R U T C O R
R ESEARCH
R E P O R T

COMPOSITE BOOLEAN SEPARATORS FOR DATA ANALYSIS

Peter L. Hammer[a] Irina I. Lozina[b]

RRR 14-2007, MARCH 2007

RUTCOR
Rutgers Center for
Operations Research
Rutgers University
640 Bartholomew Road
Piscataway, New Jersey
08854-8003
Telephone: 732-445-3804
Telefax: 732-445-5472
Email: rrr@rutcor.rutgers.edu
http://rutcor.rutgers.edu/~rrr

[a] RUTCOR, Rutgers University, 640 Bartholomew Rd., Piscataway, NJ 08854-8003
[b] RUTCOR, Rutgers University, 640 Bartholomew Rd., Piscataway, NJ 08854-8003, ilozina@rutcor.rutgers.edu

RUTCOR RESEARCH REPORT
RRR 14-2007, MARCH 2007

COMPOSITE BOOLEAN SEPARATORS FOR DATA ANALYSIS

Peter L. Hammer Irina I. Lozina

Abstract. In [12], we proposed a simple procedure for generating artificial Boolean variables. In this paper, we present a formal description of the procedure and apply the new variables to different problems in machine-learning / data-mining. In particular, we demonstrate the usefulness of these concepts by showing how the introduction of artificial variables can enhance the accuracy of classification systems; we employ the new variables for identifying misclassified observations and examine how deletion of such observations and reversal of their class influence the classification accuracy; we apply the new artificial variables to the attribute selection problem, i.e., to the problem of identifying informative subsets of the original attributes. All the results have been tested on eight publicly available datasets and validated by five well-known machine-learning / data-mining methods.

1 Introduction

In [12], we proposed a simple procedure for generating artificial Boolean variables. In this paper, we present a formal description of the procedure and apply the new variables to different problems in machine-learning / data-mining. In particular,

- ✓ we demonstrate the usefulness of these concepts by showing how the introduction of artificial variables can enhance the accuracy of classification systems;
- ✓ we employ the new variables for identifying misclassified observations and examine how deletion of such observations and reversal of their class influence the classification accuracy;
- ✓ we apply the new artificial 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.

In our computational experiments we use five well-known machine-learning / data-mining methods:

- ➢ support vector machines (SMO),
- ➢ artificial neural networks (MP),
- ➢ linear logistic regression (SL),
- ➢ decision trees (C4.5),
- ➢ logical analysis of data (LAD).

For the first four methods we have used software available in the WEKA package [21], while for LAD we have used the Datascope software developed at RUTCOR [14]. The methods were experimentally applied to the following eight publicly available benchmark datasets:

- BUPA liver-disorders (**bld**),
- German credit (**ger**),
- Pima Indians Diabetes (**pid**),
- Cleveland heart disease (**hea**),
- Australian credit (**aus**),
- Ionosphere (**ion**),
- Wisconsin breast cancer (**bcw**),
- Congressional voting records (**vot**),

taken from the Repository of the University of California at Irvine [13]. The description and binarization of these datasets can be found in [12].

2 Generation of Composite Boolean Separators

In this section we recall basic notions described in [12]. 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.

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 [12], we described a procedure for creating a set of artificial Boolean variables. In this section, we formalize this procedure. Observe that in the present paper we use a different terminology to name the artificial variables and call them *Composite Boolean Features* (CBFs).

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.

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:

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

Obviously, we can exclude from our consideration the two constant functions 0 and 1. Also, we do not need 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 paper, 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_1x_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:

$$\bar{x}_2,\ x_1\bar{x}_2,\ \ x_1 \vee x_3,\ x_1 \oplus x_3,\ \ \bar{x}_2x_3,\ \ \bar{x}_2\bar{x}_4,\ \ \bar{x}_2 \vee x_4,\ \ \overline{x_2 \oplus x_4}\ .$$

Obs.	x_1	x_2	x_3	x_4	$\bar{x}_2$	$x_1\bar{x}_2$	$x_1 \vee x_3$	$x_1 \oplus x_3$	$\bar{x}_2x_3$	$\bar{x}_2\bar{x}_4$	$\bar{x}_2 \vee x_4$	$\overline{x_2 \oplus x_4}$
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	$x_5 = \bar{x}_2$	$x_6 = x_1\bar{x}_2$	$x_7 = x_1 \vee x_3$	$x_8 = x_1 \oplus x_3$	$x_9 = \bar{x}_2x_3$	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.

The CBFs identified in the process described above can be regarded as synthetic variables associated to the dataset. As such, they can be simply added to the original data, and the process 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* (CBSes) of the original dataset (in [12], composite Boolean separators were named Approximate Boolean Classifiers).

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 $x_1, \ldots, x_9$ 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.	$f_1 = x_5x_7$	$f_2 = x_6 \vee x_8$	$f_3 = x_6 \vee x_9$	$f_4 = x_8 \vee x_9$	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, …, x_9 as functions of the original variables x_1, …, 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 Classification with Composite Boolean Separators

It has been shown in [12] 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.

The CBSes can be used for classification purposes in two different ways. First, we interpret composite Boolean separators as artificial variables. Second, we view each separator as a classification system.

3.1 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 1 reported below we present the results of applying the five classification methods to the eight datasets listed in the introduction, 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 1
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 1 show *high quality of CBSes as artificial variables*.

3.2 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 2 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.

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.*

Table 2

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%		

4 Identification of Misclassified Observations by CBSes

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 ([1], [4], [5], [6], [16], [17], [18], [19], [22], [23], [24]). To enhance the quality of data, we propose here a technique for identifying suspicious observations, i.e., those which were supposedly misclassified.

4.1 Consistent Composite Boolean Separators and Suspicious Observations

A phenomenon observed in many datasets is that each observation in the set 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. 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 3.

Table 3
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 $R \bigcup T$ 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 $R \bigcup S \bigcup T$, with the average accuracies obtained by training on the set $R \bigcup T$ and testing on the strongly suspicious set *S*. After randomly partitioning in 20 different ways each of the datasets $R \bigcup T$ 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 4.

Table 4
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 $R \cup T$, 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 $R \cup T$ 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 5.

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 $R \cup S \cup T$.

Table 5
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 6. 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 6

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, let us 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 4). Also, we have seen in Table 5 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 7 and Table 8 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 7
AVERAGE ACCURACY OF MODELS LEARNED ON S BY 5 CLASSIFICATION METHODS

Dataset	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 8
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 7 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 8).

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.*

4.2 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, 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 9 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 9

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 10 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.

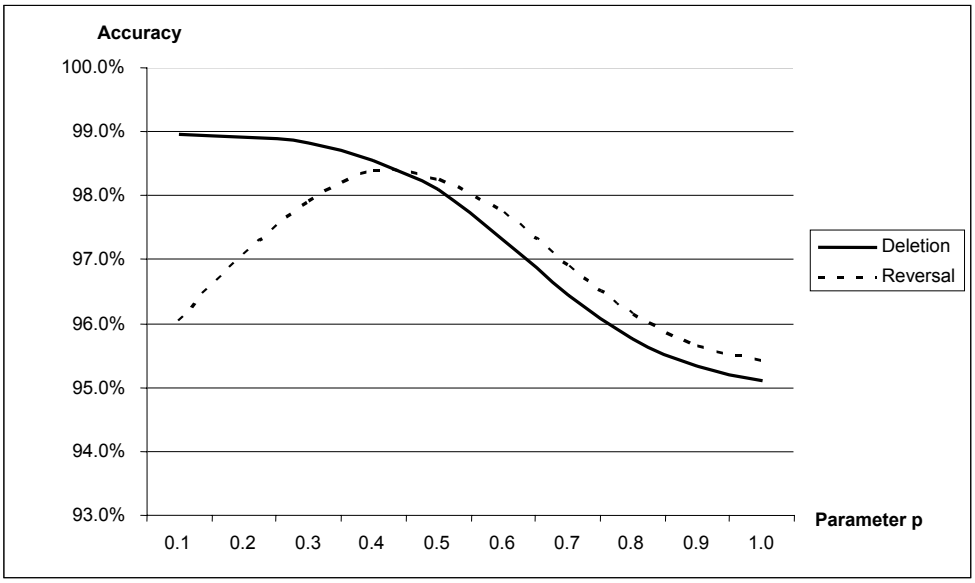
Table 10
AVERAGE ACCURACY OF 5 CLASSIFICATION METHODS FOR THE DATASET **HEA** ($0.75 \leq p \leq 1$)

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 1. It is important to remember however that this picture provides only an approximate description of the real phenomenon.

Figure 1



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 paper, 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 11
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 12 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 12
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					

5 Attribute Selection

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. There is a rich literature dedicated to the problem of identifying informative subsets of attributes (see [10], [11], [15] and for survey see [2] and [7]). In this section, we contribute to this problem by employing the concept of CBS and compare our approach with two other methods, C*FS* and *Consistency*, which are standard procedures of the WEKA package [21].

5.1 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 [3]) 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 in [12]) 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* on which this CBS depends;
- take the union of all the subsets found in the previous step.

To illustrate this approach, let us return to the example in Section 2. 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*) defines the informative set of attributes to be those *original binary variables* which are present in the formula for a CBS with the highest *CP*. 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 the set $\{ x_1, x_2, x_3 \}$ 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 the introduction. 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 13 and Table 14 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 13
RESULTS FOR ATTRIBUTE SELECTION WITH ALL_CBSES

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 14
RESULTS FOR ATTRIBUTE SELECTION WITH ONE_CBS

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 (see [12]) 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 15 and Table 16 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.

Conclusion 6. *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****).*

Table 15
RESULTS FOR ATTRIBUTE SELECTION WITH ALL_CBSES

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%	-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 16
RESULTS FOR ATTRIBUTE SELECTION WITH ONE_CBS

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%

5.2 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 [10], [11] (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 [11] (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 17 we present the percentage of correct classifications averaged over twenty 10-foding experiments obtained by the two WEKA methods.

Table 17
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 18, 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 18
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 7. *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.*

6 Conclusion

In this paper, we presented a formal description of the procedure for creating new variables that represent logical functions of the given variables (called CBSes) and demonstrated in various ways the usefulness of these artificial variables 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 methods.

Our results are purely experimental, which is in accordance with the following observation by Thomas G. Dietterich [8]:

> "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".

The results demonstrated in this paper 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 [20]).

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 [9] 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] 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.

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

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

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

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

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

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

[8] 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].

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

[10] 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.

[11] 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.

[12] 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).

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

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

[15] 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.

[16] 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.

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

[18] 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.

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

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

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

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

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

[24] 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.