. GG1	ground-glass attenuation
5. BRVX	peri-bronchovascular thickening
6. PL	polygonal lines
7. HPL	hilo-peripheral lines
8. SL	septal lines
9. AC	airspace consolidation
10. N	nodules
11. GG2	ground-glass attenuation score
12. RET[3]	reticulation score
13. GG2/RET	ground-glass attenuation/reticulation score

In this section, we analyze this dataset using the *Logical Analysis of Data*. Among previous studies dealing with applications of LAD to medical problems we mention ([3], [6], [63]).

The choice of LAD for analyzing the IIP data is due on the one hand to its proven possibility to provide highly accurate classifications, and on the other hand to the usefulness of LAD patterns in analyzing the significance and nature of attributes.

[3] RET is a generic term which get together the 3 main fibrotic lesions : ITT, HC and TB

The conclusions of LAD have been confirmed by other machine-learning / data-mining methods (SMO, MP, SL, and C4.5) described in the introduction. An additional result of the study was the identification by LAD of two outliers, which turned out to have complete medical explanation.

4.1.2. Outliers

We have constructed three different LAD models to distinguish between IPF, NSIP or DIP patients:

- model I to distinguish IPF patients (considered to be the positive observations in this model) from non-IPF patients (negative observations);
- model II to distinguish NSIP patients (positive in this model) from non-NSIP patients (negative observations);
- model III to distinguish DIP patients (positive in this model) from non-DIP patients (negative observations).

These models use only pure patterns, their degrees are at most 4, and their prevalences range between 40% and 85.7%.

4.1.2.1. Two Suspicious Observations

The classification given by the three LAD models for the 56 observations in the dataset is shown in Table 33. It can be seen that all the 56 classifications are correct, but only 54 of

them are precise. In fact the classifications of the observations s003 and s046 are vague. Since observation s003 turns out to be classified as being either a DIP or an NSIP patient, we have built an additional model to distinguish between these two classes. It turns out that the model contains only one pattern covering observation s003. This pattern shows (correctly) that s003 is a DIP patient, however it does not cover any other observation, i.e., its prevalence is so low that it cannot be considered reliable. A very similar argument concerning the observation s046 shows that in a model distinguishing IPF/NSIP cases, it is classified as being an NSIP case, however this classification is based only on extremely weak patterns, whose reliability is low. The facts signaled above, raise suspicions about the specific nature of these two observations, and raise the question of whether they should be included at all in the dataset.

4.1.2.2. Medical Confirmation

In view of the suspicions related to these two observations, the medical records of these two patients have been re-examined. It was found that patient s003 was exposed to asbestos, and therefore its classification as DIP is uncertain. Asbestosis may be responsible for a pathologic aspect similar to that of IPF, but very different from DIP. It is also possible that the pathologic result on the biopsy of a very small area of the lung was wrong. Also, it was found that the data of patient s046 are highly atypical in all the features (age, clinical data and lung pathology). Based on the clinical, radiographic and pathologic data, this patient does not seem to belong to any of the three classes in the initial classification before CT analysis, and it was suggested that in view of these reasons, (s)he should be considered non-classable and removed from the dataset.

Table 33

Observations	Given Classification	Classification by LAD Models			
		IPF/non-IPF	NSIP/non-NSIP	DIP/non-DIP	Conclusion
s001	DIP	0	?	1	DIP
s002	DIP	0	0	1	DIP
s003	DIP	0	?	?	NSIP or DIP
s004	DIP	0	0	1	DIP
s005	DIP	0	?	1	DIP
s006	DIP	0	?	1	DIP
s007	DIP	0	0	1	DIP
s008	IPF	1	0	0	IPF
s009	IPF	1	?	0	IPF
s010	IPF	1	?	0	IPF
s011	IPF	1	0	0	IPF
s012	IPF	1	0	0	IPF
s013	IPF	1	0	0	IPF
s014	IPF	1	0	0	IPF
s015	IPF	1	0	0	IPF
s016	IPF	1	?	0	IPF
s017	IPF	1	?	0	IPF
s018	IPF	1	0	0	IPF
s019	IPF	1	0	0	IPF
s020	IPF	1	0	0	IPF
s021	IPF	1	?	0	IPF
s022	IPF	1	0	0	IPF
s023	IPF	1	0	0	IPF
s024	IPF	1	0	0	IPF
s025	IPF	1	0	0	IPF
s026	IPF	1	0	0	IPF
s027	IPF	1	0	0	IPF
s028	IPF	1	0	0	IPF
s029	IPF	1	0	0	IPF
s030	IPF	1	0	0	IPF
s031	IPF	1	0	0	IPF
s032	IPF	1	0	0	IPF
s033	IPF	1	0	0	IPF
s034	IPF	1	0	0	IPF
s035	IPF	1	0	0	IPF
s036	IPF	1	0	0	IPF
s037	IPF	1	0	0	IPF
s038	IPF	1	0	0	IPF
s039	IPF	1	0	0	IPF
s040	IPF	1	0	0	IPF
s041	IPF	1	0	0	IPF
s042	NSIP	0	1	0	NSIP
s043	NSIP	0	1	0	NSIP
s044	NSIP	0	1	0	NSIP
s045	NSIP	0	1	0	NSIP
s046	NSIP	?	?	0	IPF or NSIP
s047	NSIP	0	1	0	NSIP
s048	NSIP	0	1	?	NSIP
s049	NSIP	0	1	?	NSIP
s050	NSIP	0	1	0	NSIP
s051	NSIP	0	1	0	NSIP
s052	NSIP	0	1	0	NSIP
s053	NSIP	0	1	0	NSIP
s054	NSIP	0	1	0	NSIP
s055	NSIP	0	1	0	NSIP
s056	NSIP	0	1	0	NSIP

4.1.2.3. Improving Classification Accuracy by Removing Outliers

The medical confirmation of the suspicions raised by the inability of the LAD models to classify the two unusual observations, have led us to check the ways in which the accuracy of various classification methods changes when these two observations are removed from the dataset. In order to evaluate these changes, we have applied four classification methods taken from the WEKA package [96], separately to the original dataset of 56 observations, and to the dataset of 54 observations obtained by removing the two suspicious ones.

Twenty 3-folding experiments were carried out for each of the three classification problems (IPF/non-IPF, NSIP/non-NSIP, DIP/non-DIP). In each of the experiments the dataset was randomly partitioned into three approximately equal parts, two of which were used as the training set, and the third one as the testing set. By rotating the subset taken in the role of the test set, in fact each experiment consisted of 3 tests, i.e., a total of 60 experiments were carried out for each of the three classification problems. The average accuracy of these 1440 experiments (i.e., four methods applied 60 times to original and reduced datasets of three problems) measured on the test sets is shown in Table 34.

It can be seen from Table 34 that by removing the two outliers, the accuracy of every single classification method was improved for each of the three models. It also can be seen that the removing only two these observations cuts the error rate from 7.5% to 20.3% for these problems.

Table 34

CLASSIFICATION ACCURACIES BEFORE/AFTER ELIMINATION OF OUTLIERS

	Dataset	SMO	MP	SL	C4.5	Average	Average change in accuracy	Average error rate reduction
NSIP/non-NSIP	Original	66.50%	65.11%	62.53%	62.14%	64.07%	+7.32%	20.37%
	Reduced	71.96%	74.09%	71.40%	68.12%	71.39%		
IPF/non-IPF	Original	79.60%	79.44%	80.66%	81.27%	80.24%	+1.49%	7.54%
	Reduced	79.62%	81.24%	82.15%	83.90%	81.73%		
DIP/non-DIP	Original	60.88%	64.38%	62.30%	71.18%	64.69%	+5.28%	14.92%
	Reduced	63.96%	73.02%	62.66%	80.21%	69.96%		

In conclusion, the suspicions generated by the weakness of the coverage with patterns of two of the observations, lead to the identification of these two patients as outliers, and eventually to medical explanations of the inappropriateness of maintaining them in the dataset. The "cleaned" dataset obtained by eliminating these two outliers was shown to allow a substantial improvement in the accuracy of all the tested classification methods.

4.1.3. Support Sets

4.1.3.1. Set Covering Formulation

Although the dataset involves 13 variables, some of them may be redundant. Following the terminology of LAD ([18], [27], [41]) we shall call an irredundant set of *variables* or *attributes* or *features* a *support set* of the dataset, if projecting on this subset the 13 dimensional vector representing the patients, there will be no overlap between the three different types of IIPs.

The determination of a minimum size support set was formulated as a set covering problem. The basic idea of the set covering formulation of this problem consists in the simple observation that a subset S is a support set if and only if the projections on S of the positive and the negative observations in the dataset are disjoint.

In order to illustrate this reduction we shall identify a subset of the variables in the dataset which are capable of distinguishing IPF observations from non-IPF observations. We shall assume that the three numerical variables x_{11}, x_{12}, x_{13} have been "*binarized*", i.e., each of them had been replaced by one or several 0-1 variables. The binarized variables are associated to *cut-points*. For instance, there are two cut-points (5.5 and 6.5) associated to the numerical variable x_{11}, and the corresponding binary variables $x_{11}^{5.5}$ and $x_{11}^{6.5}$ are then defined in the following way:

$$x_{11}^{5.5}=1 \text{ if } x_{11}\geq 5.5, \text{ and } x_{11}^{5.5}=0 \text{ if } x_{11}<5.5,$$

$$x_{11}^{6.5}=1 \text{ if } x_{11}\geq 6.5, \text{ and } x_{11}^{6.5}=0 \text{ if } x_{11}<6.5.$$

Similarly, two cut-points (7.5, 8.5) are introduced for x_{12}, along with two associated binary variables. The variable x_{13} is binarized using four 0-1 variables associated to the cut-points 0.5, 1, 1.05 and 1.2.

Using the original 10 binary variables along with the eight binarized variables (which replace the numerical variables x_{11}, x_{12}, x_{13}), we shall now represent the observations as 18 dimensional binary vectors ($x_1,\ldots,x_{10}, x_{11}^{5.5}, x_{11}^{6.5}, x_{12}^{7.5}, x_{12}^{8.5}, x_{13}^{0.5}, \ldots, x_{13}^{1.2}$). For example, the positive (i.e., IPF) observation s008 = (0, 1, 1, 1, 0, 1, 1, 0, 0, 0, 2, 9, 0.22) will become in this way the binary vector b008 = (0, 1, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0,

0). Similarly the negative (i.e., non-IPF) observation s006 = (0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 5, 2, 2.5) becomes the binary vector b006 = (0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1).

Clearly, the positive binarized observation b008 and the negative binarized observation b006 differ only in the following eight components: x_2, x_3, $x_{12}^{7.5}$, $x_{12}^{8.5}$, $x_{13}^{0.5}$, …, $x_{13}^{1.2}$. It follows that any support set S must include at least one of these variables, since otherwise the projections on S of the positive observation b008 and the negative observation b006 could not be distinguished. Therefore, if we denote by (s_1,…,s_{10}, $s_{11}^{5.5}$, $s_{11}^{6.5}$, $s_{12}^{7.5}$, $s_{12}^{8.5}$, $s_{13}^{0.5}$, …, $s_{13}^{1.2}$) the characteristic vector of S, one of the necessary conditions for S to be a support set is

$$s_2 + s_3 + s_{12}^{7.5} + s_{12}^{8.5} + s_{13}^{0.5} + \ldots + s_{13}^{1.2} \geq 1.$$

A similar inequality can be written for every pair consisting of a positive (IPF) and a negative (non-IPF) observation in the binarized dataset. The 34*22=748 pairs of positive-negative observations define the constraints of a set covering problem for finding a minimum size support set. Since our dataset consists of a rather limited number of observations, in order to increase the accuracy of the models to be built on the support sets obtained in this way, we have further strengthened the above set covering-type constraints, by replacing the 1 on their right-hand side, by 3 (the choice of 3 is based on empirical considerations, the basic idea being simply to sharpen the requirements of separating positive and negative observations).

Clearly the objective function of this set covering type problem is simply the sum

$$s_1 + \ldots + s_{10} + s_{11}^{5.5} + s_{11}^{6.5} + s_{12}^{7.5} + s_{12}^{8.5} + s_{13}^{0.5} + \ldots + s_{13}^{1.2}.$$

4.1.3.2. Three Support Sets

By solving this problem we found that the binary variables x_3, x_4, x_9, x_{10} are redundant, and that a support set (using the original binary and numerical variables) consists of the attributes 1, 2, 5, 6, 7, 8, 11, 12 and 13.

In a similar way we can see that a support set distinguishing DIP observations from non-DIP ones consists of the six original attributes: 1, 2, 3, 5, 12 and 13, while a support set distinguishing NSIP patients from non-NSIP ones consists of the eight original attributes: 1, 2, 5, 6, 7, 8, 11 and 12.

4.1.3.3. Accuracy of Classification on Support Sets

It is important to point out that the elimination of redundant variables does not reduce the accuracy of classification. In order to demonstrate the qualities of the support sets obtained for the IPF/non-IPF, DIP/non-DIP and NSIP/non-NSIP problems we have carried out twenty 3-folding classification experiments on these three problems using four different classification methods mentioned above. These experiments used first the original 13 variables, and after that the support sets of nine, six, and eight variables

respectively, obtained above for these three problems. The results of these experiments are presented in Table 35.

Table 35

CLASSIFICATION ACCURACIES ON ALL ORIGINAL VARIABLES AND ON SUPPORT SETS

	Support Set	SMO	MP	SL	C4.5	Average	Average change in accuracy
NSIP/non-NSIP	Original	71.96%	74.09%	71.40%	68.12%	71.39%	+1.34%
	Reduced	71.88%	78.96%	72.21%	67.88%	72.73%	
IPF/non-IPF	Original	79.62%	81.24%	82.15%	83.90%	81.73%	+0.44%
	Reduced	79.15%	83.66%	81.12%	84.74%	82.17%	
DIP/non-DIP	Original	63.96%	73.02%	62.66%	80.21%	69.96%	+3.07%
	Reduced	60.16%	74.27%	70.83%	86.88%	73.03%	

In conclusion we can see from Table 35 that the elimination of those features which were identified as redundant does not only maintain the accuracy of classification, but actually on average increases it in each of the three models.

4.1.4. Patterns and Models

Using the support sets developed in the previous section, we shall apply now the LAD methodology to this dataset for generating patterns and classification models. It turns out that in spite of the very small size of this dataset, some surprisingly strong patterns can be identified in it. For example, in the IPF/non-IPF model, 14 (i.e., 70%) of the 20 non-IPF patients satisfy the simple pattern "GG2/RET $\geq$ 1.2"; moreover none of the 34 IPF patients satisfy this condition. While this simple pattern involves a single variable, other

more complex patterns exist and are capable of explaining the IPF or non-IPF character of large groups of patients. For instance, the negative pattern

$$\text{“RET} \leq 8 \quad \text{and} \quad \text{GG2/RET} > 1\text{”}$$

is satisfied by 70% of the non-IPF patients, and by none of the IPF patients. As an example of a positive pattern, we mention

$$\text{“HC} = 1, \ \text{HPL} = 0 \quad \text{and} \quad \text{GG2/RET} \leq 1.2\text{”};$$

24 (i.e., 70.6%) of the 34 IPF patients satisfy all the three constraints of this pattern, and none of the non-IPF patients satisfy simultaneously these three conditions.

While the above patterns can distinguish large groups of patients having a certain type of IIP from those of other types of IIP, larger collections of patterns constructed by LAD can classify collectively the entire set of 54 observations in the dataset. We shall first illustrate the way the classification works by considering the problem of distinguishing IPF and non-IPF patients.

We present in Table 36 a model consisting of 20 positive and 20 negative patterns allowing the accurate classification of IPF/non-IPF patients. Note that the equality of the numbers of positive and negative patterns in this model is a simple coincidence.

Beside the IPF/non-IPF model, we have also constructed a model to distinguish the 14 NSIP patients from the 40 non-NSIP patients, and another model to distinguish the 6 DIP patients from the 48 non-DIP patients. The NSIP/non-NSIP model is built on the support set of eight attributes described in the previous section, and includes 16 positive and 4

negative patterns. The DIP/non-DIP model is built on the support set of six attributes described in the previous section, and includes 7 positive and 15 negative patterns.

Table 36

IPF/NON-IPF MODEL

Pattern	attr.1	attr.2	attr.5	attr.6	attr.7	attr8	attr.11	attr.12	attr.13	Pos Prevalence	Neg Prevalence
	IIT	HC	BRVX	PL	HPL	SL	GG2	RET	GG2/RET		
P1		1			0				≤1.2	70.6%	0
P2		1			0			≥8		47.1%	0
P3	1	1					≥4		≤1.2	47.1%	0
P4		1			0	0		≥6		47.1%	0
P5	1	1			0			≥6		47.1%	0
P6	1	1					≥4	≥8		41.2%	0
P7		1	0						≤0.5	41.2%	0
P8		1	0					≤8	≤1.2	41.2%	0
P9	1	1							>0.5, ≤1.2	38.2%	0
P10				1					≤1.2	32.4%	0
P11	1	1						≥8	>0.5	32.4%	0
P12	1	1						≥9		29.4%	0
P13		1	0				≤3			29.4%	0
P14		1					≥4	≤8	≤1.2	26.5%	0
P15					0			≥8	≤0.5	26.5%	0
P16			0	1				≥6		26.5%	0
P17	0							≤8	≤1.2	20.6%	0
P18						1		≥8		20.6%	0
P19				1			≤3			20.6%	0
P20				1	0					17.6%	0
N1								≤8	>1	0	70.0%
N2									>1.2	0	70.0%
N3		0				0	≥4			0	50.0%
N4								≤5		0	50.0%
N5		0				0			>0.5	0	50.0%
N6		0		0	0					0	45.0%
N7		0		0				≤7		0	45.0%
N8		0			0				>0.5	0	40.0%
N9		0			0		≥4			0	40.0%
N10					0				>1	0	40.0%
N11		0		0		0				0	40.0%
N12	0								>1	0	35.0%
N13	1	0					≥4	≤7		0	30.0%
N14					1	0		≤8	>0.5	0	30.0%
N15					1	0	≥4	≤8		0	30.0%
N16				0	1	0		≤8		0	20.0%
N17						1			>1	0	15.0%
N18	0				1	0	≥4			0	15.0%
N19			1		1	0		≤8		0	15.0%
N20	0				1	0			>0.5	0	15.0%

It is not surprising to find, in Table 36, honeycombing and a high score of reticulation in the majority of IPF patterns since it is known that IPF can be suspected with confidence in the 50% of cases with a bilateral, predominantly basal, predominantly subpleural reticular pattern, associated with honeycombing and/or traction bronchectasis. However the distribution of the lesions has not been tested in this study since NSIP and DIP have almost the same distribution. On the contrary, it is known that ground glass is the predominant finding in the majority of cases of NSIP and DIP explaining the high scores of ground glass and of ground glass on reticulations in 10 of the non-IPF patterns.

The combination of the three models allows the drawing of additional conclusions. For example, if the results of the three classifications are 0, 0 and ? respectively, and one knows that each patient is exactly of one type of IIP, one can conclude that the "?" in the classification of the third condition can be replaced by "1".

The results of the classification of the 54 patients given by the three models, along with the conclusions derived from the knowledge of all the three classifications are presented in Table 37. The accuracy of this classification is 100%.

It is usually said that the CT diagnosis of NSIP is difficult. In a recent study [48] experienced observers considered the CT pattern indistinguishable from IPF in 32% of cases. In another investigation the author assessed the value of CT in the diagnosis of 129 patients with histologically proven idiopathic interstitial pneumonias [58]. Two independent observers were able to make a correct first choice diagnosis in more than

Table 37

Observations	Given Classification	Classification by LAD Models			
		IPF/non-IPF	NSIP/non-NSIP	DIP/non-DIP	Conclusion
s001	DIP	0	?	1	DIP
s002	DIP	0	0	1	DIP
s004	DIP	0	0	1	DIP
s005	DIP	0	?	1	DIP
s006	DIP	0	?	1	DIP
s007	DIP	0	0	1	DIP
s008	IPF	1	0	0	IPF
s009	IPF	1	?	0	IPF
s010	IPF	1	?	0	IPF
s011	IPF	1	0	0	IPF
s012	IPF	1	0	0	IPF
s013	IPF	1	0	0	IPF
s014	IPF	1	0	0	IPF
s015	IPF	1	0	0	IPF
s016	IPF	1	?	0	IPF
s017	IPF	1	0	0	IPF
s018	IPF	1	0	0	IPF
s019	IPF	1	0	0	IPF
s020	IPF	1	0	0	IPF
s021	IPF	1	?	0	IPF
s022	IPF	1	0	0	IPF
s023	IPF	1	0	0	IPF
s024	IPF	1	0	0	IPF
s025	IPF	1	0	0	IPF
s026	IPF	1	0	0	IPF
s027	IPF	1	0	0	IPF
s028	IPF	1	0	0	IPF
s029	IPF	1	0	0	IPF
s030	IPF	1	0	0	IPF
s031	IPF	1	0	0	IPF
s032	IPF	1	0	0	IPF
s033	IPF	1	0	0	IPF
s034	IPF	1	0	0	IPF
s035	IPF	1	0	0	IPF
s036	IPF	1	0	0	IPF
s037	IPF	1	0	0	IPF
s038	IPF	1	0	0	IPF
s039	IPF	1	0	0	IPF
s040	IPF	1	0	0	IPF
s041	IPF	1	0	0	IPF
s042	NSIP	0	1	0	NSIP
s043	NSIP	0	1	?	NSIP
s044	NSIP	0	1	0	NSIP
s045	NSIP	0	1	0	NSIP
s047	NSIP	0	1	0	NSIP
s048	NSIP	0	1	?	NSIP
s049	NSIP	0	1	0	NSIP
s050	NSIP	0	1	0	NSIP
s051	NSIP	0	1	?	NSIP
s052	NSIP	0	1	0	NSIP
s053	NSIP	0	1	0	NSIP
s054	NSIP	0	1	0	NSIP
s055	NSIP	0	1	0	NSIP
s056	NSIP	0	1	0	NSIP

70% of IPF cases, in more than 60% of DIP, but only in 9% of NSIP cases. In that study, NSIP was confused most often with DIP, and less often with IPF. It seems that LAD makes possible to distinguish NSIP from the other entities in the majority of cases.

4.1.5. Validation

It has been shown in the previous section (Table 37) that the accuracy of classifying by LAD the 54 patients is of 100%. It should be added however that this result represents only the correctness of the proposed classification model when the entire dataset is used both as a training set, and as a test set. In order to establish the reliability of these classifications they have to be validated. Because of the very limited size of the dataset (in particular because of the availability of data for only 6 DIP patients and only 14 NSIP patients) the traditional partitioning of the dataset into a training and a test set would produce extremely small subsets, and therefore highly unreliable conclusions. In view of this fact, we shall test the accuracy of the LAD classification by cross-validation, using the so-called "jackknife" or "leave-one-out" method. As an example, the cross-validation of the classification results for the IPF/non-IPF model will be presented below.

The basic idea of the "leave-one-out" method is very simple. One of the observations is temporarily removed from the dataset, a classification method is "learned" from the set of all the remaining observations, and it is applied then to classify the extracted observation. This procedure is then repeated separately for every one of the observations in the

dataset. For example in the case of the IPF/non-IPF model we have to apply this procedure 54 times.

Table 38 shows the results of the "leave-one-out" procedure applied to the models. The table includes the results of directly applying leave-one-out experiments to the three models (IPF/non-IPF, NSIP/non-NSIP, DIP/non-DIP), as well as the resulting combined classifications. The combined classifications are then used to derive the final conclusion about the IPF/non-IPF character of each observation; the correctness of the conclusion (compared with the given classification) is presented in the last column of Table 38 ("evaluation").

The average accuracy of directly applying leave-one-out experiments for each problem is the following:

IPF/non-IPF	NSIP/non-NSIP	DIP/non-DIP
84.85%	78.21%	89.58%

For the combined classifications it can be seen that out of 54 observations, 44 are classified correctly, there are 6 errors (the IPF patients s009, s010 and s021 are classified as non-IPF, and the non-IPF patients s042, s047, and s053 are classified as IPF), two patients (s007 and s052) are unclassified, and for two other patients (s016 and s055) the classifications ("IPF or NSIP") are imprecise. The accuracy of the combined classifications for the IPF/non-IPF problem is 83.6%.

In view of the very small size of the dataset, the results of the leave-one-out tests can be viewed as extremely encouraging.

Table 38

Obs.	Given Classification	Classification by Leave-One-Out			Derived Classification	Conclusion	
		IPF/non-IPF	NSIP/non-NSIP	DIP/non-DIP		IPF/non-IPF	Evaluation
s001	DIP	0	?	1	DIP	0	correct
s002	DIP	0	0	1	DIP	0	correct
s004	DIP	0	0	1	DIP	0	correct
s005	DIP	0	1	1	DIP or NSIP	0	correct
s006	DIP	0	1	1	DIP or NSIP	0	correct
s007	DIP	0	0	0	?	?	unclassified
s008	IPF	1	0	0	IPF	1	correct
s009	IPF	0	?	0	NSIP	0	error
s010	IPF	0	1	0	NSIP	0	error
s011	IPF	1	0	0	IPF	1	correct
s012	IPF	1	0	0	IPF	1	correct
s013	IPF	1	0	0	IPF	1	correct
s014	IPF	1	0	0	IPF	1	correct
s015	IPF	1	0	0	IPF	1	correct
s016	IPF	1	1	0	IPF or NSIP	?	imprecise
s017	IPF	1	0	0	IPF	1	correct
s018	IPF	1	0	0	IPF	1	correct
s019	IPF	1	0	0	IPF	1	correct
s020	IPF	1	0	0	IPF	1	correct
s021	IPF	0	1	0	NSIP	0	error
s022	IPF	1	0	0	IPF	1	correct
s023	IPF	1	0	0	IPF	1	correct
s024	IPF	1	0	0	IPF	1	correct
s025	IPF	1	0	0	IPF	1	correct
s026	IPF	1	0	0	IPF	1	correct
s027	IPF	1	0	0	IPF	1	correct
s028	IPF	1	0	0	IPF	1	correct
s029	IPF	1	0	0	IPF	1	correct
s030	IPF	1	0	0	IPF	1	correct
s031	IPF	1	0	0	IPF	1	correct
s032	IPF	1	0	0	IPF	1	correct
s033	IPF	1	0	0	IPF	1	correct
s034	IPF	?	0	0	IPF	1	correct
s035	IPF	1	0	0	IPF	1	correct
s036	IPF	1	0	0	IPF	1	correct
s037	IPF	1	0	0	IPF	1	correct
s038	IPF	1	0	0	IPF	1	correct
s039	IPF	1	0	0	IPF	1	correct
s040	IPF	1	0	0	IPF	1	correct
s041	IPF	1	0	0	IPF	1	correct
s042	NSIP	1	0	0	IPF	1	error
s043	NSIP	0	1	?	NSIP	0	correct
s044	NSIP	0	1	0	NSIP	0	correct
s045	NSIP	0	1	0	NSIP	0	correct
s047	NSIP	1	?	0	IPF	1	error
s048	NSIP	0	?	1	DIP	0	correct
s049	NSIP	0	1	0	NSIP	0	correct
s050	NSIP	0	1	0	NSIP	0	correct
s051	NSIP	0	1	?	NSIP	0	correct
s052	NSIP	0	0	0	?	?	unclassified
s053	NSIP	1	0	0	IPF	1	error
s054	NSIP	0	1	0	NSIP	0	correct
s055	NSIP	1	1	0	NSIP or IPF	?	imprecise
s056	NSIP	0	1	0	NSIP	0	correct

4.1.6. Attribute Analysis

4.1.6.1. Importance of Attributes

A simple measure of the importance of an attribute is the frequency of its inclusion in the patterns appearing in the model. For example, attribute 1 (IIT) appears in 11 (i.e., in 27.5%) of the 40 patterns of the IPF/non-IPF model in Table 36. The frequencies of all the 13 attributes in the models are shown in Table 39 for the three LAD models considered, along with the averages of these three indicators.

Table 39

FREQUENCIES OF ATTRIBUTES IN MODELS

Attributes	IPF/non-IPF	NSIP/non-NSIP	DIP/non-DIP	Average
IIT	0.275	0.25	0.343	0.289
HC	0.525	0.813	0.357	0.565
TB	0	0	0.238	0.079
GG1	0	0	0	0.000
BRVX	0.125	0.219	0.381	0.242
PL	0.2	0.094	0	0.098
HPL	0.4	0.375	0	0.258
SL	0.3	0.156	0	0.152
AC	0	0	0	0.000
N	0	0	0	0.000
GG2	0.25	0.688	0	0.313
RET	0.5	0.5	0.376	0.459
GG2/RET	0.475	0	0.662	0.379

Two of the most important conclusions which can be seen in this table indicate that:

- the most influential attributes are ground-glass attenuation/reticulation score (GG2/RET), honeycombing (HC), ground-glass attenuation score (GG2) and reticulation score (RET);
- the attributes ground-glass attenuation (GG1), airspace consolidation (AC) and nodules (N) have no influence on the classification.

4.1.6.2. Promoting and Blocking Attributes

We shall illustrate the promoting or blocking nature of some attributes on the IPF/non-IPF model shown in Table 36. It can be seen in the table that every positive pattern which includes a condition on HC (honeycombing) requires that HC=1. Conversely, every negative pattern which includes a condition on HC requires that HC=0. This means that if a patient is known to be a non-IPF case with HC=1, and all the attributes of another patient have identical values except for HC which is 0, then this second patient is certainly not an IPF case. This type of monotonicity means simply that HC is a "promoter" of IPF. It is easy to see that the attribute PL (polygonal lines) has a similar property.

On the other hand, the attribute BRVX (peri-bronchovascular thickening) appears to have a converse property. Indeed, every positive pattern which includes this attribute requires that BRVX=0, while the only negative pattern (N19) which includes it requires that BRVX=1. Therefore if a patient's BRVX would change from 1 to 0, the patient's condition would not change from IPF to non-IPF (assuming again that none of the other

attributes change their values). Similarly to the previous case, this type of monotonicity means simply that BRVX is a "blocker" of IPF.

In this way the IPF/non-IPF model allows the identification of two promoters and of one blocker. None of the other attributes in the support set appear to be promoters or blockers.

A similar analysis of the DIP/non-DIP model shows that honeycombing (HC) and ground-glass attenuation/reticulation score (GG2/RET) are promoters of DIP, while peri-bronchovascular thickening (BRVX), intralobular interstitial thickening (IIT), traction bronchiectasis (TB) and reticulation score (RET) are blockers. Also, the analysis of the NSIP/non-NSIP model shows that peri-bronchovascular thickening (BRVX) is a promoter of NSIP, while honeycombing (HC), polygonal lines (PL) and septal lines (SL) are blockers of NSIP.

To conclude, we show in Table 40 the promoters and blockers which have been identified for the three forms of idiopathic interstitial pneumonias.

Table 40
PROMOTERS AND BLOCKERS FOR CT DATA

	Idiopathic Pulmonary Fibrosis	Desquamative Interstitial Pneumonia	Non Specific Interstitial Pneumonia
honeycombing	promoter	promoter	blocker
polygonal lines	promoter		blocker
peri-bronchovascular thickening	blocker	blocker	promoter
ground-glass attenuation/reticulation score		promoter	
intralobular interstitial thickening		blocker	
traction bronchiectasis		blocker	
reticulation score		blocker	
septal lines			blocker

4.1.7. Conclusion

It has been shown that it is possible to use a computational technique (LAD) for analyzing CT data for distinguishing with high accuracy different entities (IPF, NSIP and DIP) of idiopathic interstitial pneumonias (IIPs). This is particularly important for NSIP which is as yet poorly defined. It was also shown that the patterns developed by LAD techniques provide additional information about outliers, redundant features, the relative significance of the attributes, and allow to identify promoters and blockers of various forms of IIPs. These encouraging results will form the basis of a forthcoming study of a broader population of IIPs, which will include not only CT data, but also clinical and biological ones.

4.2. Breast Cancer Prognosis by Combinatorial Analysis of Gene Expression Data[4]

Microarray gene expression technology has provided extensive datasets that describe patients with cancer in a new way. Several methodologies have been used to extract information from these datasets. In this section we use the methodology of logical analysis of data to reanalyze the publicly available microarray dataset reported by van't Veer *et al*. [90]. The motivation for using yet another method to analyze these data was the expectation that the specific aspects of LAD, and especially the combinatorial nature of its approach, would allow the extraction of new information on the problem of

[4] The results presented in this section are based on joint work with G. Alexe, S. Alexe, D.E. Axelrod, T.O. Bonates, M. Reiss, and P.L. Hammer, published in [5].

metastasis-free survival of breast cancer patients, and in particular on the role of various significant combinations of genes that may have an influence on this outcome.

The main goal of the study by van't Veer *et al*. was to predict the clinical outcome of breast cancer (that is, to identify those patients who will develop metastases within 5 years) based on analysis of gene expression signatures. The crucial importance of this problem arises from the fact that the available adjuvant (chemo or hormone) therapy, which reduces by about one-third the risk for distant metastases, is not really necessary for 70-80% of the patients who currently receive it. Moreover, this therapy can have serious side effects and involves high medical costs. The study by van't Veer *et al*. illustrates clearly that machine learning techniques, data mining, and other new techniques applied to DNA microarray analysis can outperform most clinical predictors currently in use for breast cancer. The study concludes that the new findings, '… provide a strategy to select patients who would benefit from adjuvant therapy'.

A specific feature of datasets coming from genomics is the presence of a very large number of measurements concerning gene expressions but only a relatively small number of observations. For instance, the attributes in the van't Veer study correspond to more than 25,000 human genes, whereas the number of cases was only 97. In that dataset, each case is described by the expression levels of 25,000 genes, as measured by fluorescence intensities of RNA hybridized to microarrays of oligonucleotides. The cases included in the dataset are 97 lymph-node-negative breast cancer patients, who are grouped into a training set of 78 and a test set of 19 cases. The training set includes 34 positive cases

(having a ‘poor prognosis’ signature; that is, having fewer than 5 years of metastasis-free survival) and 44 negative cases (having a ‘good prognosis’ signature; i.e., having more than 5 years of metastasis-free survival). The test set includes 12 positive and 7 negative cases.

The van’t Veer study used DNA microarray analysis in primary breast tumors, and “applied supervised classification to identify gene expression signature strongly predictive of a short interval to distant metastases (‘poor prognosis’ signature) in patients without tumor cells in local lymph nodes at diagnosis (lymph node negative)”. The study identified 231 genes as being significant markers of metastases, all of whose correlations with outcome exceeded 0.3 in absolute value, and it constructed an optimal prognosis classifier based on the best 70 genes. In the training set the system predicted correctly the class of 65 of the 78 cases (that is, with an accuracy of 83.3%, corresponding to a weighted accuracy of 83.6%), whereas in the test set it predicted correctly the class of 17 of the 19 cases (that is, with an accuracy of 89.5%, corresponding to a weighted accuracy of 88.7%). Weighted accuracy is defined as the average of the proportion of correctly predicted cases within the set of positive cases and that of correctly predicted negative cases in the dataset.

Numerous statistical and machine-learning methods have been successfully applied to the analysis of microarray datasets; these methods include cluster analysis (hierarchical clustering ([14], [23], [34], [37]), self-organizing maps ([25], [86], [88]), and two-way clustering [36], regression analysis [56], nearest neighborhood methods [98], decision

trees ([19], [85], [99]), artificial neural networks ([38][59]), support vector machines ([23], [35], [76], [84]), principal component analysis ([9], [49], [50], [77], [87]), singular value decomposition ([10], [11], [51], [64]), and multidimensional scaling ([60], [98]). A pattern-based recognition method has been developed using other kinds of data for prediction of outcome in preclinical and clinical trials of cancer patients ([57], [61]).

Specific features of the LAD approach include the exhaustive examination of the entire set of genes (without excluding those that have low statistical correlations with the outcome, or those that have low expression levels), focusing on the classification power of combinations of genes (without confining attention only to individual genes) and on the possibility of extracting novel information on the role of genes and of combinations of genes through the analysis of these exhaustive lists.

LAD has been shown to offer important insights into problems ranging from oil exploration [18], labor productivity analysis [45] and country creditworthiness evaluation [46], to medical application (for example, risk evaluation among cardiac patients ([6], [63]), polymer design for artificial bones [1], genomic-based diagnosis and prognosis of lymphoma [4], and proteomics-based ovarian cancer diagnosis [3].

We develop a new type of classification model that can distinguish between patients who will have a metastasis-free survival of 5 years from the others.

4.2.1. Materials and Methods

It can be expected that ‘large’ or ‘small’ values of the expression levels of certain genes can determine the poor or bad prognosis of a breast cancer patient. In order to express such relations in more precise terms, it is natural to replace terms such as ‘large’ and ‘small’ with conditions of the type ‘ … is more than’ or ‘… is less than’ a certain value. It is therefore natural to examine the role of well chosen cut points associated with the expression levels of genes. For instance, the observation that low intensity levels of gene Contig15031_RC are (more or less) characteristic for a poor prognosis is imprecise; it can be reformulated as the ultra-simplistic classification system, ‘If the intensity level of gene Contig15031_RC is at most 0.055 then the patient has a poor prognosis’. The assumption of this rule is valid for 25 positive and 11 negative cases in the training set (that is, it has a sensitivity of 25/34 = 73.5% and a specificity of 33/44 = 75%).

Combinations of such cut point based conditions naturally extend this idea. For instance, the combined requirement of satisfying simultaneously the three conditions ‘The intensity level of gene Contig15031_RC is at most 0.055’ and ‘The intensity level of gene NM_004035 is at least -0.106’ and ‘The intensity level of the gene NM_003239 is at most -0.014’ is fulfilled by 22 of the 34 positive cases in the dataset and by none of the negative ones. Again, these three requirements could be viewed as a classification system of poor prognosis cases, having a sensitivity of 64.7% and a specificity of 100%.

Such ideas are at the foundation of LAD. The essence of LAD is to detect patterns, or combinatorial biomarkers (i.e., simple classifiers consisting of restrictions imposed on the

values of the expression levels of the intensities of a combination of several genes); to generate patterns exhaustively and in an algorithmically efficient way; to use the collection of patterns as a prognostic system and thoroughly validate it; to extract from this collection as much additional information as possible about the role and nature of genes in the dataset (that is, to detect promoters and blockers); and to study the common characteristics of groups of patients that satisfy similar patterns.

The LAD method was trained on the same training set of 78 samples used by van't Veer *et al.* [90]. The prognosis results for LAD were validated on the same test set of 19 samples used by van't Veer *et al.* The samples in the test set were disregarded during the training procedure.

Support Set Selection

In order to distinguish between measurements of good and of poor prognosis patients, only a tiny fraction of the information contained in the (original or binarized) dataset is needed. In particular, all of the information about the vast majority of the genes in the dataset is redundant. Moreover, even for the genes that are not redundant, only a few (usually only one) of the corresponding binary variables are needed. A set of binary variables that are sufficient to distinguish poor from good prognosis cases is called a support set. A support set is called 'minimal' if none of its proper subsets is a support set; clearly, not every minimal support set is of minimum size. It is important to note that a dataset may admit hundreds or thousands of minimal support sets. The reduction of a large dataset to a substantially smaller one that includes only the variables in the chosen

support set allows a major simplification of the problem, and has great importance for diagnosis and prognosis (although, in some cases, the presence of a limited number of redundant variables may be acceptable in terms of ensuring greater stability of results).

The problem of finding minimal support sets has been modeled elsewhere ([17], [18], [27]) as a typical 'set-covering' problem, and numerous methods are known in combinatorial optimization for the solution of this problem. In our case, the excessive dimensions of the associated set-covering problem (approximately 20,000 constraints involving between 2 and 3 million 0-1 variables) required the use of powerful heuristics to trim down the size of the problem. In order to be able to handle the large problems typical for genomic and proteomic datasets, a general heuristic size-reduction procedure has been developed [8]. The essence of this method is to balance the conflicting criteria of minimizing size and maximizing discrimination between positive and negative observations**.** In contrast to many statistically based methods, the support set generation procedures of LAD are guided by the collective strength of the subsets of variables, without being necessarily restricted to those variables that have the highest individual correlation coefficients with the outcome.

We restricted our study only to those 13,387 genes whose log-ratio measurements of fluorescence intensities are known for every single patient (that is, we eliminated those genes that include missing data). After that the feature selection procedure [8] was applied. This procedure consists of two stages. In a first 'filtering' stage, a relatively small subset of relevant features was identified on the basis of several combinatorial,

statistical, and information/theoretical criteria (for example, separation measure, envelope eccentricity, system entropy, signal to noise ratio). In the second stage, the importance of variables selected in the first step was evaluated based on the frequency of their participation in the set of all patterns and generated using an efficient, total polynomial time algorithm [7], and a large proportion of the low impact variables was eliminated. This step was applied iteratively, until a Pareto-optimal support set of 17 variables (shown in Table 41) was arrived at, which balanced the conflicting criteria of simplicity and accuracy.

Binarization

We have used a simple binarization technique to replace the expression level of each gene by several binary (0-1) variables, simply indicating whether the expression level does or does not exceed certain thresholds. In order to achieve this, we introduced nine cut points into the range of fluorescence intensities of each gene.

Pattern and Model Generation

In order to ensure high reliability of the patterns used in the model, we restricted our search to patterns of prevalence at least 20. Furthermore, in order to maximize the explanatory power of the patterns detected, we restricted our search to patterns of degree 3 at most (that is, involving at most three genes). In this way, using the support set of 17 genes we have identified a pandect of 201 positive and 232 negative patterns and extracted from it the model consisting of only 20 positive and 20 negative patterns, as shown in Table 42.

Each row in Table 42 describes a pattern. The first entry in the row is the name of the pattern (for example, P1 in the first row describes the first positive pattern). The next 17 entries describe the defining conditions of that pattern (for example, P1 is described by the three conditions 'Gene NM_001756 > -0.42', 'Contig15031_RC ≤ 0.09', and 'Contig65439 ≤ 0.06'). The last two entries indicate the positive and negative coverages (that is, the number of cases satisfying the defining conditions of the pattern) and prevalences (that is, proportion of positive, or negative, cases satisfying the defining conditions) of the pattern on the training set. For instance, P1 covers 19 of the 34 positive cases and none of the negative cases in the training set; therefore, its positive and negative prevalences on the training set are 55.9% and 0%, respectively.

Prognosis

The availability of the model makes it possible to classify new (that is, not yet seen) observations as being positive or negative. As a matter of fact, diagnosis and prognosis are perhaps the most important applications of LAD to biomedical problems. The most direct way to apply LAD to prognostic problems is to examine which patterns are displayed by a new case. If the case displays only positive patterns, then it is assigned a poor prognosis. Similarly, if it displays only negative patterns, then it is assigned a good prognosis. If the case does not display any pattern, then no prognosis can be assigned to it; it should be noted that this situation is extremely rare and did not occur at all in the present study. Finally, if a case displays both positive and negative patterns, then a simple weighting procedure is applied to determine whether the positive or the negative patterns are predominant. The weighting procedure consists simply of comparing the proportion

of the displayed positive patterns in the set of all positive patterns contained in the model (pandect), with the analogous proportion of negative patterns.

To illustrate the way in which a model can be used to predict the positive (negative) nature of a 'new' patient, let us consider one which, for example, will satisfy one of the 20 positive patterns and five of the 20 negative patterns appearing in the model. Therefore, the 'prognostic index' of this patient will be (1/20) - (5/20) = -0.2; because the prognostic index is negative, the model predicts this patient to be in the 'negative' class.

Calibration

The quality of the prognosis given by the model is a consequence of the choice of several control parameters. The collection of control parameters include the number of cutpoints per gene, upper bounds on the size of support sets, pattern degrees, and lower bounds on pattern prevalence. The control parameters define uniquely the model. The best values of the control parameters were determined iteratively by assigning some values to them, constructing the associated model, verifying the correctness of its predictions, reassigning the values, and continuing this sequence of steps until we arrived at a model with highly accurate predictions.

The entire calibration process was conducted only on the training set and it was intended to identify the best parameters to be used in the construction of the LAD model.

Validation

The model has been validated in two ways. First, the prediction of the model built on the training set was checked on the test set. This is the most frequently used validation method. In order to increase the reliability of the proposed model, an additional validation procedure was applied. In this second validation procedure, we created a new dataset consisting of all of the observations in the original training and test sets. The procedure consisted in the application of the usual cross-validation techniques (twenty 5-folding) to this augmented dataset, using the parameters found at the calibration stage.

4.2.2. Results

4.2.2.1. Prognostic System

We examine the model built on the 17 genes shown in Table 41. The functions of these genes, obtained from the DAVID database [30], are summarized in Table 41.

Based on this 17-gene support set we constructed the model shown in Table 42, which consists of 20 positive and 20 negative patterns. It can be seen that the patterns are very robust, having prevalences of up to almost 56% in the positive case and above 34% in the negative case.

The classification provided by the model for the 34 patients with poor prognosis and the 44 patients with good prognosis makes no errors in the training set (weighted accuracy = 100%). More significantly, on the 19-case test set (which includes 12 positive and seven

Table 41

THE 17-GENE SUPPORT SET

Gene Index	Van't Veer id	GeneBank	DAVID_GENE_NAME
1	AB033007	AB033007	KIAA1181 protein
2	NM_001661	NM_001661	ADP-ribosylation factor 4-like
3	NM_001756	NM_001756	Serine (or cysteine) proteinase inhibitor, clade A (alpha-1 antiproteinase, antitrypsin), member 6
4	AF148505	AF148505	Aldehyde dehydrogenase 6 family, member A1
5	Contig42421_RC	AI912791	F-box protein 16
6	NM_003748	NM_003748	Aldehyde dehydrogenase 4 family, member A1
7	NM_020974	NM_020974	Signal peptide, CUB domain, EGF-like 2
8	AL080059	AL080059	TSPY-like 5
9	AL110129	AL110129	Mitochondrial ribosomal protein S22
10	Contig15031_RC	AI347425	Oligodendrocyte myelin glycoprotein
11	Contig65439	AI572600	Chromosome 20 open reading frame 178
12	Contig37063_RC	AA579843	Poly (ADP-ribose) glycohydrolase
13	Contig41383_RC	AA142876	Asparaginase like 1
14	AL049689	AL049689	Tenascin N
15	Contig63102_RC	AI583960	Hypothetical protein FLJ11354
16	Contig55574_RC	AA524093	F-box protein 41
17	Contig38451_RC	AA497035	Not available

Table 42

LAD MODEL CONSISTING OF 20 POSITIVE AND 20 NEGATIVE PATTERNS ON SUPPORT SET OF 17 GENES

Patterns	Definition of Patterns																	Patterns' coverages (prevalences) on training set	
	AB033007	NM_001661	NM_001756	AF148505	Contig42421_RC	NM_003748	NM_020974	AL080059	AL110129	Contig15031_RC	Contig65439	Contig37063_RC	Contig41383_RC	AL049689	Contig63102_RC	Contig55574_RC	Contig38451_RC		
	Attr. 1	Attr. 2	Attr. 3	Attr. 4	Attr. 5	Attr. 6	Attr. 7	Attr. 8	Attr. 9	Attr. 10	Attr. 11	Attr. 12	Attr. 13	Attr. 14	Attr. 15	Attr. 16	Attr. 17	Pos Prev	Neg Prev
P1			>-0.42							≤0.09	≤0.06							19 (55.9%)	0
P2						≤0.07					≤-0.01		≤0.07					18 (52.9%)	0
P3			>-0.42						≤0.06		≤0.06							18 (52.9%)	0
P4											≤-0.01	≤0.38	≤0.07					18 (52.9%)	0
P5						≤0.07	≤-0.45				≤0.06							17 (50.9%)	0
P6	≤0.33										≤-0.01				≤-0.104			16 (47.1%)	0
P7						≤0.07					≤-0.01					>-0.02		16 (47.1%)	0
P8						≤0.07		>-0.295		≤0.033								16 (47.1%)	0
P9			>-0.42								≤0.06			≤-0.001				14 (41.2%)	0
P10		>-0.1			≤-0.11						≤-0.01							14 (41.2%)	0
P11		≤0.03					≤-0.45						≤0.19					13 (38.2%)	0
P12						≤0.07		>-0.295								>0.08		13 (38.2%)	0
P13		≤0.35						>-0.295								>0.08		13 (38.2%)	0
P14												≤0.3		≤-0.001		>0.08		13 (38.2%)	0
P15		≤0.03											>-0.16 ≤0.07					12 (35.3%)	0
P16						≤0.35	>-0.96, ≤-0.7											10 (29.4%)	0
P17			>-0.22				≤0.055	>-0.295										10 (29.4%)	0
P18			>-0.22			>-0.48		>-0.1										10 (29.4%)	0
P19			>-0.22		≤0.32		≤0.055											10 (29.4%)	0
P20			>-0.22					>-0.1			>-0.27							10 (29.4%)	0

N1		>0.09													>-0.005			0	15 (34.1%)
N2								≤-0.295				>0.06				≤-0.02		0	15 (34.1%)
N3					>-0.11							≤0.14				≤-0.02		0	15 (34.1%)
N4							>0.055									≤-0.02	≤1.88	0	14 (31.8%)
N5													>0.07	>-0.001 ≤0.17				0	14 (31.8%)
N6							>0.055									≤-0.02		0	14 (31.8%)
N7			≤-0.22								>0.06				>-0.005			0	14 (31.8%)
N8							>0.055									≤-0.02		0	14 (31.8%)
N9		>0.09													>-0.005		>-0.083	0	14 (31.8%)
N10										>0.09						>-0.12 ≤0.08		0	13 (29.5%)
N11				>-0.03								>0.06 ≤0.14						0	13 (29.5%)
N12				>0.077		≤0.35	>0.055											0	13 (29.5%)
N13				>0.077					≤0.34							≤-0.0213		0	13 (29.5%)
N14			≤-0.22		≤0.18		>0.055											0	13 (29.5%)
N15	≤0.21			>0.077												≤-0.0213		0	12 (27.3%)
N16								≤-0.49								>-0.1207	≤1.877	0	12 (27.3%)
N17												>0.06 ≤0.14				≤-0.0213		0	12 (27.3%)
N18												≤0.22				>-0.12 ≤-0.02		0	12 (27.3%)
N19		≤0.16	≤-0.42				>-0.197											0	12 (27.3%)
N20						>-0.204	>-0.197				>0.13							0	11 (25.0%)

negative cases), the system makes only one error and classifies correctly all of the other cases; thus, the system's weighted accuracy is 92.9%. The only error is of type 2, and it is due to the incorrect classification of negative sample 119. The supplementary validation tests based on an additional series of twenty 5-folding experiments on the combined dataset of 97 cases showed an average weighted accuracy of 81.7%.

4.2.2.2. Significant Biomarkers

Based on the frequency of inclusion of genes in the positive patterns, it can be seen that Contig65439 (chromosome 20 open reading frame 178*)* plays a significant role in determining a poor prognosis, because it appears in 10 of the 20 positive patterns of the model. Similarly, Contig 55574_RC (F-box protein 41) plays a significant role in determining good prognosis, because it appears in 11 of the 20 negative patterns of the model.

4.2.2.3. Promoters and Blockers

A gene with the property that an increase in the intensity level of its expression (while the expression levels of the all other genes remain unchanged) can sometimes worsen the prognosis, but can never improve it, will be called a '*promoter*'. Similarly, a gene with the property that a decrease in the intensity level of its expression (while the expression levels of the all other genes remain unchanged) can sometimes improve the prognosis,

but can never worsen it, will be called a '*blocker*'. Clearly, not every gene is a promoter or a blocker.

The model can identify promoters and blockers in the following way. If every occurrence of a gene among the positive patterns imposes a lower bound on its expression level (i.e., in all those patterns whose definition includes a condition concerning that gene, the condition is of the form 'the expression level of that gene is ≥ than a prescribed level'), while every occurrence of the same gene among the negative patterns imposes an upper bound of its expression level (that is, in all those patterns whose definition includes a condition concerning that gene, the condition is of the form 'the expression level of that gene is ≤ than a prescribed level'), then it can be concluded that an increase in the expression level of that gene (assuming that the expression levels of all the other genes remain unchanged) may have as a result the activation of more positive patterns and/or the deactivation of some negative ones. Therefore, an increase in the expression level of such a gene can only increase the chances of metastasis formation. Such a gene will be called a promoter.

Similarly, if every occurrence of a gene among the positive patterns imposes an upper bound on its expression level (namely, in all those patterns whose definition includes a condition concerning that gene, the condition is of the form 'the expression level of that gene is ≤ than a prescribed level'), while every occurrence of the same gene among the negative patterns imposes a lower bound of its expression level (namely, in all those patterns whose definition includes a condition concerning that gene, the condition is of

the form ‘the expression level of that gene is ≥ than a prescribed level’), then it can be concluded that an increase in the expression level of that gene (assuming that the expression levels of all the other genes remain unchanged) may have as a result the activation of more negative patterns and/or the deactivation of some positive ones. Therefore, an increase of the expression level of such a gene can only decrease the chances of metastasis formation. Such a gene will be called a blocker.

Using these definitions, it is shown in Table 42 that genes NM_001756, AL080059, and Contig55574_RC are promoters, whereas genes NM_020974, Contig65439, Contig15031_RC, Contig41383_RC, and Contig63102_RC are blockers. The genes AF148505 and AL049689 also exhibit blocker characteristics although to a somewhat lesser extent; we view them as weak blockers.

4.2.2.4. Special Classes of Positive Cases

In order to discover special classes, we conducted a series of two-means clustering experiments of the positive observations, but they did not reveal the existence of any special subgroups of observations. However, using the pattern-based representation of the positive cases (as described in the introduction), two-means clustering revealed the existence of two very special classes of patients. Despite the random element present in the nature of the two-means clustering procedure, it transpired that in the 100 experiments we have carried out, the positive observations were repeatedly and consistently clustered into the same two subgroups, which are denoted below by P^{+++}

(consisting of patient numbers 48, 50, 51, 59, 66, 68, and 69) and P^{+} (consisting of patient numbers 46, 52, 54, 55, 60, 62, 63, 73, and 78) respectively; these subgroups have the following distinctive properties.

Cohesion

The seven patients belonging to P^{+++} are assigned to a common cluster in 86% of the experiments, whereas the nine patients belonging to P^{+} are assigned to another common cluster in 98% of the experiments.

Predictability

In each validation experiment of the prognostic system by the leave-one-out method, the prognosis of every single observation in P^{+++} was correct; the 100% accuracy of the prognostic system on set P^{+++} is much higher than its 82.3% accuracy on the set of positive cases not contained in P^{+++}. On the other hand, the accuracy of predictions for the patients in class P^{+} is only 55.6%.

Distinctive coverage by patterns

Each patient belonging to class P^{+++} satisfies 50-90% of the positive patterns (68.5% on average), whereas each patient belonging to P^{+} satisfies only 10-30% of the positive patterns (20% on average).

Distinctive gene expression ranges

The smallest interval of the 17-dimensional real space containing P^{+++} does not contain any other positive or negative observation, whereas the one containing P^{+} also contains seven negative observations (Table 43).

Table 43

DESCRIPTION OF THE CASES IN THE SPECIAL POSITIVE CLASS P^{+++}

	Gene Accession Number	AB033007	NM_001661	NM_001756	AF148505	Contig42421_RC	NM_003748	NM_020974	AL080059	AL110129	Contig15031_RC	Contig65439	Contig37063_RC	Contig41383_RC	AL049689	Contig63102_RC	Contig55574_RC	Contig38451_RC
P^{+++}	Lower bound	-0.13	-0.123	-0.193	-0.362	-0.281	-0.372	-1.125	-0.066	-0.078	-0.077	-0.268	-0.193	-0.095	-0.242	-0.453	-0.369	-0.119
	Upper bound	0.108	0.044	0.381	0.116	-0.058	0.041	0.783	0.518	0.054	0.071	-0.009	0.116	0.07	0.115	0.048	0.525	0.268
Positive cases not in P^{+++}	Lower bound	-0.174	-0.129	-0.708	-0.514	-0.601	-2	-1.337	-0.783	-2	-0.044	-0.263	-0.222	-0.567	-0.291	-0.345	-0.334	-0.147
	Upper bound	0.363	0.329	0.638	0.386	0.671	0.487	0.942	0.776	0.418	0.418	0.211	0.494	0.393	0.431	0.444	0.256	2

Statistical distinctions of clinical features

We shall say that feature '*f*' is a 'contrastor' of subset *S*` of the positive cases from the complementary set *S*`` (consisting of those positive cases that do not belong to *S*`) if the following two conditions hold: the average value of *f* in *S*` does not belong to the 95% confidence interval of the values of *f* in *S*``; and the average value of *f* in *S*`` does not belong to the 95% confidence interval of the values of *f* in *S*`. With this definition, it can be seen in Table 44 that the diameter and, to some extent, the grade are contrastors, which distinguish P^{+++} from its complement in the positive class. It can be also observed (see Table 45) that class P^{+} has some distinguishing characteristics (for example, the average PRp [progesterone receptor] of the patients in this class is 55.6, whereas the average PRp of the positive patients outside class P^{+} is 27.6, with the 95% confidence interval ranging from 12.6 to 42.6).

Table 44

CONTRASTORS DIFFERENTIATING THE POSITIVE CASES IN P^{+++} FROM THE POSITIVE CASES OUTSIDE P^{+++}

		Diameter (mm)	Grade
P^{+++}	Average	30.71	3.00
	CI (95%)	25.31	3.00
		36.12	3.00
Positive cases outside P^{+++}	Average	22.67	2.81
	CI (95%)	20.11	2.67
		25.22	2.96

Table 45

CONTRASTORS DIFFERENTIATING THE POSITIVE CASES IN P^+ FROM THE POSITIVE CASES OUTSIDE P^+

		PRp
P^+	Average	55.56
	CI (95%)	26.50
		84.61
Positive cases outside P^+	Average	27.60
	CI (95%)	12.59
		42.61

Summary

It is clear that the classes P^+ and P^{+++} are very special and that all of the characteristics listed above indicate that it is most likely that the patients belonging to class P^{+++} have a very strong tendency toward developing metastases, whereas those in P^+ have a substantially reduced tendency.

4.2.2.5. Special Classes of Negative Cases

Using the pattern-based representation of cases described in the introduction, we also carried out 100 two-means clustering experiments within the set of negative observations. Despite of the random element present in the nature of the two-means clustering procedure, it transpired that, similar to the positive class, the negative class also contains

two disjointed (but not exhaustive) special subclasses. These are denoted below by N^{---} (consisting of patient numbers 10, 18, 21, 23, 30, 32, 37, and 38) and N^{-} (consisting of patient numbers 2, 3, 4, 6, 8, 9, 11, 12, 13, 15, 16, 17, 19, 20, 22, 24, 26, 27, 28, 33, 34, 36, 39, 40, 41, and 44), respectively, and have the following distinctive properties.

Cohesion

The eight patients belonging to N^{---} are assigned to a common cluster in 88% of the experiments, whereas the 26 patients belonging to N^{-} are assigned to a common cluster in 95% of experiments.

Predictability

In each validation experiment of the prognostic system by the leave-one-out method, the prognosis of every single observation in N^{---} was correct; the 100% accuracy of the prognostic system on set N^{---} is much higher than its 77.8% accuracy on the set of negative cases not contained in N^{---}. On the other hand, the accuracy of predictions for the patients in class N^{-} is only 73.1%.

Distinctive coverage by patterns

Each patient belonging to the class N^{---} satisfies 50-70% of the negative patterns (57.5% on average), whereas each patient belonging to N^{-} satisfies only 5-35% of the negative patterns (20% on average).

Distinctive gene expression ranges

The smallest interval of the 17-dimensional real space containing N^{---} does not contain any other positive or negative observation, whereas the one containing N^{-} also contains eight positive observations (Table 46).

Statistical distinctions of clinical features

Similar to the positive case, we shall say that feature '*f*' is a 'contrastor' of subset *S*` of the negative cases from the complementary set *S*`` (consisting of those negative cases that do not belong to *S*`) if the following two conditions hold: the average value of *f* in *S*` does not belong to the 95% confidence interval of the values of *f* in *S*``; and the average value of *f* in *S*`` does not belong to the 95% confidence interval of the values of *f* in *S*`. With this definition, it can be seen in Table 47 that grade, estrogen receptor positive, and (to some extent) lymphocytic infiltrate are contrastors of N^{---}. As far as class N^{-} goes, Table 48 shows the differences between the average values of some of the parameters in class N^{-} compared with average values of the same parameters in the set of negative cases outside N^{-}.

Summary

It is clear that the classes N^{-} and N^{---} are very special; all of the characteristics listed above indicate that it is most likely that patients belonging to class N^{---} are very strongly resistant to development of metastases, whereas those in class N^{-} have a substantially milder resistance.

Table 46

DESCRIPTION OF THE CASES IN THE SPECIAL NEGATIVE CLASS N^{---}

	Gene Accession Number	AB033007	NM_001661	NM_001756	AF148505	Contig42421_RC	NM_003748	NM_020974	AL080059	AL110129	Contig15031_RC	Contig65439	Contig37063_RC	Contig41383_RC	AL049689	Contig63102_RC	Contig55574_RC	Contig38451_RC
N^{---}	Lower bound	0.041	-0.112	-0.65	0.007	-0.126	0.059	-0.976	0.05	0.05	0.085	-0.266	0.062	-0.251	0.039	-0.022	-0.255	0.02
	Upper bound	0.21	0.228	0.166	0.453	0.386	0.675	-0.038	0.394	0.394	0.285	0.293	0.247	0.401	0.35	0.278	-0.024	0.303
Negative cases not in N^{---}	Lower bound	-0.144	-0.106	-0.734	-0.294	-0.407	-1.253	-0.844	-0.214	-0.214	-0.115	-0.307	-0.179	-0.291	-0.21	-0.395	-0.343	-0.206
	Upper bound	0.345	0.443	1.135	0.363	0.521	0.881	0.311	0.477	0.477	0.273	0.293	0.433	0.482	0.455	0.335	0.22	0.323

Table 47

CONTRASTORS DIFFERENTIATING THE NEGATIVE CASES IN N^{---} FROM THE NEGATIVE CASES OUTSIDE N^{---}

		Grade	ERp	Lymphocytic infiltrate
N^{---}	Average	1.75	78.75	0.00
	CI (95%)	1.43	61.60	0.00
		2.07	95.90	0.00
Positive cases outside N^{---}	Average	2.42	57.22	0.14
	CI (95%)	2.17	44.98	0.02
		2.67	69.46	0.25

Table 48

CONTRASTORS DIFFERENTIATING THE NEGATIVE CASES IN N^- FROM THE NEGATIVE CASES OUTSIDE N^-

		Follow-up time (years)	Grade	ERp	PRp	Lymphocytic infiltrate
N^-	Average	8.16	2.58	47.31	36.92	0.19
	CI (95%)	7.28	2.31	32.62	23.19	0.04
		9.04	2.85	62.00	50.65	0.35
Negative cases outside N^-	Average	9.48	1.89	81.11	56.94	0.00
	CI (95%)	8.20	1.58	71.23	40.61	0.00
		10.76	2.20	90.99	73.28	0.00

4.2.3. Discussion

On the training set of 34 positive and 44 negative cases, the model reported by van 't Veer *et al.* [90] misclassifies 12 positive and 3 negative cases. The proposed model classifies 100% of the cases in the training set correctly. On the 19-case test set, the van't Veer model misclassifies two cases, whereas the proposed model misclassifies one. We do not know whether the performance of the model presented by van't Veer *et al.* [90] has been subjected to cross-validation (for example, by *k*-folding or leave-one-out experiments), and therefore we can not conduct a comparison with the cross-validation results of LAD, as shown in Table 49.

Table 49

COMPARISON OF WEIGHTED ACCURACIES OF THE VAN'T VEER CLASSIFIER AND THE LAD MODEL

	Training set (78 cases)		Test set (19 cases)	Entire dataset (78 + 19 cases)
	Direct classification (%)	Cross-validation (%)	Direct classification (%)	Cross-validation (%)
Van 't Veer classifier	83.6	Not reported	88.7	Not reported
Enhanced LAD model	100	82.52	92.86	81.74

4.2.3.1.Comparison of Support Sets

The study by van't Veer *et al*. [90] considered two support sets consisting of 70 and 231 selected genes, whereas the model proposed in the present study used a support set of 17 genes. Accuracy in distinguishing cases of poor and good breast cancer prognosis provided by the subset of 70 genes selected by van't Veer *et al*. was revalidated and confirmed by van de Vijver and colleagues [91] in a different cohort of patients.

In order to assess further the performance of the reported subsets of 231 and of 70 genes selected by van't Veer *et al*. [90], and of the support set of 17 genes selected for the proposed LAD model, we applied LAD to each of these three subsets of genes. We then constructed separate predictive models on the training set and on the entire dataset (consisting of 78 and 97 samples, respectively), and tested their accuracy direct

application both to the training set of 78 and to the entire dataset of 97 samples, and also by cross-validation, consisting of twenty 5-folding experiments. The results are shown in Table 50.

Table 50

COMPARISON OF WEIGHTED ACCURACIES OF THE LAD MODELS CONSTRUCTED ON THREE DIFFERENT SUPPORT SETS

Support set	Training set (78 cases)		Test set (19 cases)	Entire dataset (78 + 19 cases)
	Direct classification (%)	Cross-validation (%)	Direct classification (%)	Cross-validation (%)
231 genes (van't Veer)	**100.00**	**79.48**	**84.52**	**78.35**
70 genes (van't Veer)	**99.26**	**75.43**	**84.52**	**74.06**
Proposed support set of 17 genes	**100.00**	**82.52**	**92.86**	**81.74**

Furthermore, we repeated the same type of experiments by comparing the weighted accuracies of applying five frequently used classification methods to the three support sets discussed above; these classification methods include artificial neural networks, support vector machines, logistic regression, nearest neighbors and decision trees, and are included in the publicly available software WEKA [96]. The results are given in Table 51 - Table 53 and show that the average weighted accuracy of the five methods applied to the support set of 17 genes compares favorably with the results obtained using the two larger support sets of van't Veer *et al.*

Table 51

WEIGHTED ACCURACIES OF VARIOUS MODELS CONSTRUCTED ON THE SUPPORT SET IDENTIFIED BY LAD

Method	Support set of 17 genes (LAD)			
	Training set (78 cases)		Test set (19 cases)	Entire dataset (78 + 19 cases)
	Direct classification (%)	Cross-validation (%)	Direct classification (%)	Cross-validation (%)
Artificial neural networks (1 hidden layer)	100.00	76.55	84.21	78.65
Support vector machines (linear kernel)	87.18	76.43	63.16	77.27
Logistic regression	94.87	76.87	73.68	77.95
Nearest neighbors	100.00	80.55	63.16	76.34
Decision trees (C4.5)	96.15	67.48	57.90	67.01
95% CI	91.03-100	71.33-79.82	59.20-77.65	71.25-79.64

Table 52

WEIGHTED ACCURACIES OF VARIOUS MODELS CONSTRUCTED ON THE SUPPORT SET OF 70 GENES IDENTIFIED BY VAN'T VEER ET AL.

Method	Support set of 70 genes (van 't Veer)			
	Training set (78 cases)		Test set (19 cases)	Entire dataset (78+19 cases)
	Direct classification (%)	Cross-validation (%)	Direct classification (%)	Cross-validation (%)
Artificial neural networks (1 hidden layer)	100.00	80.16	42.11	71.65
Support vector machines (linear kernel)	96.15	82.01	57.90	77.03
Logistic regression	100.00	73.52	47.37	73.79
Nearest neighbors	100.00	71.58	63.16	71.77
Decision trees (C4.5)	96.15	60.49	42.11	61.89
95% CI	96.61-100	66.09-81.01	42.15-58.91	66.27-76.18

Table 53

WEIGHTED ACCURACIES OF VARIOUS MODELS CONSTRUCTED ON THE SUPPORT SET OF 231 GENES IDENTIFIED BY VAN'T VEER ET AL.

Method	Support set of 231 genes (van 't Veer)			
	Training set (78 cases)		Test set (19 cases)	Entire dataset (78 + 19 cases)
	Direct classification (%)	Cross-validation (%)	Direct classification (%)	Cross-validation (%)
Artificial neural networks (1 hidden layer)	100.00	72.24	73.68	73.96
Support vector machines (linear kernel)	100.00	72.79	73.68	74.88
Logistic regression	100.00	71.21	73.68	75.63
Nearest neighbors	100.00	72.94	78.94	77.15
Decision trees (C4.5)	97.44	60.70	73.68	66.64
95% CI	98.48-100.00	65.39-74.56	72.67-76.79	70.07-77.24

From these tables we can estimate the comparative average weighted accuracies of the different predictive models constructed on the 17 genes of the proposed model, and on the 70 and 231 genes selected by van't Veer *et al*. [90]. It can be seen that the 95% confidence intervals of weighted accuracy of direct classification estimated on the test set for the three predictive models that use 17, 70 and 231 genes were 59.20-77.65, 42.15-58.91 and 72.67-76.79, respectively. Clearly, we can conclude that the weighted accuracy in distinguishing patients with good and poor breast cancer prognosis is best for the model using 231 genes, is at a comparable (although slightly lower) level for the model using 17 genes, and is at a substantially lower level for the model using 70 genes.

4.2.3.2. Individual versus Collective Biomarkers

One of the important hypotheses raised by the LAD approach concerns the role played in an accurate prognostic system by those genes that have the greatest correlation with outcome. In contrast to the conventional approach, LAD aims to go beyond the straightforward goal of identifying genes with important individual contributions to distinguishing between breast cancer patients with good and poor prognosis, instead focusing on those genes that - taken as a group - have the greatest collective prognostic potential.

The breast cancer prognostic system developed in the present study confirms the hypothesis that the most accurate prognostic systems do not necessarily include only genes with strong correlations with outcome. Indeed, the 70 biomarkers used in the study by van't Veer *et al.* [90] are extracted from the pool of 231 genes that (taken individually) are most highly correlated with the outcome. On the other hand, the 17-gene support set selected by LAD includes several genes whose correlation with the outcome in absolute value is very low. The average absolute value of Pearson correlation with the outcome of the 17 individual genes in the support set of the LAD model is only 0.33. However, the average absolute value correlation with the outcome of the 40 positive and negative patterns (which can be viewed as collective biomarkers) is higher, at 0.46.

It is interesting to note that the overlap between the set of 17 genes selected by LAD and the set of the 70 genes used in the study by van't Veer *et al.* [90] consists of only four

genes (AL080059, NM_003748, NM_020974 and Contig63102_RC). Also, the overlap between the set of 17 genes and the pool of 231 genes, from which the 70 biomarkers were extracted by van't Veer *et al.*, consists of only eight genes (the four mentioned above and AB033007, AF148505, Contig42421_RC, and Contig37063_RC).

The high accuracy of the LAD model is not due to the role of the individual genes selected, but rather to the interactions among various genes in the 'collective biomarkers' represented by patterns. The concept of collective biomarkers is crucial to the LAD approach.

4.2.3.3. Contrast between Training and Test Sets

One of the most frequently used validation techniques in a model learned on a training set is to apply it to a test set, and to compare the accuracies of the model's predictions on the two sets. It is usually assumed that characteristics of the training and test sets are very similar. The accuracy of predictions obtained by LAD and other machine-learning methodologies on the test set is usually lower than that on the training set. This phenomenon can be easily explained by the fact that any such model learns the obvious and less obvious characteristics of the training set, not all of which may be represented in the test set. Surprisingly, in our analysis, the weighted accuracy on the test set (92.9%) turned out to be even higher than that estimated by cross-validation on the training set (82.5%). This suggests that a previously unrecognized, possibly substantial, difference existed between the training and the test sets. In fact, we determined that this is the case.

Indeed, it can be seen that for the set of all observations in the training set, with the exception of case number 70 (Sample 70), the intensity levels of gene NM_005839 (Ser/Arg-related nuclear matrix protein [plenty of prolines 101-like]) are consistently less than or equal to 0.19. On the other hand, on the test set the intensity levels of the same gene are consistently greater than 0.19. Therefore, it is clear that the intensity levels of gene NM_005839 distinguish completely the observations in the training set (with the exception of observation 70) from all the observations in the test set.

The above finding is made even clearer by considering patterns. It transpires that hundreds of patterns of degree 2 can be found that completely separate the training set and the test set, without any exceptions (not even for the observation 70 mentioned above).

The existence of pairs of genes that can distinguish between the training and test sets is an extremely rare situation. The existence of individual genes allowing such a distinction is clearly even more surprising. Even in datasets in which the training and test samples are collected in different laboratories, the existence of such genes or pairs of genes is highly unlikely. For instance, no such separation exists for the microarray dataset Leukemia AML-ALL studied by Golub *et al.* [37].

As an additional distinguishing characteristic of the training and test sets, let us consider the upper and the lower bounds of each variable for the 19 test cases, as shown in Table 54. It is clear that the measurements of none of the training set cases fit into the ranges of

the 17 variables in the table. Technically, this means that if we define the interval closure of a set *S* of points as being the smallest interval [*S*] of the 17-dimensional Euclidean space R^{17} spanned by the points in *S*, then the interval [test set] does not contain any of the observations included in the training set.

Table 54

INTERVAL CONTAINING ALL THE 19 CASES IN THE TEST SET AND NONE OF THE 78 CASES IN THE TRAINING SET

Gene Accession Number	AB033007	NM_001661	NM_001756	AF148505	Contig42421_RC	NM_003748	NM_020974	AL080059	AL110129	Contig15031_RC	Contig65439	Contig37063_RC	Contig41383_RC	AL049689	Contig63102_RC	Contig55574_RC	Contig38451_RC
Lower bound	-0.212	-0.227	-0.541	-0.268	-0.301	-0.295	-1.085	-0.606	-0.144	-0.282	-0.325	-0.307	-0.106	-0.219	-0.347	-0.325	-0.245
Upper bound	0.187	0.017	0.29	0.22	0.394	0.309	0.557	0.401	0.241	0.062	0.303	0.272	0.117	0.304	0.331	0.576	0.183

The observations presented above led us to the conclusion that the training and the test sets have different characteristics.

4.2.3.4. Individualized Therapy

An important consequence of the identification of genes that are promoters or blockers is the possibility of targeting therapies in such a way that they should raise the expression of some blockers and/or lower those of some promoters. An even more attractive challenge

is that of developing individualized therapies, which target the particular blockers and promoters present in the specific positive and negative patterns 'triggered' by the expression levels of an individual's genes.

4.2.4. Conclusion

In summary, the LAD-based analysis of the van't Veer data [90] identified

- a new support set of 17 genes, capable of fully distinguishing cases with poor prognosis from cases with good prognosis; the selection of the set of 17 genes took into account their collective interactive role in distinguishing cancer cases from controls (i.e., did not simply select those genes which, taken individually, have particularly high expression levels or high correlations with the outcome);
- an explicit and highly accurate classification model for breast cancer diagnosis, in which every decision is explicit and transparent, i.e., fully described by the patterns of gene expression displayed by each individual patient;
- the relative importance of each of the 17 genes, and identified those which have a blocking or contributing influence on breast cancer.

This study suggests the applicability of the nonparametric combinatorial method of LAD to genomic analysis of other human cancers, as well as to the design of individualized therapies based on the specific patterns of gene expressions for each patient.

4.3. Results for the Two Real-life Datasets Obtained with Composite Boolean Separators

In this section, we apply the CBS techniques developed in Chapters 2 and 3 to the two real-life datasets analyzed above. We show that the classification accuracy improves if a CBS with the highest *CP* is added to the data. We present new results on the attribute selection problem, and provide more evidence of the suspiciousness of the observations which were identified in the previous studies for the CT data. We also show that the special classes P^{+++} and N^{---} in the breast cancer dataset contain strongly reliable observations.

4.3.1. Computed Tomography Data

As we mentioned before, there are three types of patients in the CT data: DIP, IPF and NSIP. In Section 4.1, we constructed and validated three different models: DIP_non-DIP, IPF_non-IPF and NSIP_non-NSIP. Now we apply the iterative algorithm described in Section 2.3 for generation of CBSes in order to obtain separators for each of the problems. Below we present CBSes with the highest *CP*s and show, using four machine-learning / data-mining methods, the accuracy improvement (the error rate reduction) if only one CBS is added to the original data (see Table 55). Each entry in Table 55 is the average result obtained in twenty 3-folding experiments.

CBS with the highest *CP* for the DIP_non-DIP problem is:

$$a_3 \vee a_5 a_8 \vee a_5 a_9 \vee a_8 a_9 \vee \overline{a}_{16} \vee \overline{a}_{18};$$

for the IPF_non-IPF problem is**:**

$$\overline{a}_2 a_9 \vee \overline{a}_2 a_{15} \vee a_9 a_{16} \vee a_{15} a_{16} a_{18}\,;$$

and for the NSIP_non-NSIP problem is:

$$\overline{a}_2 \overline{a}_6 \vee \overline{a}_2 a_{11} \vee \overline{a}_6 a_7\,.$$

The *CPs* of these three CBSes are 97.96%, 90.24% and 89.35% respectively.

Table 55

AVERAGE ACCURACY ON ORIGINAL DATA AND ON ORIGINAL DATA WITH ONE CBS WITH THE HIGHEST CP

	Dataset	SMO	MP	SL	C4.5	Average	Average change in accuracy	Average error rate reduction
DIP/non-DIP	Original	60.88%	64.38%	62.30%	71.18%	64.69%	+23.84%	67.52%
	Original+CBS	92.90%	86.38%	94.35%	80.50%	88.53%		
IPF/non-IPF	Original	79.60%	79.44%	80.66%	81.27%	80.24%	+4.68%	23.68%
	Original+CBS	84.20%	83.19%	89.10%	83.21%	84.92%		
NSIP/non-NSIP	Original	66.50%	65.11%	62.53%	62.14%	64.07%	+10.13%	28.19%
	Original+CBS	63.86%	69.61%	87.88%	75.47%	74.20%		

It can be seen from Table 55 that by adding only one CBS, the average accuracy of every single classification method was improved for each of the problems and the average error rate was reduced.

In Section 4.1, we showed that observations s003 and s046 are suspicious and this was confirmed by medical doctors. A natural question to ask is whether CBSes are able to identify these observations as suspicious. Table 56 provides an answer to this question.

Table 56

CLASSIFICATION OF OBSERVATIONS S003 AND S046 BY CBSES

Observations	Given Classification	Classification by CBSes			
		IPF/non-IPF	NSIP/non-NSIP	DIP/non-DIP	Conclusion
s003	DIP	0	1	1	NSIP or DIP
s046	NSIP	1	0	0	IPF

All found CBSes for the NSIP/non-NSIP problem identify the observation s003 as an NSIP patient; additionally, all found CBSes for the DIP/non-DIP problem identify this observation s003 as a DIP patient. Therefore the patient cannot be classified.

The observation s046 is classified by all the CBSes, found for the IPF/non-IPF problem, as being an IPF patient. All found CBSes for the other two problems also confirm that this seems to be an IPF patient. This makes it suspicious with respect to class, since the original classification for this observation is NSIP.

In Section 4.1, we presented informative subsets of attributes (support sets). In this section, we compare the results obtained on these subsets with the results obtained on new subsets of attributes, constructed by the two CBS approaches given in Section 3.3.1. It should be noted that before these experiments we deleted observations s003 and s046. Each entry in Table 57 is the average result of twenty 3-folding cross-validation experiments.

Table 57

ATTRIBUTE SELECTION RESULTS FOR CT DATA

Dataset	Method	# of variables in informative subset	Average accuracy obtained by				Average accuracy of 4 methods	Difference between average accuracy of original dataset and informative subset
			SMO	MP	SL	C4.5		
DIP/non-DIP	LAD	6	60.16%	74.27%	70.83%	86.88%	73.03%	3.07%
	One_CBS	5	57.40%	72.45%	69.74%	91.72%	72.83%	2.86%
	All_CBSes	5	57.40%	72.45%	69.74%	91.72%	72.83%	2.86%
IPF/non-IPF	LAD	9	79.15%	83.66%	81.12%	84.74%	82.17%	0.44%
	One_CBS	3	79.01%	81.29%	84.47%	82.03%	81.70%	-0.03%
	All_CBSes	6	78.58%	80.95%	83.15%	82.03%	81.18%	-0.55%
NSIP/non-NSIP	LAD	8	71.88%	78.96%	72.21%	67.88%	72.73%	1.34%
	One_CBS	4	71.34%	80.97%	71.47%	70.87%	73.67%	2.27%
	All_CBSes	11	73.49%	81.84%	70.49%	66.12%	72.98%	1.59%

It can be seen that the results obtained by the CBS techniques are comparable with those obtained in Section 4.1.3.2. For the DIP/non-DIP problem, the subset from Section 4.1.3.2 provides a better accuracy (by 0.2%), but the subsets obtained with CBSes have a smaller size. For the IPF/non-IPF problem, the size of the subset obtained with *One_CBS* is three times less than the size of the support set obtained in Section 4.1.3.2, and the average accuracy on the former is only 0.4% less than on the latter. For the NSIP/non-NSIP problem, the best informative subset of attributes was also obtained by using *One_CBS* approach. The number of variables in this subset is half of the number of variables in the support set found in Section 4.1.3.2, and the average accuracy is 0.93% higher.

To conclude this section, we show the accuracy improvement (error rate reduction) on the informative subsets of attributes if one CBS with the highest *CP* is added to each of these subsets (see Table 58).

Table 58

RESULTS ON THE INFORMATIVE SUBSETS OF ATTRIBUTES WITH ONE CBS WITH THE HIGHEST CP FOR CT DATA

Dataset	Method	Average accuracy obtained by				Average accuracy of 4 methods	Difference between average accuracy of original dataset and informative subset	Average error rate reduction
		SMO	MP	SL	C4.5			
DIP/non-DIP	LAD	91.46%	90.57%	91.51%	90.52%	91.02%	21.05%	70.10%
	One_CBS	96.15%	76.41%	92.55%	96.98%	90.52%	20.56%	68.44%
	All_CBSes	96.15%	76.41%	92.55%	96.98%	90.52%	20.56%	68.44%
IPF/non-IPF	LAD	91.83%	86.96%	91.40%	90.64%	90.21%	8.48%	46.41%
	One_CBS	91.95%	90.50%	91.95%	91.59%	91.49%	9.76%	53.42%
	All_CBSes	91.95%	88.48%	91.83%	90.87%	90.78%	9.05%	49.53%
NSIP/non-NSIP	LAD	89.53%	82.58%	90.66%	89.56%	88.08%	16.69%	58.34%
	One_CBS	92.12%	86.59%	90.49%	89.95%	89.79%	18.40%	64.31%
	All_CBSes	89.62%	83.68%	91.23%	89.54%	88.51%	17.12%	59.84%

The following conclusion can be made for this section.

Conclusion 13. *The CBSes obtained for the problems DIP/non-DIP, IPF/non-IPF and NSIP/non-NSIP have simple formulas and high CPs. The CBS technique for identifying suspicious observations confirms the suspiciousness of the observations s003 and s046. The subsets of attributes obtained by using the CBS approaches have small sizes. The classification accuracies obtained on them are comparable with the accuracies obtained with all original variables and are even 2% higher for the DIP/non-DIP and NSIP/non-*

NSIP problems. Removing the two suspicious observations and the redundant variables from the data and adding one CBS with the highest CP substantially improve the average classification accuracies (substantially reduce error rates). These accuracy improvements for the three considered problems are at least 9% and at most 21% (the error rate reductions are at least 46% and at most 70%).

4.3.2. Breast Cancer Gene Expression Microarray Data

In this section, we use CBSes to analyze the breast cancer gene expression microarray dataset. We apply them in two different ways. First, we use CBSes to find smaller subsets of variables that adequately describe the data. Second, we analyze how the addition of a CBS with the highest *CP* to the chosen subsets affects the classification accuracy. In our analysis we use the same partitioning of the data into training set and test set as was used in Section 4.2.

Since the number of variables for this dataset is very large (25,000 genes), we cannot apply the iterative procedure for finding CBSes directly to the original dataset. Instead, we apply it to the support set of 17 variables obtained in Section 4.2. As a result, we obtain 21 CBSes. The separator with the highest *CP,* denoted by *f*, is very complicated, so we present below its negation which has a simpler formula. *CP* of *f* is 95.92% on the training set, 85.71% on the test set and 90.82% on the entire data.

$$\bar{f} = a_7 a_{12} a_{13} a_{22} \vee a_7 a_{13} a_{14} a_{22} \vee a_7 a_{13} \bar{a}_{16} a_{17} \vee a_4 a_{12} \vee a_{11} a_{12} a_{27} \vee a_4 \bar{a}_{16} a_{17} \vee$$

$$\bar{a}_5 a_{11} a_{33} \vee \bar{a}_5 a_{12} \vee \bar{a}_5 a_{14} \vee a_{14} \bar{a}_{16} a_{17}$$

Using f and the union of all obtained separators, we identify two informative sets of variables, which are subsets of the support set of 17 variables. The first of them contains 10 variables and will be denoted by S_{10}. The second subset contains 13 variables and will be denoted by S_{13}. For notational consistency, we shall denote the support set of 17 variables by S_{17}. Table 59 and Table 60 show the classification accuracy obtained on these three subsets by the five classification methods. Each entry in Table 60 is the result of twenty 5-folding experiments.

Table 59

ATTRIBUTE SELECTION RESULTS FOR GENE EXPRESSION MICROARRAY DATA

DIRECT CLASSIFICATION

Subset		Average accuracy obtained by					Average accuracy of 5 methods
		SMO	MP	SL	C4.5	LAD	
S_{17}	Tra	86.95%	100.00%	94.80%	95.90%	100.00%	95.53%
	Test	61.90%	81.55%	70.20%	57.70%	92.86%	72.84%
S_{10}	Tra	82.90%	100.00%	88.10%	98.85%	100.00%	93.97%
	Test	81.55%	74.40%	67.30%	77.35%	85.71%	77.26%
S_{13}	Tra	87.80%	97.75%	92.50%	97.40%	100.00%	95.09%
	Test	69.05%	74.40%	70.20%	43.45%	85.71%	68.56%

Table 60

ATTRIBUTE SELECTION RESULTS FOR GENE EXPRESSION MICROARRAY DATA

CROSS-VALIDATION

Subset		Average accuracy obtained by					Average accuracy of 5 methods
		SMO	MP	SL	C4.5	LAD	
S_{17}	Tra	76.43%	76.55%	76.87%	67.48%	82.52%	75.97%
	Tra+Test	77.27%	78.65%	77.95%	67.01%	81.74%	76.52%
S_{10}	Tra	75.23%	77.17%	78.39%	68.96%	82.09%	76.37%
	Tra+Test	75.52%	76.37%	75.50%	70.25%	78.96%	75.32%
S_{13}	Tra	72.68%	76.44%	77.89%	68.40%	81.79%	75.44%
	Tra+Test	74.30%	76.03%	73.84%	69.21%	79.59%	74.60%

It can be seen from the above tables that the subset of variables obtained by *One_CBS* has the smallest size. The cross-validation results obtained on this subset of 10 variables are very close to the results obtained on the support set of 17 variables (on average the difference is within 1%). It also can be seen that the direct application of the models obtained on the subset of 10 variables to the training set decreases the classification accuracy for the SMO and Simple Logistic methods and increases it for C4.5. The application of the models to the test set decreases the classification accuracy for Multilayer Perceptron, Simple Logistic and LAD by minimum 3% and maximum 7%, and increases accuracy for SMO and C4.5 by approximately 20%. The new LAD model (see Table 61) obtained on 10 variables results in 100% correct prediction on the training set and in 85.7% correct prediction on the test set (17 cases out of 19 are correctly predicted and two negative cases are predicted as positive). This result is slightly worse than that obtained on 17 variables (only one case in test set was predicted as positive), but it is better than the results reported by van't Veer and coworkers [90]. Moreover, the new model consists of only 6 positive and 8 negative patterns (recall that the model reported in Section 4.2 consists of 20 positive and 20 negative patterns). All patterns are pure and only two cutpoints are introduced into the range of intensities of each gene.

Table 61

LAD MODEL ON THE INFORMATIVE SUBSET OF 10 GENES

Pattern	NM_001661	NM_001756	AF148505	NM_003748	NM_020974	AL080059	AL110129	Contig65439	AL049689	Contig38451_RC	Patterns' prevalences on training set	
	Attr.1	Attr.2	Attr.3	Attr.4	Attr.5	Attr.6	Attr.7	Attr.8	Attr.9	Attr.10	Pos Prev	Neg Prev
P1		>-0.455		≤0.1095		>-0.1245					18 (52.9%)	0
P2				≤0.1095	≤0.086		≤0.02				17 (50.0%)	0
P3		>-0.455					≤0.02	≤0.044			15 (44.1%)	0
P4				≤0.1095				≤-0.066		≤0.142	15 (44.1%)	0
P5	≤-0.02					>-0.3295	≤0.157				14 (41.2%)	0
P6		>-0.275	≤0.095			>-0.3295					11 (32.4%)	0
N1		≤-0.455								>0.037	0	19 (43.2%)
N2		≤-0.455		>0.1095							0	16 (36.4%)
N3						≤-0.3295		>0.044			0	16 (36.4%)
N4					>0.086				>-0.016		0	16 (36.4%)
N5	>-0.02		>0.095						>-0.016		0	15 (34.1%)
N6	>0.086							>-0.066		>0.037	0	15 (34.1%)
N7	>-0.02		>0.095	>0.1095							0	13 (29.5%)
N8		≤-0.275					>0.157		>-0.016		0	12 (27.3%)

Now let us analyze how the addition of CBSes affects the classification accuracy. We add the CBS *f* with the highest *CP* to each of the three sets S_{10}, S_{13}, S_{17} obtaining in this way three extended sets S'_{10}, S'_{13}, S'_{17}. To each of these sets we apply the five classification methods used in this study. An interesting phenomenon revealed in these experiments is that the model built by LAD on the extended set of variables depends only on part of these variables. Therefore, since the classification accuracy does not depend on the presence of the other part, we can disregard those redundant variables. Removing the redundant variables from each of the sets S'_{10}, S'_{13}, S'_{17} results in three new sets S''_{10},

S_{13}'', S_{17}'', each containing nine variables. Moreover, S_{13}'' and S_{17}'' coincide and we report classification results for only one of them. The intersection of S_{10}'' with S_{17}'' contains six variables. The new classification results are presented in Table 62 and Table 63.

Table 62

RESULTS ON THE INFORMATIVE SUBSETS OF VARIABLES WITH ONE CBS WITH THE HIGHEST CP FOR GENE EXPRESSION MICROARRAY DATA

DIRECT CLASSIFICATION

Subset		Average accuracy obtained by					Average accuracy of 5 methods
		SMO	MP	SL	C4.5	LAD	
S_{17}	Tra	95.90%	98.55%	95.90%	95.90%	100.00%	97.25%
	Test	85.70%	73.20%	85.70%	85.70%	85.71%	83.20%
S_{10}	Tra	95.90%	98.55%	95.90%	95.90%	100.00%	97.25%
	Test	85.70%	85.70%	85.70%	85.70%	85.71%	85.70%

Table 63

RESULTS ON THE INFORMATIVE SUBSETS OF VARIABLES WITH ONE CBS WITH THE HIGHEST CP FOR GENE EXPRESSION MICROARRAY DATA

CROSS-VALIDATION

Subset		Average accuracy obtained by					Average accuracy of 5 methods
		SMO	MP	SL	C4.5	LAD	
S_{17}	Tra	95.96%	91.16%	95.59%	93.73%	93.31%	93.95%
	Tra+Test	94.93%	89.60%	94.88%	93.75%	89.80%	92.59%
S_{10}	Tra	95.96%	90.81%	95.61%	93.73%	92.62%	93.75%
	Tra+Test	94.93%	89.87%	94.88%	93.75%	89.00%	92.49%

Comparing Table 62 with Table 59 and Table 63 with Table 60 and adding the results for error rate reduction we obtain the following tables.

Table 64

AVERAGE CHANGE IN ACCURACY AND AVERAGE ERROR RATE REDUCTION FOR GENE EXPRESSION MICROARRAY DATA

DIRECT CLASSIFICATION

Subset		Average change in accuracy	Average error rate reduction
S_{17}	Tra	1.72%	38.48%
	Test	10.36%	38.14%
S_{10}	Tra	3.28%	54.39%
	Test	8.44%	37.12%

Table 65

AVERAGE CHANGE IN ACCURACY AND AVERAGE ERROR RATE REDUCTION FOR GENE EXPRESSION MICROARRAY DATA

CROSS-VALIDATION

Subset		Average change in accuracy	Average error rate reduction
S_{17}	Tra	17.98%	74.82%
	Tra+Test	16.07%	68.44%
S_{10}	Tra	17.38%	73.55%
	Tra+Test	17.17%	69.57%

From the above tables we can conclude that:

Conclusion 14. *The addition of one CBS significantly improves classification accuracy (significantly reduces error rate) for each machine-learning / data-mining method used.*

We conclude this section with the demonstration of one more useful aspect of CBSes. In Section 4.2.2 we showed that there are special classes of positive and negative cases for the given dataset. It turned out that the patients who belong to class P^{+++} are strongly reliable in the sense defined in Section 3.2.1, i.e., they are classified correctly by all the CBSes. Analogous results are obtained for the negative cases which belong to class N^{---}.

Conclusion 15. *CBSes confirm that the patients belonging to class P^{+++} have a very strong tendency toward developing metastases and patients belonging to class N^{---} are very strongly resistant to development of metastases.*

5. CONCLUSION

In this thesis, we introduced the concept of composite Boolean separators and demonstrated in various ways the usefulness of the new concept for data analysis. In particular,

- we demonstrated how the introduction of CBSes can enhance the accuracy of classification systems; CBSes also proved to be a promising tool as classification systems themselves;
- we employed CBSes for identifying misclassified observations and examined how deletion of such observations and reversal of their class influence the classification accuracy; the obtained results showed high effectiveness of the proposed technique, since it reduces the error rate more than in half;
- we applied CBSes to the attribute selection problem and demonstrated that the CBS based methods allow to identify small subsets of attributes that provide as much information for determining the class of the observations in the dataset as the original set of attributes.

All the results have been tested on eight publicly available datasets and validated by five well-known machine-learning / data-mining techniques. We also applied CBSes, along with other techniques, to analyze two real-life medical datasets: computed tomography data and breast cancer gene expression microarray data.

The results demonstrated in this thesis showed that for many real-life datasets, CBSes

have noticeable advantages over other techniques (higher classification accuracy, smaller informative subsets of attributes identified, etc.). For some data, CBSes do not provide improvements, though show results comparable with other techniques, verifying the so-called *No Free Lunch Theorem*:

> "All algorithms are equivalent, on average. Or to put it another way, for any two learning algorithms, there are just as many situations in which algorithm one is superior to algorithm two as vice versa" (D.H. Wolpert [95]).

We hope that along with other techniques the CBSes will be a useful tool in the area of machine-learning / data-mining.

The usefulness of CBSes has been already confirmed by a practical application. Richard Hoshino (Senior Project Officer, Canada Border Services Agency, Government of Canada) in his talk given at DIMACS [32] reported the application of composite Boolean separators to Marine Container Security. We hope that in the future this concept will find many other applications.

REFERENCES

[1] S. Abramson, G. Alexe, P.L. Hammer, D. Knight, J. Kohn, "Using logical analysis of data (LAD) based modeling to understand patterns of physio-mechanical data which lead to specific cellular outcomes", *J Biomed Materials Res A*, 73, pp. 116-24, 2005.

[2] D.W. Aha, "Incremental constructive induction: An instance-based approach", *In Proc. 8th International Conference on Machine Learning*, pp. 117-121, 1991.

[3] G. Alexe, S. Alexe, P.L. Hammer, L. Liotta, E. Petricoin, M. Reiss, "Logical Analysis of Proteomic Ovarian Cancer Dataset", *Proteomics*, 4, pp. 766-783, 2004.

[4] G. Alexe, S. Alexe, D.E. Axelrod, D. Weissmann, P.L. Hammer, "Logical analysis of diffuse large B-cell lymphomas", *Artif Intell Med*, 34, pp. 235-267, 2005.

[5] G. Alexe, S. Alexe, D.E. Axelrod, I.I. Lozina, M. Reiss, and P.L. Hammer, "Breast Cancer Prognosis by Combinatorial Analysis of Gene Expression Data", *Breast Cancer Research*, Vol.8 N.4 R41, 2006. (Available online at http://breast-cancer-research.com/content/8/4/R41)

[6] S. Alexe, E. Blackstone, P.L. Hammer, H. Ishwaran, M.S. Lauer, C.E.P. Snader, "Coronary Risk Prediction by Logical Analysis of Data", *Annals of Operations Research,* 119, pp. 15-42, 2003.

[7] G. Alexe, P.L. Hammer, "Spanned patterns in logical analysis of data", *Discr Appl Math*, 154, pp. 1039-1049, 2006.

[8] G. Alexe, S. Alexe, P.L. Hammer, B. Vizvari, "Pattern-based feature selection in genomics and proteomics", *Ann Oper Res* 2006:in press.

[9] A.A. Alizadeh, M.B. Eisen, R.E. Davis, C. Ma, I.S. Lossos, A. Rosenwald, J.C. Boldrick, H. Sabet, T. Tran, X. Yu, *et al*., "Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling", *Nature*, 403, pp. 503-511, 2000.

[10] O. Alter, P.O. Brown, D. Botstein, "Singular value decomposition for genome-wide expression data processing and modeling", *Proc Natl Acad Sci USA*, 97, pp. 10101-10106, 2000.

[11] O. Alter, P.O. Brown, D. Botstein, "Generalized singular value composition for comparative analysis of genome-scale expression data sets of two different organisms", *Proc Natl Acad Sci USA*, 100, pp. 3351-3356, 2003.

[12] American Thoracic Society / European Respiratory Society International Multidisciplinary Consensus. "Classification of the Idiopathic Interstitial Pneumonias", *Amer J Respir Crit Care Med*, 165, 2002, pp. 277-304, 2002.

[13] P. Auer and N. Cesa-Bianchi, "On-line learning with malicious noise and the closure algorithm", Annals of mathematics and artificial intelligence, 23, pp. 83-99, 1998.

[14] M. Bittner, P. Meltzer, Y. Chen, Y. Jiang, E. Seftor, M. Hendrix, M. Radmacher, R. Simon, Z. Yakhini, A. Ben-Dor, *et al*., "Molecular classification of cutaneous malignant melanoma by expression profiling", *Nature*, 406, pp. 536-540, 2000.

[15] A. Blum and P. Langley, "Selection of relevant features and examples in machine learning", *Artificial Intelligence*, 97, pp. 245-271, 1997.

[16] M. Bongrad , "Pattern recognition", *Spartan books*, 1970.

[17] E. Boros, P.L. Hammer, T. Ibaraki, A. Kogan, "Logical Analysis of Numerical Data", *Mathematical Programming*, 79, pp.163-190, 1997.

[18] E. Boros, P.L. Hammer, T. Ibaraki, A. Kogan, E. Mayoraz, I. Muchnik, "An Implementation of the Logical Analysis of Data", *IEEE Transactions on Knowledge and Data Engineering,* 12, No.2, pp.292-306, 2000.

[19] A.L. Boulesteix, G. Tutz, K.A. Strimmer, "CART-based approach to discover emerging patterns in microarray data", *Bioinformatics*, 19, pp. 2465-2472, 2003.

[20] M.W. Brauner, N. Brauner, P.L. Hammer, I. Lozina, D. Valeyre, "Logical analysis of computed tomography data to differentiate entities of idiopathic interstitial pneumonias", In *Data Mining in Biomedicine*. P. Pardalos, V. Boginski and A. Vazacopoulos (Eds.), Springer, pp. 193-208, 2007.

[21] C.E. Brodley, M.A. Friedl, "Identifying and eliminating mislabeled training instances", *Proc. of 13th National conf. on artificial intelligence*, pp. 799-805, 1996.

[22] C.E. Brodley, M.A. Friedl, "Identifying mislabeled training data", *Journal of Artificial Intelligence research*, 11, pp. 131-167, 1999.

[23] M.P.S. Brown, W.N. Grundy, D. Lin, N. Cristanini, C.W. Sugnet, T.S.Furey, M. Jr Ares, D. Haussler, "Knowledge-based analysis of microarray gene expression data by using support vector machines", *Proc Natl Acad Sci USA,* 97:262-267, 2000.

[24] I. Cantador, J.R. Dorronsoro, "Boosting parallel perceptrons for label noise reduction in classification problems", IWINAC 2005, *LNCS* 3562, pp.586-593, 2005.

[25] J.J. Chen, K. Peck, T.M. Hong, S.C. Yang, Y.P. Sher, J.Y. Shih, R. Wu, J.L. Cheng, S.R. Roffler, C.W. Wu, *et al*., "Global analysis of gene expression in invasion by a lung cancer model", *Cancer Res*, 61, pp. 5223-5230, 2001.

[26] D.R. Cox, "The Analysis of Binary Data", Methuen, London, 1970.

[27] Y. Crama, P.L. Hammer, and T. Ibaraki, "Cause-Effect Relationships and Partially Defined Boolean Functions", *Annals of Operations Research*, 16, pp.299-326, 1988.

[28] Z. Csizmadia, P.L. Hammer, and B. Vizvari, "Generation of Artificial Attributes for Data Analysis", *RUTCOR Research Report*, 1-2006.

[29] M. Dash and H.Liu, "Feature selection for classification", *Intelligent Data Analysis*, 1, pp. 131-156, 1997.

[30] DAVID (Database for Annotation, Visualization and Integrated Discovery) [http://apps1.niaid.nih.gov/david]

[31] T.G. Dietterich, "Fundamental Experimental Research in Machine Learning", A section of the document *Basic Topics in Experimental Computer Science* edited by John McCarthy, 1997, [http://web.engr.oregonstate.edu/~tgd/projects/tutorials.html].

[32] http://dimacs.rutgers.edu/Events/2006/abstracts/hoshino.html

[33] S.T. Dumais, E.Osuna, J.Platt, and B. Scholkopf, "Support vector machines", *IEEE Intelligent Systems*, pp. 18-28, 1998.

[34] M.B. Eisen, P.T. Spellman, P.O. Brown, D. Botstein, "Cluster analysis and display of genome-wide expression patterns", *Proc Natl Acad Sci USA*, 95, pp. 14863-14868, 1998.

[35] T.S. Furey, N. Cristianini, N. Duffy, D.W. Bednarski, M. Schummer, D. Haussler, "Support vector machine classification and validation of cancer tissue samples using microarray expression data", *Bioinformatics*, 16, pp. 906-914, 2000.

[36] G. Getz, E. Levine, E. Domany, "Coupled two-way clustering analysis of gene microarray data", *Proc Natl Acad Sci USA*, 97, pp. 12079-12084, 2000.

[37] T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, H. Coller, M.L. Loh, J.R. Downing, M.A. Caligiuri, *et al*., "Molecular classification of cancer: class discovery and class prediction by gene expression monitoring", *Science*, 286, pp. 531-537, 1999.

[38] S. Gruvberger, M. Ringner, Y. Chen, S. Panavally, L.H. Saal, A. Borg, M. Ferno, C. Peterson, P.S. Meltzer, "Estrogen receptor status in breast cancer is associated with remarkably distinct gene expression patterns", C*ancer Res*, 61, pp. 5979-5984, 2001.

[39] M.A. Hall, "Correlation-based feature selection for discrete and numeric class machine learning", *Proceedings of the 17th International Conference on Machine Learning* (*ICML 2000*), pp. 359-366, 2000.

[40] M.A. Hall and G. Holmes, "Benchmarking attribute selection techniques for discrete class data mining", *IEEE Transactions on Knowledge and Data Engineering*, 15, pp. 1437-1447, 2003.

[41] P.L. Hammer, "Partially Defined Boolean Functions and Cause-Effect Relationships", *International Conference on Multi-Attribute Decision Making Via OR-Based Expert Systems*, University of Passau, Passau, Germany, 1986.

[42] P. L. Hammer and T. Bonates, "Logical Analysis of Data: From Combinatorial Optimization to Medical Applications", *Annals of Operations Research* 2006:in press (http://rutcor.rutgers.edu/~rrr/2005.html)

[43] P. L. Hammer and I. I. Lozina, "Boolean Separators and Approximate Boolean Classifiers", *RUTCOR Research Report*, RRR 14-2006. (http://rutcor.rutgers.edu/pub/rrr/reports2006/14_2006.pdf)

[44] P. L. Hammer and I. I. Lozina, "Composite Boolean Separators for Data Analysis", *RUTCOR Research Report*, RRR 14-2007. (http://rutcor.rutgers.edu/pub/rrr/reports2007/14_2007.pdf)

[45] A. Hammer, P. L. Hammer, I. Muchnik, "Logical analysis of Chinese productivity patterns", *Ann Oper Res*, 87, pp. 165-176, 1999.

[46] P. L. Hammer, A. Kogan, M.A. Lejeune, "Country risk rating: statistical and combinatorial non-recursive models", *RUTCOR Research Report*, RRR 8-2004

[47] K. Haraguchi, T. Ibaraki, and E. Boros, "Classifiers based on iterative compositions of features", *Proc. 1st Intl. Conf. Knowledge Engineering and Decision Support*, pp.143-150, Porto, Portugal, Aug. 2004.

[48] T.E. Hartman, S.J. Swensen, D.M. Hansell, T.V. Colby, J.L. Myers, H.D. Tazelaar, A.G. Nicholson, A.U. Wells, J.H. Ryu, D.E. Midthun, R.M. du Bois, N.L. Muller, "Nonspecific Interstitial Pneumonia: Variable Appearance at High-Resolution Chest CT", *Radiology*, 217(3), pp. 701-705, 2000.

[49] T. Hastie, R. Tibshirani, M.B. Eisen, A. Alizadeh, R. Levy, L. Staudt, W.C. Chan, D. Botstein, P. Brown, " 'Gene shaving' as a method for identifying distinct sets of genes with similar expression patterns", *Genome Biol*, 1, pp. 1-21, 2000.

[50] S.G. Hilsenbeck, W.E. Friedrichs, R. Schiff, P. O'Connell, R.K. Hansen, C.K. Osborne, S.A. Fuqua, "Statistical analysis of array expression data as applied to the problem of tamoxifen resistance", *J Natl Cancer Inst*, 91, pp. 453-459, 1999.

[51] N.S. Holter, M. Mitra, A. Maritan, M. Cieplak, J.R. Banavar, N.V. Fedoroff, "Fundamental patterns underlying gene expression profiles: simplicity from complexity", *Proc Natl Acad Sci USA*, 97, pp. 8409-8414, 2000.

[52] D.W. Hosmer, S. Lemeshow, "Applied logistic regression", *John Willey & Sons*, 1989.

[53] *http://www.ics.uci.edu/~mlearn/ MLRepository.html*

[54] *http://rutcor.rutgers.edu/~salexe/LAD_kit/SETUP-LAD-DS-SE20.zip*

[55] Y. Hu and D. Kibler, "Generation of Attributes for Learning Algorithms", *in Proceeding of the 13th National Conference on Artificial Intelligence*, pp.806-811, 1996.

[56] X. Huang, W. Pan, "Linear regression and two-class classification with gene expression data", *Bioinformatics*, 19, pp. 2072-2078, 2003.

[57] A.M. Jackson, A.V. Ivshina, O. Senko, A. Kuznetsova, A. Sundan, M.A. O'Donnell, S. Clinton, A.B. Alexandroff, P.J. Selby, K. James, *et al*., "Prognosis of intravesical bacillus Calmette-Guerin therapy for superficial bladder cancer by immunological urinary measurements: statistically weighted syndromes analysis", *J Urol*, 159, pp. 1054-1063, 1998.

[58] T. Johkoh, N.L. Muller, Y. Cartier, P.V. Kavanagh, T.E. Hartman, M. Akira, K. Ichikado, M. Ando, H. Nakamura, "Idiopathic Interstitial Pneumonias: Diagnostic Accuracy of Thin-Section CT in 129 Patients", *Radiology*, 211(2), pp. 555-560, 1999.

[59] J. Khan, J.S. Wei, M. Ringner, L.H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C.R. Antonescu, C. Peterson, *et al*., "Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks", *Nature Med*, 7, pp. 673-679, 2001.

[60] J. Khan, R. Simon, M. Bittner, Y. Chen, S.B. Leighton, T. Pohida, P.D. Smith, Y. Jiang, G.C. Gooden, J.M. Trent, P.S. Meltzer, "Gene expression profiling of alveolar rhabdomyosarcoma with cDNA microarrays", *Cancer Res*, 58, pp. 5009-5013, 1998.

[61] V.A. Kuznetsov, A.V. Ivshina, O.V. Sen'ko, A.V. Kuznetsova, "Syndrome approach for computer recognition of fuzzy systems and its application to immunological diagnostics and prognosis of human cancer", *Math Comp Modelling*, 23, pp. 95-120, 1996.

[62] N. Landwehr, M. Hall, E. Frank, "Logistic Model Trees", *Machnie Learning*, 59, pp. 161-205, 2005

[63] M.S. Lauer, S. Alexe, C.E.P. Snader, E. Blackstone, H. Ishwaran, P.L. Hammer, "Use of the "Logical Analysis of Data" Method for Assessing Long-Term Mortality Risk After Exercise Electrocardiography", *Circulation,* 106, pp. 685-690, 2002.

[64] L. Liu, D.M. Hawkins, S. Ghosh, S.S. Young, "Robust singular value decomposition analysis of microarray data", *Proc Natl Acad Sci USA*, 100, pp. 13167-13172, 2003.

[65] H. Liu and R. Setiono, "A probabilistic approach to feature selection – a filter solution", *In Machine learning, Proc. of the 13th International Conference, Bari, Italy*, pp.319-327, 1996.

[66] S. Markovitch and D. Rosenstein, "Feature Generation Using General Constructor Functions", *Machine Learning*, Volume 49, No.1, pp.59-98, 2002.

[67] C. J. Matheus and L. A. Rendell, "Constructive Induction On Decision Trees", *In Proceeding of the 11th International Joint Conference on Artificial Intelligence*, pp.645-650, 1989.

[68] W. McCulloch, and W. Pitts, "A logical calculus of the ideas immanent in nervous activity", *Bulletin of Mathematical Biophysics*, 7, pp. 115 – 133, 1943.

[69] R.A. McDonald, D.J. Hand, and I.A. Eckley, "An empirical comparison of three boosting algorithms on real data sets with artificial class noise", MCS 2003, *LNCS* 2709, pp. 35-44, 2003.

[70] M. Minsky, and S. Papert, "Perceptrons: An Introduction to Computational Geometry", *MIT Press, Cambridge, MA*, 1969.

[71] F. Muhlenbach, S. Lallich, D. A. Zighed, "Identifying and Handling Mislabelled Instances", *Journal of Intelligent Information Systems*, 22, 89–109, pp. 2004.

[72] G. Pagallo and D. Haussler, "Boolean feature discovery in empirical learning", Machine *Learning*, 5, pp.71-99 , 1990.

[73] J.R. Quinlan, "Induction of Decision Trees", *Machine Learning*, 1, pp.81-106, 1986.

[74] J.R. Quinlan, "C4.5: Programs for Machine Learning", *San Mateo, California: Morgan Kaufmann Publishers*, 1993.

[75] H. Ragavan, L. Rendell, M. Shaw, A. Tessmer, "Complex Concept Acquisition through Directed Search and Feature Caching", *In Proceeding of the 13th International Joint Conference on Artificial Intelligence*, pp.946-951, 1993.

[76] S. Ramaswamy, P. Tamayo, R. Rifkin, S. Mukherjee, C.H. Yeang, M. Angelo, C. Ladd, M. Reich, E. Latulippe, J.P. Mesirov, *et al*., "Multiclass cancer diagnosis using tumor gene expression signatures", *Proc Natl Acad Sci USA*, 98, pp. 15149-15154, 2001.

[77] S. Raychaudhuri, J.M. Stuart, R.B. Altman, "Principle components analysis to summarize microarray experiments: application to sporulation time series", *Pacific Symp Biocomputing*, 5, pp. 452-463, 2000.

[78] F. Rosenblatt, "The perceptron: a theory of statistical separability in cognitive systems (Project Para)", *Buffalo, N.Y. : Cornell Aeronautical Laboratory*, 1958.

[79] F. Rosenblatt, "A comparison of several perceptron models", In *Self-Organizing Systems*, Spartan Books, Washington, DC, 1962.

[80] D. Rumelhart, G. Hinton, and R. Williams, "Learning internal representations by error propagation", In *Parallel Distributed Processing*, MIT Press, Cambridge, MA, pp. 318-362, 1986.

[81] B. Scholkopf and A.J. Smola, "Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond", *MIT Press Cambridge*, MA, USA, 2001.

[82] J, Shawe-Taylor, N. Cristianini, "Kernel Methods for Pattern Analysis", *Cambridge University Press*, 2004.

[83] A. Smola, P. Bartlett, B. Scholkopf, D. Schuurmans, "Advances in Large-margin Classifiers", *The MIT Press*, 2000.

[84] A.I. Su, J.B. Welsh, L.M. Sapinoso, S.G. Kern, P. Dimitrov, H. Lapp, P.G. Schultz, S.M. Powell, C.A. Moskaluk, H.F. Jr Frierson, *et al*., "Molecular classification of human carcinomas by use of gene expression signatures", *Cancer Res*, 61, pp. 7388-7393, 2001.

[85] T.R. Sutter, X.R. He, P. Dimitrov, L. Xu, G. Narasimhan, E.O. George, C.H. Sutter, C. Grubbs, R. Savory, M. Stephan-Gueldner, *et al*., "Multiple comparisons model-based clustering and ternary pattern tree numerical display of gene response to treatment: procedure and application to the preclinical evaluation of chemopreventive agents", *Mol Cancer Ther*, 1, pp. 1283-1292, 2002.

[86] P. Tamayo, D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E.S. Lander, T.R. Golub, "Interpreting patterns of gene expression with self-organizing

maps: Methods and application to hematopoietic differentiation", *Proc Natl Acad Sci USA*, 96, pp. 2907-2912, 1999.

[87] Y. Tan, L. Shi, W. Tong, C. Wang, "Multi-class cancer classification by total principal component regression (TPCR) using microarray gene expression data", *Nucleic Acids Res*, 33, pp. 56-65, 2005.

[88] P. Toronen, M. Kolehmainen, G. Wong, E. Castren, "Analysis of gene expression data using self-organizing maps", *FEBS Lett*, 451, pp. 142-146, 1999.

[89] J. Truett, J. Cornfield, W. Kannel, "A multivariate analysis of the risk of coronary heart disease in Framingham", *Journal of Chronic Diseases*, 20, pp. 511-524, 1967.

[90] L.J. van 't Veer, H. Dai, M.J. van de Vijver, Y.D. He, A.M. Hart, M. Mao, H.L. Peterse, K. van der Kooy, M.J. Marton, A.T. Witteveen, *et al*.: "Gene expression profiling predicts clinical outcome of breast cancer", *Nature*, 415, pp. 530-535, 2002.

[91] M.J. van de Vijver, Y.D. He, L.J. van 't Veer, H. Dai, A.M. Hart, D.W. Voskuil, G.J. Schreiber, J.L. Peterse, C. Roberts, M.J. Marton, *et al*., "A gene-expression signature as a predictor of survival in breast cancer", *N Engl J Med*, 347, pp. 1999-2009, 2002.

[92] V. Vapnik, "Statistical Learning Theory", Springer, N.Y., 1998

[93] S. Venkataraman, D. Metaxas, D. Fradkin, C. Kulikowski, I.Muchnik, "Distinguishing Mislabeled Data from Correctly Labeled Data in Classifier Design", *Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2004)*, pp. 668-672, 2004.

[94] S. Verbaeten and A. Van Assche, "Ensemble methods for noise elimination in classification problems", MCS 2003, *LNCS* 2709, pp. 317-325, 2003.

[95] D.H. Wolpert, "The Supervised Learning No-Free-Lunch Theorems", *In Proc. 6th Online World Conference on Soft Computing in Industrial Applications*, 2001.

[96] I.H. Witten, E. Frank, "Data Mining: Practical machine learning tools and techniques", *2nd Edition, Morgan Kaufmann*, San Francisco, 2005.

[97] X. Zeng, T.R. Martinez, "An algorithm for correcting mislabeled data", *Intelligent data analysis*, 5, pp. 491-502, 2001.

[98] H. Zhang, C-Y Yu, B. Singer, M. Xiong, "Recursive partitioning for tumor classification with gene expression microarray data", *Proc Natl Acad Sci USA* 2001, 98, pp. 6730-6735.

[99] H. Zhang, C-Y Yu, B. Singer, "Cell and tumor classification using gene expression data: construction of forests", *Proc Natl Acad Sci USA*, 100, pp. 4168-4172, 2003.

[100] X. Zhu, X. Wu, "Class noise vs. attribute noise: a quantitative study of their impacts", *Artificial intelligence review*, 22, pp.177-210, 2004.

[101] X. Zhu, X. Wu, Q. Chen, "Bridging local and global data cleansing: Identifying class noise in large, distributed data datasets", *Data mining and Knowledge discovery,*12, pp.275-308, 2006.

APPENDIX

Table A

BUPA LIVER-DISORDERS

Binary variable	Original variable	Cutpoint
a_1	mcv	87
a_2	mcv	89
a_3	mcv	90
a_4	mcv	92
a_5	alkphos	51
a_6	alkphos	65
a_7	alkphos	77
a_8	alkphos	84.5
a_9	sgpt	16
a_{10}	sgpt	17
a_{11}	sgpt	19
a_{12}	sgpt	21
a_{13}	sgpt	23
a_{14}	sgpt	26
a_{15}	sgpt	39
a_{16}	sgpt	48
a_{17}	sgot	19
a_{18}	sgot	20

a_{19}	sgot	22
a_{20}	sgot	24
a_{21}	sgot	44
a_{22}	gammagt	7
a_{23}	gammagt	20
a_{24}	gammagt	29
a_{25}	gammagt	36
a_{26}	gammagt	116
a_{27}	drinks	3
a_{28}	drinks	5
a_{29}	drinks	12

Table B

GERMAN CREDIT

Binary variable	Original variable	Cutpoint
a_1	Attribute 1	1.5
a_2	Attribute 1	2.5
a_3	Attribute 1	3
a_4	Attribute 2	13.5
a_5	Attribute 2	15.5
a_6	Attribute 2	18
a_7	Attribute 2	21
a_8	Attribute 2	24

a_9	Attribute 2	30
a_{10}	Attribute 3	2
a_{11}	Attribute 3	2.5
a_{12}	Attribute 3	3
a_{13}	Attribute 4	14
a_{14}	Attribute 4	19
a_{15}	Attribute 4	24
a_{16}	Attribute 4	30.5
a_{17}	Attribute 4	40
a_{18}	Attribute 4	53
a_{19}	Attribute 5	1.5
a_{20}	Attribute 5	2
a_{21}	Attribute 5	2.5
a_{22}	Attribute 5	3
a_{23}	Attribute 6	2.5
a_{24}	Attribute 6	3
a_{25}	Attribute 6	3.5
a_{26}	Attribute 6	4
a_{27}	Attribute 7	2.5
a_{28}	Attribute 7	3.5
a_{29}	Attribute 8	2
a_{30}	Attribute 8	2.5
a_{31}	Attribute 8	3

a_{32}	Attribute 8	3.5
a_{33}	Attribute 9	1.5
a_{34}	Attribute 9	2
a_{35}	Attribute 9	2.5
a_{36}	Attribute 9	3
a_{37}	Attribute 10	27
a_{38}	Attribute 10	30
a_{39}	Attribute 10	33
a_{40}	Attribute 10	36
a_{41}	Attribute 10	39.5
a_{42}	Attribute 10	44.5
a_{43}	Attribute 11	2
a_{44}	Attribute 11	2.5
a_{45}	Attribute 12	1.5
a_{46}	Attribute 13	1.5
a_{47}	Attribute 14	1.5
a_{48}	Attribute 15	1.5
a_{49}	Attribute 16	0.5
a_{50}	Attribute 17	0.5
a_{51}	Attribute 18	0.5
a_{52}	Attribute 19	0.5
a_{53}	Attribute 20	0.5
a_{54}	Attribute 21	0.5

a_{55}	Attribute 22	0.5
a_{56}	Attribute 23	0.5
a_{57}	Attribute 24	0.5

Table C

PIMA INDIANS DIABETES

Binary variable	Original variable	Cutpoint
a_1	Number of times pregnant	1
a_2	Number of times pregnant	3.5
a_3	Number of times pregnant	7
a_4	Plasma glucose concentration a 2 hours in an oral glucose tolerance test	127
a_5	Plasma glucose concentration a 2 hours in an oral glucose tolerance test	165
a_6	Diastolic blood pressure	67
a_7	Diastolic blood pressure	71
a_8	Diastolic blood pressure	76
a_9	Diastolic blood pressure	80
a_{10}	Triceps skin fold thickness	19.5
a_{11}	Triceps skin fold thickness	26
a_{12}	Triceps skin fold thickness	32
a_{13}	2-Hour serum insulin	69

a_{14}	2-Hour serum insulin	110
a_{15}	2-Hour serum insulin	164
a_{16}	2-Hour serum insulin	197.5
a_{17}	Body mass index	30.1
a_{18}	Body mass index	45.4
a_{19}	Diabetes pedigree function	0.347
a_{20}	Diabetes pedigree function	0.5
a_{21}	Diabetes pedigree function	0.673
a_{22}	Age (years)	23
a_{23}	Age (years)	41

Table D

CLEVELAND HEART DISEASE

Binary variable	Original variable	Cutpoint
a_1	age	55.5
a_2	age	62
a_3	sex	0.5
a_4	cp	3
a_5	trestbps	109
a_6	trestbps	136
a_7	trestbps	156
a_8	chol	233
a_9	fbs	0.5

a_{10}	restecg	1
a_{11}	thalach	125.583
a_{12}	exang	0.5
a_{13}	oldpeak	0.3
a_{14}	oldpeak	0.5
a_{15}	slope	1
a_{16}	ca	0
a_{17}	thal	3

Table E

AUSTRALIAN CREDIT

Binary variable	Original variable	Cutpoint
a_1	A1	0.5
a_2	A2	23.375
a_3	A2	27.035
a_4	A2	29.71
a_5	A2	33.96
a_6	A2	39.915
a_7	A3	1.395
a_8	A3	2.665
a_9	A3	4.0425
a_{10}	A3	5.895
a_{11}	A3	7.9375

a_{12}	A4	1.5
a_{13}	A5	4.5
a_{14}	A5	6
a_{15}	A5	7.5
a_{16}	A5	8.5
a_{17}	A5	10
a_{18}	A6	2.5
a_{19}	A6	4.5
a_{20}	A6	5.5
a_{21}	A6	6
a_{22}	A7	0.375
a_{23}	A7	0.875
a_{24}	A7	1.395
a_{25}	A7	2.3125
a_{26}	A7	3.875
a_{27}	A8	0.5
a_{28}	A9	0.5
a_{29}	A10	0.5
a_{30}	A10	1.5
a_{31}	A10	2.5
a_{32}	A10	4
a_{33}	A10	5.5
a_{34}	A11	0.5

a_{35}	A12	1.5
a_{36}	A13	87
a_{37}	A13	130
a_{38}	A13	170.5
a_{39}	A13	220
a_{40}	A13	280
a_{41}	A14	10
a_{42}	A14	76
a_{43}	A14	223
a_{44}	A14	518.5
a_{45}	A14	1401

Table F

IONOSPHERE

Binary variable	Original variable	Cutpoint
a_1	Attribute 1	0.5
a_2	Attribute 2	0.26223
a_3	Attribute 2	0.614535
a_4	Attribute 3	-0.46032
a_5	Attribute 3	0.62245
a_6	Attribute 4	0.09844
a_7	Attribute 4	0.824325
a_8	Attribute 5	-0.44056

a_9	Attribute 6	0.149065
a_{10}	Attribute 6	0.516665
a_{11}	Attribute 7	-0.79194
a_{12}	Attribute 8	-0.779705
a_{13}	Attribute 8	-0.028715
a_{14}	Attribute 8	0.32418
a_{15}	Attribute 9	-0.45817
a_{16}	Attribute 9	0.25767
a_{17}	Attribute 9	0.62414
a_{18}	Attribute 10	-0.81263
a_{19}	Attribute 10	0.28425
a_{20}	Attribute 10	0.63587
a_{21}	Attribute 11	-0.813615
a_{22}	Attribute 11	-0.451775
a_{23}	Attribute 12	-0.82287
a_{24}	Attribute 12	-0.46797
a_{25}	Attribute 12	0.269625
a_{26}	Attribute 13	-0.411275
a_{27}	Attribute 14	-0.81353
a_{28}	Attribute 14	-0.036675
a_{29}	Attribute 14	0.3254
a_{30}	Attribute 15	-0.455625
a_{31}	Attribute 16	-0.82353

a_{32}	Attribute 16	0.277815
a_{33}	Attribute 16	0.640465
a_{34}	Attribute 17	-0.81818
a_{35}	Attribute 18	-0.8139
a_{36}	Attribute 18	-0.07269
a_{37}	Attribute 18	0.639865
a_{38}	Attribute 19	-0.81772
a_{39}	Attribute 19	-0.459005
a_{40}	Attribute 19	-0.1047
a_{41}	Attribute 19	0.652285
a_{42}	Attribute 20	-0.822895
a_{43}	Attribute 20	-0.10837
a_{44}	Attribute 20	0.245355
a_{45}	Attribute 21	-0.8181
a_{46}	Attribute 22	-0.818885
a_{47}	Attribute 22	-0.46174
a_{48}	Attribute 23	-0.823075
a_{49}	Attribute 23	-0.470615
a_{50}	Attribute 23	-0.118475
a_{51}	Attribute 24	-0.82329
a_{52}	Attribute 25	-0.82396
a_{53}	Attribute 25	-0.1123
a_{54}	Attribute 25	0.614975

a_{55}	Attribute 26	-0.816185
a_{56}	Attribute 27	-0.813725
a_{57}	Attribute 27	-0.076245
a_{58}	Attribute 28	-0.82436
a_{59}	Attribute 28	-0.463945
a_{60}	Attribute 28	-0.1021
a_{61}	Attribute 29	-0.82214
a_{62}	Attribute 29	-0.10603
a_{63}	Attribute 29	0.61558
a_{64}	Attribute 30	-0.774455
a_{65}	Attribute 30	-0.361305
a_{66}	Attribute 30	0.00406
a_{67}	Attribute 30	0.357855
a_{68}	Attribute 31	-0.822685
a_{69}	Attribute 31	0.622015
a_{70}	Attribute 32	-0.82067
a_{71}	Attribute 33	-0.81478

Table G

WISCONSIN BREAST CANCER

Binary variable	Original variable	Cutpoint
a_1	Clump Thickness	3
a_2	Clump Thickness	4

a_3	Clump Thickness	6
a_4	Clump Thickness	7
a_5	Clump Thickness	8
a_6	Uniformity of Cell Size	2
a_7	Uniformity of Cell Size	3
a_8	Uniformity of Cell Size	4
a_9	Uniformity of Cell Shape	2
a_{10}	Uniformity of Cell Shape	6
a_{11}	Marginal Adhesion	1
a_{12}	Marginal Adhesion	2
a_{13}	Marginal Adhesion	5
a_{14}	Single Epithelial Cell Size	4.25
a_{15}	Bare Nuclei	2
a_{16}	Bare Nuclei	4
a_{17}	Bland Chromatin	4
a_{18}	Normal Nucleoli	3
a_{19}	Normal Nucleoli	9
a_{20}	Mitoses	3

Table H

CONGRESSIONAL VOTING RECORDS

Binary variable	Original variable
a_1	handicapped-infants

a_2	water-project-cost-sharing
a_3	adoption-of-the-budget-resolution
a_4	physician-fee-freeze
a_5	el-salvador-aid
a_6	religious-groups-in-schools
a_7	anti-satellite-test-ban
a_8	aid-to-nicaraguan-contras
a_9	mx-missile
a_{10}	immigration
a_{11}	synfuels-corporation-cutback
a_{12}	education-spending
a_{13}	superfund-right-to-sue
a_{14}	crime
a_{15}	duty-free-exports
a_{16}	export-administration-act-south-africa

Curriculum Vita

Name Irina I. Lozina

Education

1981 – 1986: Department of Computational Mathematics and Cybernetics, University of Nizhny Novgorod, Russia

Degree earned: Master of Science in Mathematics (specialization Applied Mathematics)

2000 – 2007: RUTCOR – Rutgers Center for Operations Research, Graduate School – New Brunswick, Rutgers University, USA

Degree earned: Master of Science (May, 2003), Ph. D. candidate (graduation date: May 2007)

Positions held during the period between the conferral of the baccalaureate and the doctorate

1986 – 1989 (April): Engineer-programmer, Zavolzhsky Engine Plant, Russia

1989 (April) – 1989 (November): Engineer-programmer, Gorkovsky Radio Research-Science Institute, Russia

1989 – 2000: Engineer-constructor, Machine Design Bureau, Nizhny Novgorod, Russia

2000 – 2004: Graduate Fellow, RUTCOR, Rutgers Center for Operations Research, Graduate School–New Brunswick, Rutgers University

2004 – 2005: Graduate Assistant, RUTCOR, Rutgers Center for Operations Research, Graduate School–New Brunswick, Rutgers University

2005 – 2007: Research Assistant, RUTCOR, Rutgers Center for Operations Research, Graduate School–New Brunswick, Rutgers University.

List of Publications

M.W. Brauner, N. Brauner, P.L. Hammer, I. Lozina, D. Valeyre, "Logical analysis of computed tomography data to differentiate entities of idiopathic interstitial pneumonias",

In *Data Mining in Biomedicine*. P. Pardalos, V. Boginski and A. Vazacopoulos (Eds.), Springer, pp. 193-208, 2007.

Gabriela Alexe, Sorin Alexe, David E Axelrod, Tibérius O Bonates, Irina I Lozina, Michael Reiss, Peter L Hammer, **"**Breast cancer prognosis by combinatorial analysis of gene expression data", *Breast Cancer Research*, Vol.8 N.4 R41, 2006. (Available online at http://breast-cancer-research.com/content/8/4/R41)

Peter L. Hammer and Irina I. Lozina, "Boolean Separators and Approximate Boolean Classifiers", *RUTCOR Research Report*, RRR 14-2006. (http://rutcor.rutgers.edu/pub/rrr/reports2006/14_2006.pdf)

P. L. Hammer and I. I. Lozina, "Composite Boolean Separators for Data Analysis", *RUTCOR Research Report*, RRR 14-2007. (http://rutcor.rutgers.edu/pub/rrr/ reports2007/14_2007.pdf)