**Instituto de Ciências Matemáticas e de Computação**

**ISSN - 0103-2569**

# Applying Knowledge-Driven Constructive Induction: Some Experimental Results

**Huei Diana Lee**
**Maria Carolina Monard/ILTC**

**Nº 101**

RELATÓRIOS TÉCNICOS DO ICMC

São Carlos
Fevereiro/2000

# Applying Knowledge-Driven Constructive Induction: Some Experimental Results *

**Huei Diana Lee**
**Maria Carolina Monard/ILTC**

University of São Paulo
Institute of Mathematics and Computer Sciences
Department of Computer Science and Statistics
Laboratory of Computational Intelligence
P.O. Box 668, 13560-970 - São Carlos, SP, Brazil
e-mail: {huei, mcmonard}@icmc.sc.usp.br

**Abstract**

Inductive-learning algorithms induce a concept description from given concept instances — training examples. However, if the provided features for describing the training examples are inadequate, the learning algorithms are likely to produce inaccurate descriptions. Features can sometimes be considered inadequate for the learning task when they are weakly or indirectly relevant or inappropriately measured. However, these features can sometimes be combined conveniently, generating new constructed features that can turn out to be highly relevant. This work describes empirical results using Knowledge-driven constructive induction which is based on domain knowledge provided by an expert, aiming to construct new features that produce more accurate descriptions. Several experiments performed on four real world datasets using $\mathcal{C}4.5$rules and $\mathcal{CN}2$ as inducers are described. The reported results include, for each experiment, a description of the new features constructed, error rates and features selected by each inducer.

*Keywords:* Constructive Induction; Machine Learning.

**2000**

*Work partially supported by State University of the West of Paraná — UNIOESTE and National Research Council — CAPES and FINEP.

This document was produced with the LaTeX typeset system and the BibTeX reference management system with help of the BibView tool (Prati et al., 1999). As with all reviewing work, it almost certainly contains errors and has plenty of room for improvements. Please report any error, typos, inconsistences, omissions and suggestions for improvements to huei@icmc.sc.usp.br.

This document and possible updates can be found at the ICMC site:

ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_101.ps.zip

# Contents



# List of Figures



# List of Tables

# 1 Introduction

Conventional inductive-learning algorithms rely on existing (user) provided data to build their descriptions. Inadequate representation space or description language as well as errors in training examples can make learning problems be difficult.

Features can be considered inadequate for the learning task when they are weakly or indirectly relevant, conditionally relevant or inappropriately measured (Lee, 1999; Baranauskas and Monard, 1999; Baranauskas et al., 1999). If the provided features for describing the training examples are inadequate, the learning algorithms are likely to create excessively complex and inaccurate descriptions (Bloedorn and Michalski, 1998).

However, these individually inadequate features can sometimes be combined conveniently, generating new features which can turn out to be highly representative to the description of a concept. The process of constructing new features is called Feature Construction or Constructive Induction (Michalski, 1978).

The objective of this work is to evaluate the effects of Feature Construction when this is done with the aid of the user/specialist. We describe a series of experiments performed on four real world datasets, using two inducers: $\mathcal{C}4.5$-rules and $\mathcal{CN}2$. The reported results include, for each experiment, a description of the new features constructed, error rates, features selected by each inducer and others.

This work is organized as follows: Section 2 gives some background about Feature Construction. Section 3 briefly describes the induction algorithms used in the experiments and Section 4 gives a short description of the datasets used to run the algorithms. Section 5 shows the experimental setup used to run the experiments and Section 6 describes the results obtained from these experiments. Section 7 presents some considerations of results. Finally, Section 8 gives some conclusions.

# 2 Constructive Induction

Feature construction, also known as Constructive Induction — CI, is the process of combining primitive[1] features producing new features possibly relevant to a concept description. In other words, CI can be defined as:

> The application of constructive operators, *i.e.* operators used to compound features from the existing ones, resulting in the definition of one or more features.

It is important to notice that, unlikely feature subset selection where only selected features are shown to the inductive algorithm, thus decreasing feature search space (Lee et al., 1999; Baranauskas and Monard, 1999; Baranauskas et al., 1999), Constructive Induction augments feature search space.

Feature construction requires answers to the following questions:

> *Which constructive operators should be used?* and
>
> *Which primitive features should be combined using this operators?*

Another important observation is that, in general, the feature construction process is intractable since the number of features which can be constructed is a combinatorial function of the number

[1] Features in the original dataset.

of existing features multiplied by the number of possible operators. Consequently, CI is feasible only when articulated with heuristcs that may reduce the number of possible features and the number of constructive operators which are going to be used to construct new features.

## 2.1 Constructive Induction Approaches

Constructive Induction methods can be grouped according to the information used to search for the best representation space as follows (Bloedorn and Michalski, 1998; Wnek and Michalski, 1994; Wnek and Michalski, 1993):

1. data-driven constructive induction, based on analysis of the training data
2. hypothesis-driven constructive induction, based on analysis of inductive hypothesis. In this approach, useful concepts in the rules can be extracted and used to define new attributes
3. knowledge-driven constructive induction, based on domain knowledge provided by an expert and
4. multistrategy constructive induction, which uses two or more of the other methods.

The feature construction process can be guided and controlled by the user/specialist or can be automatically conducted by the learning system. In this work, we focus on Constructive Induction guided by user/specialist using thus the knowledge-driven approach.

## 2.2 An Example

The classical example of friend and enemy robots is given to illustrate feature construction. In this example, features *Head*, *Body*, *Holds* and *Smiles* are used to determine if a robot is a friend or an enemy. Table 2.1 shows the training examples.

| Examples | Head | Body | Smiles | Holds | Class |
|---|---|---|---|---|---|
| $E_1$ | square | square | yes | balloon | friend |
| $E_2$ | square | triangle | no | sword | enemy |
| $E_3$ | circle | circle | yes | flag | friend |
| $E_4$ | triangle | circle | yes | sword | enemy |
| $E_5$ | triangle | triangle | yes | balloon | friend |
| $E_6$ | circle | square | no | flag | enemy |

Table 2.1: Examples of Friend and Enemy Robots

After constructing the decision tree using these training examples, it can be observed that 2 of the 4 features are used to generate the tree, thus constructing a tree of depth 2 — Figure 2.1.

Suppose that a new feature named *Same-form* is constructed from the comparison between feature values *Head* and *Body* (primitive features) — see Table 2.2.

| Examples | Head | Body | Smiles | Holds | Same-form | Class |
|---|---|---|---|---|---|---|
| $E_1$ | square | square | yes | balloon | true | friend |
| $E_2$ | square | triangle | no | sword | false | enemy |
| $E_3$ | circle | circle | yes | flag | true | friend |
| $E_4$ | triangle | circle | yes | sword | false | enemy |
| $E_5$ | triangle | triangle | yes | balloon | true | friend |
| $E_6$ | circle | square | no | flag | false | enemy |

Table 2.2: Examples of Friend and Enemy Robots after Feature Construction

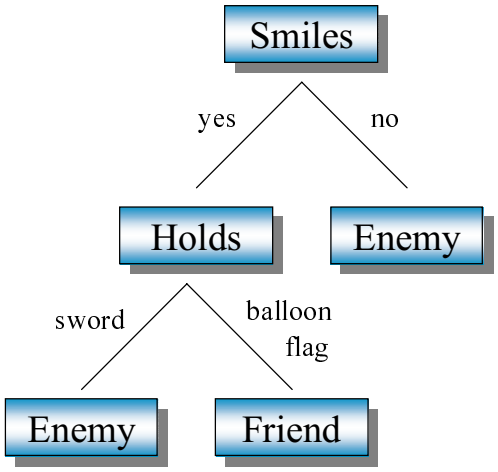


Figure 2.1: Decision Tree for the Problem of Friend and Enemy Robots

Now, it is possible to distinguish among *friend* and *enemy* robots using only this new feature. This is illustrated by the decision tree shown in Figure 2.2, which has been generated using the examples in Table 2.2. This decision tree states that a robot is a friend only if both its body and head have the same form. Thus, none of the primitive features were explicitly needed to induce a much simpler concept.

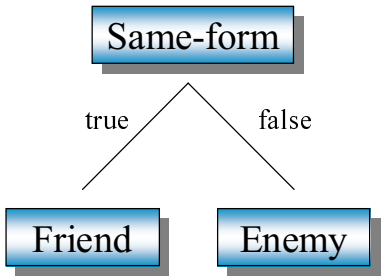


Figure 2.2: Decision Tree for the Problem of Friend and Enemy Robot after Feature Construction

The next section describes the inducers and the input data format used to run the knowledge-driven constructive induction experiments.

# 3 Inducers

Two inducers, $\mathcal{CN}2$ and $\mathcal{C}4.5$rules, found in the $\mathcal{MLC}$++ library (Kohavi et al., 1996), have been used in this work.

These inducers are well known in the ML community and belong to the eager learning approach. In this approach, the algorithms greedily compile the training data into an intentional concept description, such as a rule set or decision tree, discarding the data after this process (Aha, 1997). Only the learned concept is used to classify new cases.

### 3.1 Data Format

In supervised Machine Learning, it is generally presented to an inducer a set of training instances. Each instance is described typically by a vector of feature values and a class label whose value can be either discrete or continuous. This vector is denoted by $(\mathbf{X}, Y)$ and is known as the feature-value (either attribute-value or spreadsheet) format.

Table 3.1.1 illustrates this organization where a row $i$ refers to the $i$-th example or instance $\mathbf{X}_i$ and column entries $x_{ij}$ refer to the individual value of the $j$-th feature $f_j$ of instance $i$. The column rotulated as *class* refers to the label or classification of that instance.

| $f_1$ | $f_2$ | $\ldots$ | $f_m$ | *class* |
|---|---|---|---|---|
| $x_{11}$ | $x_{12}$ | $\ldots$ | $x_{1m}$ | $y_1$ |
| $x_{21}$ | $x_{22}$ | $\ldots$ | $x_{2m}$ | $y_2$ |
| $\vdots$ | $\vdots$ | $\ddots$ | $\vdots$ | $\vdots$ |
| $x_{n1}$ | $x_{n2}$ | $\ldots$ | $x_{nm}$ | $y_n$ |

Table 3.1.1: Feature-Value or Spreadsheet Format

The datasets file formats that $\mathcal{MLC}$++ recognizes by default are the *data*, *test* and *names* files. The *data* and *test* files contain labeled instances, one per line, of the training and test set respectively. The *names* file defines the scheme that allows parsing these two previous files; it describes the name and domain for each attribute and the label. The accuracy of the classifier produced by the inducer is measured on unseen data, *i.e.* the test set. More details can be found in (Kohavi et al., 1994; Felix et al., 1998; Baranauskas and Monard, 2000a).

### 3.2 $\mathcal{C}4.5$-rules

$\mathcal{C}4.5$-rules (Quinlan, 1993) examines the original decision tree produced by $\mathcal{C}4.5$[2] and derives from it a set of rules of the form $L \rightarrow R$. The left-hand side $L$ is a conjunction of attribute-based tests and the right-hand side is a class. One of the classes is also designated as a default.

To classify a case using a production rule model, the ordered list of rules is examined to find the first whose left-hand side is satisfied by the case. The predicted class is then the one nominated by the right-hand side of this rule. If no rule's left-hand side is satisfied, the case is predicted as belonging to the default class.

It is important to note that $\mathcal{C}4.5$-rules does not simply rewrite the tree to a collection of rules. In fact, it generalizes the rules by deleting superfluous conditions — *i.e.* irrelevant conditions that do not affect the conclusion — without affecting its accuracy, leaving the more appealing rules.

### 3.3 $\mathcal{CN}2$

$\mathcal{CN}2$ (Clark and Niblett, 1987; Clark and Niblett, 1989; Clark and Boswell, 1991) is a Machine Learning algorithm that induces '*if* <complex> *then* <class>' rules in domains where there might be noise. Each <complex> is a disjunction of conjunctions.

To classify a new instance using induced unordered rules (default $\mathcal{CN}2$ rule generation), all rules are tried and those which fire are collected. If more than one class is predicted by fired rules, the method used is to tag each rule with the distribution of covered examples among classes and

[2] $\mathcal{C}4.5$ (Quinlan, 1993) is one of the ID3 (Quinlan, 1986) successors. ID3 is a member of a more general Machine Learning family named Top Down Induction of Decision Trees – TDIDT.

then to sum these distributions to find the most probable class. For instance, consider the three rules:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **if** | Smiles=yes | **and** | Holds=sword | **then** | class=enemy | covers | [15,1] |
| **if** | Head=square | **and** | Body=square | **then** | class=friend | covers | [11,5] |
| **if** | Smiles=no | | | **then** | class=enemy | covers | [0,2] |

Here the two classes are [friend,enemy] and the first showed rule [15,1] denotes that the rule covers 15 training instances of friend and 1 of enemy. The second [11,5] denotes that the rule covers 11 training instances of friend and 5 of enemy. The third rule denotes that the rule covers 0 training instances of friend and 2 of enemy.

Given a new instance of a robot which has square head, square body, smiles and holds a sword, the first two rules are fired. $\mathcal{CN}2$ resolve this clash by summing the covered instances [36,6] and then predicting the most common class in the sum — friend.

In (Baranauskas and Monard, 2000b) a more detailed description of both algorithms can be found.

# 4 Datasets

Experiments were conducted on four real world domains. Datasets pima, cmc and hepatitis are from the UCI Irvine Repository (Blake et al., 1998). The smoke dataset can be obtained from: http://lib.stat.cmu.edu/datasets/csb/.

These four datasets were chosen from a set of nine datasets of a previous work in which the wrapper and filter approaches for feature subset selection were compared (Lee et al., 1999). The criterion used to choose these four datasets is related to our user/specialist domain knowledge since we are interested in his/her assistance to construct new features.

Section 4.2 summarizes datasets characteristics. It follows a basic datasets description.

## 4.1 General Description

**Pima** This dataset was donated by V. Sigillito, Applied Physics Laboratory, Johns Hopkins University to the UCI repository. This dataset is also a subset of a larger database maintained by the National Institute of Diabetes and Digestive and Kidney Diseases.

All patients are females at least 21 years old of Pima Indian heritage living near Phoenix, Arizona, USA. The problem is to predict whether a patient would test positive for diabetes according to World Health Organization (WHO) criteria — *i.e.*, if the 2-hour post-load plasma glucose is at least 200 mg/dl at any survey examination or if found during routine medical care — given a number of physiological measurements and medical test results.

**CMC** This dataset is composed by a subset of the 1987 National Indonesia Contraceptive Prevalence Survey and was donated by Tjen-Sien Lim. The samples are married women who were either not pregnant or do not know if they were at the time of the interview. The problem is to predict the current contraceptive method choice (no use, long-term methods or short-term methods) of a woman based on her demographic and socioeconomic characteristics. There are 1473 instances, 3 classes and 9 attributes.

**Smoke** This survey dataset (Bull, 1994) is concerned with the problem of predicting attitude toward restrictions on smoking in the workplace (prohibited, restricted or unrestricted) based on by-law-related, smoking-related and sociodemographic covariates. It is composed by 3 classes, 13 attributes and 2855 instances.

**Hepatitis** This dataset is for predicting life expectation of patients with hepatitis.

### 4.2 Datasets Summary

Table 4.2.1 summarizes the datasets used in this work. It shows, for each dataset, the number of instances (#Instances), number and percentage of duplicate (appearing more than once) or conflicting (same attribute-value but different class) instances, number of features (#Features) continuous and nominal, class distribution, the majority error and if the dataset have at least one missing value[3].

Datasets are presented in ascending order of the number of features, as will be in the remaining tables and graphs. Figure 4.1 shows datasets dimensionality, *i.e.* number of features and number of instances of each dataset. Observe that due to large variation, the number of instances in Figure 4.1 is represented as $\log_{10}$(#Instances).

| Dataset | # Instances | #Duplicate or conflicting (%) | # Features (cont.,nom.) | Class | Class % | Majority Error | Missing Values |
|---|---|---|---|---|---|---|---|
| pima | 769 | 1 (0.13%) | 8 (8,0) | 0 | 65.02% | 34.98% | N |
| | | | | 1 | 34.98% | on value 0 | |
| cmc | 1473 | 115 (7.81%) | 9 (2,7) | 1 | 42.70% | 57.30% | N |
| | | | | 2 | 22.61% | on value 1 | |
| | | | | 3 | 34.69% | | |
| smoke | 2855 | 29 (1.02%) | 13 (2,11) | 0 | 5.29% | 30.47% | N |
| | | | | 1 | 25.18% | on value 2 | |
| | | | | 2 | 69.53% | | |
| hepatitis | 155 | 0 (0%) | 19 (6,13) | die | 20.65% | 20.65% | Y |
| | | | | live | 79.35% | on value live | |

Table 4.2.1: Datasets Summary Descriptions

## 5 Experimental Setup

A series of experiments were performed, in order to evaluate the effectiveness of the new constructed features, using the algorithms and datasets described respectively in Section 3 and 4. It is important to observe that the original data has not been preprocessed in any way, for example by removing or replacing missing values or transforming nominal to numerical attributes. Furthermore, each individual inducer was run with default options setting for all parameters, *i.e.* no attempt was made to tune any inducer.

The user, consulted for the construction of new features for datasets pima and hepatitis, is a specialist in the domain. He also gave some suggestions about dataset pima. New features for datasets pima and smoke were constructed with the aid of a regular user.

The performed experiments can be divided into the following three steps – Figure 5.1:

- **First step**: After analyzing each dataset, the user/specialist suggested the construction of only one new feature (f1) for dataset hepatitis and the construction of two new features (f1 and f2) for the remaining datasets. Each original dataset was then augmented with the new features, constructing, for the four datasets considered, ten datasets named as shown in Table 5.1.

[3] These information is given by the $\mathcal{MLC}$++ *info* utility

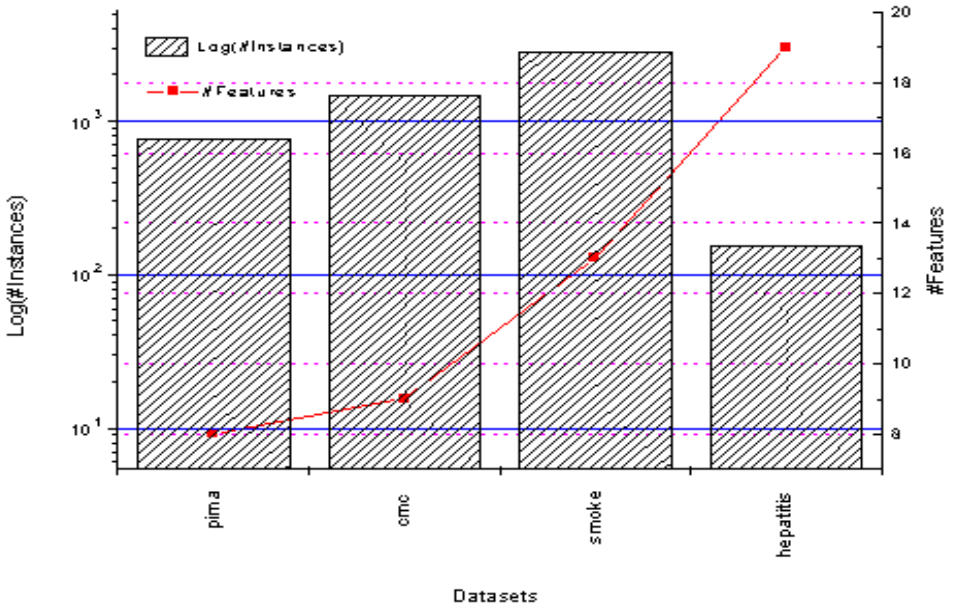


Figure 4.1: Datasets Dimensionality

| Original Dataset | Augmented Datasets | | |
|---|---|---|---|
| | f1 | f2 | f1,f2 |
| pima | pima01 | pima02 | pima-mlc |
| cmc | cmc01 | cmc02 | cmc-mlc |
| smoke | smoke01 | smoke02 | smoke-mlc |
| hepatitis | hepatiti01 | — | — |

Table 5.1: Original Datasets Augmented with Constructed Features

For example, dataset pima01 contains all features from the original dataset augmented with features f1; pima02 contains all features from original dataset augmented with feature f2 and pima-mlc contains all features from the original dataset augmented with features f1 and f2.

Afterwards, $\mathcal{C}$4.5rules and $\mathcal{CN}$2 were run once using as training set all examples in each of these ten new datasets. From now on the idea is to consider, for further investigation, only datasets which have generated rules where the new features f1 and/or f2 are mentioned, since this is an indication that these new constructed features are relevant to represent the concept.

In our experiments, as the rules induced by $\mathcal{C}$4.5rules and $\mathcal{CN}$2 used features f1 and f2 for all datasets, no one of them was discarded.

- **Second step**: Next, the two inducers $\mathcal{C}$4.5rules and $\mathcal{CN}$2 were run on each dataset — Table 5.1 — and the error rate was measured using 10-fold stratified cross-validation. After this, we selected for the next step only the datasets that verify the two following conditions:

  - accuracies improved comparing to the original datasets accuracies measured using 10-fold stratified cross-validation — SCV — and
  - at least one of the new constructed features was selected by the two inducers.

- **Third step**: Finally, for each dataset selected for this step, the primitive features used to construct the new features were removed from the dataset. Afterwards, the two inducers were run on these reduced datasets and error rate was measured using 10-fold stratified cross-validation.

  For example, considering dataset pima01, which contains all features from the original dataset pima augmented with feature f1; in this case, the two primitive features used to create feature f1 were removed. Similarly, for pima02, the two primitive features used to create f2 were removed. Finally, for pima-mlc features used to create f1 were removed as well as the ones used to create feature f2. Note that in this specific case only three primitive features were removed, since one of the primitive features used to construct the new features f1 and f2 was common to both.

Considering this whole process as a general methodology for applying Constructive Induction, the idea of running the third step was to verify if the considered algorithms could have a better performance if the primitive features used to construct the new features of each dataset were removed. Two reasons would be pointed out to do so:

- the new features are constructed from primitive features and some of these new features were selected by the inducers.
- most of Machine Learning algorithms, that are computationally feasible, do not work well or may be confused in the presence of a large number of features.

The perfect situation would be when the primitive features, used to construct the new ones, are not selected during the first step by the two inducers $\mathcal{C}4.5$rules and $\mathcal{CN}2$. In our experiments this was not the case, since the primitive features used to construct the new ones were also selected by the inducers. A possible reason for this would be that the constructed features do not capture perfectly the information embedded in each individual feature. Another reason for this would be that the datasets used in this work have already been worked out, so that the original features are, on its own, the most relevant ones.

# 6 Experimental Results

In this section the experimental results obtained are presented is detail.

## 6.1 Summary Tables Description

Six tables are presented for each original dataset:

- The first table describes each feature: feature number (features numbering starts at zero), feature name and type (continuous or nominal) in the original dataset. The last rows refer to the new features constructed which are indicated by new(#feature), f1(*name*) and f2(*name*). For nominal features, the maximum possible number of values (as described in the *names* file) and the actual number of values (the ones really found in the dataset through the $\mathcal{MLC}$++ *info* utility) are shown. It should be observed that a number of actual nominal values greater than the possible number of values indicates that there are missing values for that specific attribute. The reverse is not true.

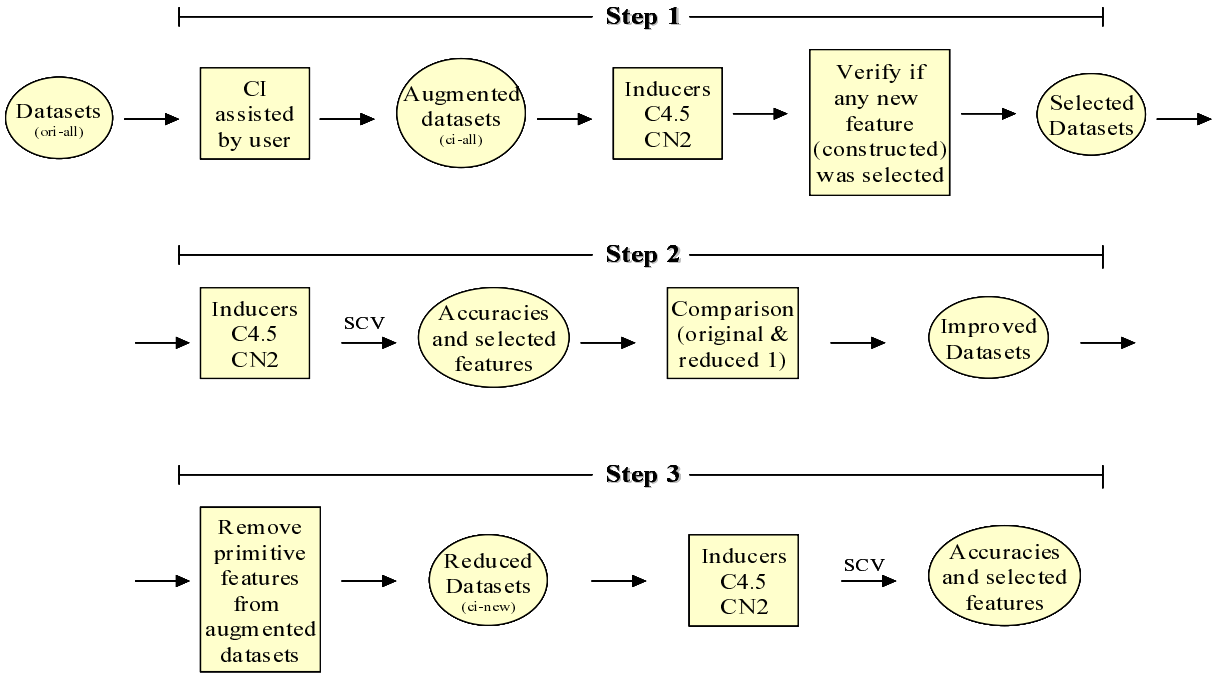


Figure 5.1: Experiments Steps

- The second table shows the features, original and constructed, for each dataset. Note again that constructed features are indicated by new(#feature).
- The third table describes the features selected by $\mathcal{C}$4.5rules and $\mathcal{CN}2$ inducers from the original dataset as well as features selected by these inducers after Constructive Induction, *i.e.* the augmented datasets. To specify the experiment, the notation (*set-of-features,inducer*) is used where:
  - *set-of-features* $\in$ {ori-all, ci-all, ci-new}. ori-all indicates that all features from the original dataset are being considered; ci-all indicates that all features from the original dataset augmented with the constructed ones are being used and ci-new indicates that the considered set of features are as ci-all but where the primitive features used to create the new features have been removed.
  - *inducer* $\in$ {$\mathcal{C}$4.5rules, $\mathcal{CN}2$ } indicates the algorithm that has been used.

  This table shows, for each original/augmented dataset and (*set-of-features,inducer*), the features subset selected, the number of features in the selected subset (#F), proportion of selected features (%F) as well as the time taken by the inducer to obtain the selected features. Time (in seconds) is related to a standard *Indigo 2* Silicon Graphics workstation. This table also shows the number of duplicate or conflicting instances for each dataset.

  For example, consider the first row for dataset pima in Table 6.2.3 (page 12):
  - the first column indicates that the subset of features — ori-all — given to $\mathcal{C}$4.5rules is the one compounded by all features from the original dataset
  - the second column presents the subset of features extracted by the correspondent inducer, *i.e.* those features that were present in the rules generated by the inducer

- the third column shows the total number of features in the dataset pima
- the fourth column shows the number of features used by the inducer to express the concept
- the fifth column presents the proportion of selected features
- the sixth column gives the time taken by $\mathcal{C}$4.5rules to induce the rules
- the seventh column presents the number of duplicate or conflicting instances for the dataset pima

The second row for dataset pima shows the same information, but using $\mathcal{CN}2$ as inducer.

In the same way, the first row for dataset pima-mlc shows similar information when given to $\mathcal{C}$4.5rules the subset of features — ci-all, *i.e.* all features from the original dataset augmented with the two new constructed features f1 and f2. The second row for dataset pima-mlc shows the same information as the first one, but using $\mathcal{CN}2$ as inducer. The third row presents the selected features, total number of features, number of selected features, proportion, time and number of duplicate or conflicting instances using $\mathcal{C}$4.5rules given the subset of features — ci-new, *i.e.* the considered set of features are as ci-all but having the primitive features used to create the new features removed.

- The fourth table shows similar information than the third one, but in a different way such that it is easy to visualize common features used by every (*set-of-features,inducer*). Note that features presented as • are features generated by Constructive Induction. The $\mathcal{C}$4.5rules inducer is represented as $\mathcal{C}$4.5r.
- The fifth table shows the error rate of each inducer (mean and standard deviation) using 10-fold stratified cross-validation (10-strat-cv) for each case described in Table 5.1 (page 7). The first column indicates the set of features given to the inducers; the second and third column indicate errors using $\mathcal{C}$4.5rules and $\mathcal{CN}2$ as inducers. For instance, the row ori-all shows errors when given the original set of features to the two inducers.
- The sixth table presents the difference in standard deviations errors (obtained from 10-stratified-cross-validation) between the original dataset and the derived datasets. This information is used to select the datasets to be considered in the third step.

Datasets marked with "—" are the ones not considered in the third step. As stated earlier, only the datasets that verify the following both conditions were selected for the third step:

- accuracy improved comparing to the original dataset accuracy and
- at least one of the new constructed features were selected by the two inducers.

For example, the dataset pima-mlc was not chosen for the third step — see page 12 — since it does not fill both conditions.

## 6.2 Pima and Derived Datasets

Two new features were constructed for this dataset with the help of the specialist. Their structure is shown bellow.

- f1(New-fc01): this features verifies if glucose and diastolic blood preassure are out of normal levels. It combines two primitive features: *plasma* and *diastolic*.

```
if (plasma>145 and diastolic>90) then new-fc01=1    % glucose and diastolic blood
                                                    % preassure are out of normal
    % levels
   else new-fc01=0
```

- f2(New-fc02): this feature verifies if glucose is out of the normal level and if 2-hour serum insulin is not present. It combines two primitive features: *plasma* and *two*.

```
if (plasma>145 and two=0) then new-fc02=1   % glucose is not at a normal level
                                            % and 2-hour serum insulin is not
  % present

   else new-fc02=0
```

| Feature Number | Feature Name | #Distinct Values | | |
|---|---|---|---|---|
| | | possible | actual | type |
| #0 | Number | - | 17 | continuous |
| #1 | Plasma | - | 136 | continuous |
| #2 | Diastolic | - | 47 | continuous |
| #3 | Triceps | - | 51 | continuous |
| #4 | Two | - | 186 | continuous |
| #5 | Body | - | 248 | continuous |
| #6 | Diabetes | - | 517 | continuous |
| #7 | Age | - | 52 | continuous |
| new#8 | f1(*New-fc01*) | - | 2 | nominal |
| new#9 | f2(*New-fc02*) | - | 2 | nominal |

Table 6.2.1: Pima and Derived Datasets – Feature Description

| Feature Number | Datasets | | | |
|---|---|---|---|---|
| | pima | pima-mlc | pima01 | pima02 |
| #0 | ◇ | ◇ | ◇ | ◇ |
| #1 | ◇ | ◇ | ◇ | ◇ |
| #2 | ◇ | ◇ | ◇ | ◇ |
| #3 | ◇ | ◇ | ◇ | ◇ |
| #4 | ◇ | ◇ | ◇ | ◇ |
| #5 | ◇ | ◇ | ◇ | ◇ |
| #6 | ◇ | ◇ | ◇ | ◇ |
| #7 | ◇ | ◇ | ◇ | ◇ |
| new#8 | | ◇ | ◇ | |
| new#9 | | ◇ | | ◇ |

Table 6.2.2: Pima and Derived Datasets – Original and Constructed Features

| | Selected Features | Total # F | # Selected F | %F | Time (s) | #Duplicate or Conflicting Instances |
|---|---|---|---|---|---|---|
| pima | | | | | | |
| (ori-all,C4.5-rules) | 1 2 5 6 7 | 8 | 5 | 62.50 | 0.2 | 1 |
| (ori-all,CN2) | 0 1 2 3 4 5 6 7 | 8 | 8 | 100.00 | 3.8 | |
| pima-mlc | | | | | | |
| (ci-all,C4.5-rules) | 1 2 4 5 6 7 | 10 | 6 | 60.00 | 1.8 | |
| (ci-all,CN2) | 0 1 2 3 4 5 6 7 9 | 10 | 9 | 90.00 | 4.7 | |
| (ci-new,C4.5-rules) | — | — | — | — | — | 1 |
| (ci-new,CN2) | — | — | — | — | — | 1 |
| 4.4 | | | | | | |
| pima01 | | | | | | |
| (ci-all,C4.5-rules) | 1 2 5 6 7 8 | 9 | 6 | 66.67 | 1.8 | |
| (ci-all,CN2) | 1 2 3 4 5 6 7 | 9 | 7 | 77.78 | 4.7 | |
| (ci-new,C4.5-rules) | 0 4 5 6 7 8 | 7 | 6 | 85.71 | 0.7 | 1 |
| (ci-new,CN2) | 0 3 4 5 6 7 8 | 7 | 6 | 85.71 | 5.1 | |

*continued on next page*

*continued from previous page*

| | Selected Features | Total # F | # Selected F | %F | Time (s) | #Duplicate or Conflicting Instances |
|---|---|---|---|---|---|---|
| pima02 | | | | | | |
| (ci-all,C4.5-rules) | 1 2 4 5 6 7 9 | 9 | 7 | 77.78 | 1.9 | |
| (ci-all,CN2) | 0 1 2 3 4 5 6 7 9 | 9 | 9 | 100.00 | 4.8 | |
| (ci-new,C4.5-rules) | — | — | — | — | — | 1 |
| (ci-new,CN2) | — | — | — | — | — | |

Table 6.2.3: Pima and Derived Datasets – Selected Features, Time for Selecting Features and Number of Duplicate or Conflicting Instances

| Dataset | pima | | pima-mlc | | | | pima01 | | | | pima02 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Feature Number | (ori-all, $\mathcal{C}$4.5r) | (ori-all, $\mathcal{CN}$2) | (ci-all, $\mathcal{C}$4.5r) | (ci-all, $\mathcal{CN}$2) | (ci-new, $\mathcal{C}$4.5r) | (ci-new, $\mathcal{CN}$2) | (ci-all, $\mathcal{C}$4.5r) | (ci-all, $\mathcal{CN}$2) | (ci-new, $\mathcal{C}$4.5r) | (ci-new, $\mathcal{CN}$2) | (ci-all, $\mathcal{C}$4.5r) | (ci-all, $\mathcal{CN}$2) | (ci-new, $\mathcal{C}$4.5r) | (ci-new, $\mathcal{CN}$2) |
| #0 | | ○ | | ○ | | ○ | | | ○ | ○ | | ○ | — | — |
| #1 | ○ | ○ | ○ | ○ | | | ○ | ○ | | | ○ | ○ | — | — |
| #2 | ○ | ○ | ○ | ○ | | | ○ | ○ | | | ○ | ○ | — | — |
| #3 | | ○ | | ○ | ○ | ○ | | ○ | | ○ | | ○ | — | — |
| #4 | | ○ | ○ | ○ | | | | ○ | ○ | ○ | ○ | ○ | — | — |
| #5 | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | — | — |
| #6 | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | — | — |
| #7 | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | — | — |
| #8 | | | | | • | • | • | | • | | | | — | — |
| #9 | | | | • | • | • | | | | | • | • | — | — |
| # Selected F | 5 | 8 | 6 | 9 | 6 | 7 | 6 | 8 | 6 | 6 | 7 | 9 | | |
| Total # F | 8 | 8 | 10 | 10 | 7 | 7 | 9 | 9 | 7 | 7 | 9 | 9 | | |
| % F | 62.50% | 100% | 60.00% | 90.00% | 85.71% | 100% | 66.67% | 88.89% | 85.71% | 85.71% | 77.78% | 100% | | |

Table 6.2.4: Pima Before and After Constructive Induction – Selected Features

| | $\mathcal{C}$4.5rules | $\mathcal{CN}$2 |
|---|---|---|
| pima 10-strat-cv | | |
| ori-all | 26,00±1,03 | 25,38±1,38 |
| pima-mlc 10-strat-cv | | |
| ci-all | 26,27±0,83 | 25,51±1,68 |
| ci-new | 26,78±1,88 | 28,63±1,35 |
| pima01 10-strat-cv | | |
| ci-all | 25,61±1,12 | 25,90±1,15 |
| ci-new | 31,73±1,37 | 32,66±1,33 |
| pima02 10-strat-cv | | |
| ci-all | 26,52±1,14 | 25,77±1,33 |
| ci-new | — | — |

Table 6.2.5: Pima and Derived Datasets – Error Rate

| | $\mathcal{C}$4.5rules | $\mathcal{CN}$2 |
|---|---|---|
| pima-mlc | | |
| ci-all | 0,29 | 0,08 |
| ci-new | 0,51 | 2,38 |
| pima01 | | |
| ci-all | -0,36 | 0,41 |
| ci-new | 4,73 | 5,37 |
| pima02 | | |
| ci-all | 0,48 | 0,29 |
| ci-new | — | — |

Table 6.2.6: Difference in Standard Deviations Between Original Dataset Pima and Derived Datasets

### 6.3 Cmc and Derived Datasets

Two new features were constructed for this dataset with the help of the user. Their structure is shown bellow.

- f1(Same-edu): this feature shows how equal or different the educational level of wife and husband are. It combines two primitive features: *wedu* and *hedu*.

```
if (wedu=hedu) then same-edu=exactly-same                % wife and husband have
                                                         % the same educational
                                                         % level
  else if (wedu-hedu=1) then same-edu=almost-wedu        % wife's educational
                                                         % level is higher
    else if (wedu-hedu=-1) then same-edu=almost-hedu     % husband's educational
                                                         % level is higher
      else if (wedu-hedu=2) then same-edu=wwedu          % wife's educ. level is
                                                         % two levels higher
        else if (wedu-hedu=-2) then same-edu=hhedu       % husband's educ. level
                                                         % is two levels higher
          else if (wedu-hedu=3) then same-edu=wedu-higher% wife's educ. level
                                                         % is much more higher
            else same-edu=hedu-higher                    % husband's educ. level
                                                         % is much more higher
```

- f2(Same-wedu-std): this feature shows if wife has a standard of living compatible with her educational level. It combines two primitive features: *wedu* and *stdliv*.

```
if (wedu=stdliv) then same-wedu-std=1 % wife has a standard of living compatible
                                      % with her educational level
   else same-wedu-std=0               % standard of living is not the same level of
                                      % the eduactional level
```

| Feature Number | Feature Name | #Distinct Values | | |
|---|---|---|---|---|
| | | possible | actual | type |
| #0 | Wage | - | 34 | continuous |
| #1 | Wedu | - | 4 | nominal |
| #2 | Hedu | - | 4 | nominal |
| #3 | Nchi | - | 15 | continuous |
| #4 | Wrel | - | 2 | nominal |
| #5 | Work | - | 2 | nominal |
| #6 | Hocu | - | 4 | nominal |
| #7 | Stdliv | - | 4 | nominal |
| #8 | Medexp | - | 2 | nominal |
| new#9 | f1(*Same-edu*) | - | 7 | Nominal |
| new#10 | f2(*Same-wedu-std*) | - | 2 | Nominal |

Table 6.3.1: Cmc and Derived Datasets – Feature Description

| Feature Number | Datasets | | | |
|---|---|---|---|---|
| | cmc | cmc-mlc | cmc01 | cmc02 |
| #0 | ◇ | ◇ | ◇ | ◇ |
| #1 | ◇ | ◇ | ◇ | ◇ |
| #2 | ◇ | ◇ | ◇ | ◇ |
| #3 | ◇ | ◇ | ◇ | ◇ |
| #4 | ◇ | ◇ | ◇ | ◇ |
| #5 | ◇ | ◇ | ◇ | ◇ |
| #6 | ◇ | ◇ | ◇ | ◇ |

*continued on next page*

*continued from previous page*

| Feature Number | Datasets | | | |
|---|---|---|---|---|
| | cmc | cmc-mlc | cmc01 | cmc02 |
| #7 | ◇ | ◇ | ◇ | ◇ |
| #8 | ◇ | ◇ | ◇ | ◇ |
| new#9 | | ◇ | ◇ | |
| new#10 | | ◇ | | ◇ |

Table 6.3.2: Cmc and Derived Datasets – Original and Constructed Features

| | Selected Features | Total # F | # Selected F | %F | Time (s) | #Duplicate or Conflicting Instances |
|---|---|---|---|---|---|---|
| cmc | | | | | | |
| (ori-all,C4.5-rules) | 0 1 2 3 4 5 6 7 8 | 9 | 9 | 100.00 | 13.5 | 115 |
| (ori-all,CN2) | 0 1 2 3 4 5 6 7 8 | 9 | 9 | 100.00 | 17.1 | |
| cmc-mlc | | | | | | |
| (ci-all,C4.5-rules) | 0 1 2 3 4 5 6 7 8 | 11 | 9 | 81.82 | 16.6 | |
| (ci-all,CN2) | 0 1 2 3 4 5 6 7 8 9 10 | 11 | 11 | 100.00 | 20.1 | |
| (ci-new,$\mathcal{C}$4.5-rules) | — | — | — | — | — | 115 |
| (ci-new,$\mathcal{CN}$2) | — | — | — | — | — | |
| cmc01 | | | | | | |
| (ci-all,C4.5-rules) | 0 1 2 3 4 5 6 7 8 | 10 | 9 | 90.00 | 16.1 | |
| (ci-all,CN2) | 0 1 2 3 4 5 6 7 8 9 | 10 | 10 | 100.00 | 20.8 | |
| (ci-new,C4.5-rules) | 0 3 4 5 6 7 8 9 | 8 | 8 | 100.00 | 10.6 | 167 |
| (ci-new,CN2) | 0 3 4 5 6 7 8 9 | 8 | 8 | 100.00 | 18.0 | |
| cmc02 | | | | | | |
| (ci-all,C4.5-rules) | 0 1 2 3 4 5 6 7 8 10 | 10 | 10 | 100.00 | 16.6 | |
| (ci-all,CN2) | 0 1 2 3 4 5 6 7 8 10 | 10 | 10 | 100.00 | 20.1 | |
| (ci-new,C4.5-rules) | 0 2 3 4 5 6 8 10 | 8 | 8 | 100.00 | 8.4 | 240 |
| (ci-new,CN2) | 0 2 3 4 5 6 8 10 | 8 | 8 | 100.00 | 16.6 | |

Table 6.3.3: Cmc and Derived Datasets – Selected Features, Time for Selecting Features and Number of Duplicate or Conflicting Instances

| Dataset | cmc | | cmc-mlc | | | | cmc01 | | | | cmc02 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Feature Number | (ori-all, $\mathcal{C}$4.5r) | (ori-all, $\mathcal{CN}$2) | (ci-all, $\mathcal{C}$4.5r) | (ci-all, $\mathcal{CN}$2) | (ci-new, $\mathcal{C}$4.5r) | (ci-new, $\mathcal{CN}$2) | (ci-all, $\mathcal{C}$4.5r) | (ci-all, $\mathcal{CN}$2) | (ci-new, $\mathcal{C}$4.5r) | (ci-new, $\mathcal{CN}$2) | (ci-all, $\mathcal{C}$4.5r) | (ci-all, $\mathcal{CN}$2) | (ci-new, $\mathcal{C}$4.5r) | (ci-new, $\mathcal{CN}$2) |
| #0 | ○ | ○ | ○ | ○ | — | — | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| #1 | ○ | ○ | ○ | ○ | — | — | ○ | ○ | | | ○ | ○ | | |
| #2 | ○ | ○ | ○ | ○ | — | — | ○ | ○ | | | ○ | ○ | ○ | ○ |
| #3 | ○ | ○ | ○ | ○ | — | — | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| #4 | ○ | ○ | ○ | ○ | — | — | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| #5 | ○ | ○ | ○ | ○ | — | — | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| #6 | ○ | ○ | ○ | ○ | — | — | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| #7 | ○ | ○ | ○ | ○ | — | — | ○ | ○ | ○ | ○ | ○ | ○ | | |
| #8 | ○ | ○ | ○ | ○ | — | — | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| #9 | | | | ● | — | — | | ● | ● | ● | | | | |
| #10 | | | | ● | — | — | | | | | ● | ● | ● | ● |
| # Selected F | 9 | 9 | 9 | 11 | | | 9 | 10 | 8 | 8 | 10 | 10 | 8 | 8 |
| Total # F | 9 | 9 | 11 | 11 | | | 10 | 10 | 8 | 8 | 10 | 10 | 8 | 8 |
| % F | 100% | 100% | 81.82% | 100% | | | 90.00% | 100% | 100% | 100% | 100% | 100% | 100% | 100% |

Table 6.3.4: Cmc Before and After Constructive Induction – Selected Features

| | $\mathcal{C}$4.5rules | $\mathcal{CN}$2 |
|---|---|---|
| cmc 10-strat-cv | | |
| ori-all | 45,90±1,38 | 49,64±1,01 |
| cmc-mlc 10-strat-cv | | |
| ci-all | 46,98±1,36 | 50,37±1,06 |
| ci-new | — | — |
| cmc01 10-strat-cv | | |
| ci-all | 47,87±1,54 | 49,50±1,04 |
| ci-new | 47,86±0,84 | 51,87±0,84 |

*continued on next page*

*continued from previous page*

| | $\mathcal{C}$4.5rules | $\mathcal{CN}2$ |
|---|---|---|
| cmc02 10-strat-cv | | |
| ci-all | 46,37±0,97 | 52,22±1,09 |
| ci-new | 47,73±0,88 | 52,61±1,29 |

Table 6.3.5: Cmc and Derived Datasets – Error Rate

| | $\mathcal{C}$4.5-rules | $\mathcal{CN}2$ |
|---|---|---|
| cmc-mlc | | |
| ci-all | 0,45 | 0,14 |
| ci-new | — | — |
| cmc01 | | |
| ci-all | 1,09 | -0,68 |
| ci-new | 1,51 | 1,72 |
| cmc02 | | |
| ci-all | -0,07 | 1,85 |
| ci-new | 1,35 | 2,02 |

Table 6.3.6: Difference in Standard Deviations Between Cmc and Derived Datasets

## 6.4 Smoke and Derived Datasets

Two new features were constructed for this dataset with the help of the user. Their structure is shown bellow.

- f1(Smoking): this feature represents the status of the interviewed person at the time of survey. It combines four primitive features: *smoking1*, *smoking2*, *smoking3* and *smoking4*.

```
if (smoking1=0 and smoking2=0 and smoking3=0) % smoking1, smoking2 and smoking3
                                              % = 0 indicate that has never
  then smoking=never                          % smoked
  else if (smoking1=1)                        % indicates that the interviewed
          then smoking=current                % person is a current smoker
    else if (smoking2=1)                      % indicates that the interviewed
            then smoking=quit <= 6 months     % person has quit smoking less (or
                                              % equal) than 6 months ago
      else if (smoking3=1 and smoking4=1)     % interviewed person has quit from
              then smoking=quit 6-12 months   % smoking 6 to 12 months ago
        else if (smoking3=1 and smoking4=0)   % interviewed person has quit
                then smoking=quit > 12 months % smoking more than one year ago
           else smoking=not-defined           % some inconsistency in data
               % was found
```

- f1(Place): this feature shows a comparison between the place the interviewed person works with respect to the city of Toronto-Canada (work1), if he (she) works at home or not (work2) and if he (she) lives in the city of Toronto or outside it (residence). It combines three primitive features: *work1*, *work2* and *residence*.

```
if (work1=work2) then place=0                 % work place city is the same as
                                              % work place home
   else if (work1=residence) then place=1     % work place city is the same as
                                              % residence
      else place=2
```

| Feature Number | Feature Name | #Distinct Values possible | actual | type |
|---|---|---|---|---|
| #0 | Weight | - | 128 | continuous |
| #1 | Time | - | 2 | nominal |
| #2 | Work1 | - | 2 | nominal |
| #3 | Work2 | - | 2 | nominal |
| #4 | Residence | - | 2 | nominal |
| #5 | Smoking1 | - | 2 | nominal |
| #6 | Smoking2 | - | 2 | nominal |
| #7 | Smoking3 | - | 2 | nominal |
| #8 | Smoking4 | - | 2 | nominal |
| #9 | Knowledge | - | 13 | nominal |
| #10 | Sex | - | 2 | nominal |
| #11 | Age | - | 73 | continuous |
| #12 | Education | - | 5 | nominal |
| new#13 | f1(*Smoking*) | - | 5 | nominal |
| new#14 | f2(*Place*) | - | 3 | nominal |

Table 6.4.1: Smoke and Derived Datasets – Feature Description

| Feature Number | Datasets smoke | smoke-mlc | smoke01 | smoke02 |
|---|---|---|---|---|
| #0 | ◇ | ◇ | ◇ | ◇ |
| #1 | ◇ | ◇ | ◇ | ◇ |
| #2 | ◇ | ◇ | ◇ | ◇ |
| #3 | ◇ | ◇ | ◇ | ◇ |
| #4 | ◇ | ◇ | ◇ | ◇ |
| #5 | ◇ | ◇ | ◇ | ◇ |
| #6 | ◇ | ◇ | ◇ | ◇ |
| #7 | ◇ | ◇ | ◇ | ◇ |
| #8 | ◇ | ◇ | ◇ | ◇ |
| #9 | ◇ | ◇ | ◇ | ◇ |
| #10 | ◇ | ◇ | ◇ | ◇ |
| #11 | ◇ | ◇ | ◇ | ◇ |
| #12 | ◇ | ◇ | ◇ | ◇ |
| new#13 | | ◇ | ◇ | |
| new#14 | | ◇ | | ◇ |

Table 6.4.2: Smoke and Derived Datasets – Original and Constructed Features

| | Selected Features | Total # F | # Selected F | %F | Time (s) | #Duplicate or Conflicting Instances |
|---|---|---|---|---|---|---|
| smoke | | | | | | |
| (ori-all,C4.5-rules) | 0 1 2 3 4 5 6 8 9 10 11 12 | 13 | 12 | 92.31 | 68.1 | 29 |
| (ori-all,CN2) | 0 1 2 3 4 5 6 7 8 9 10 11 12 | 13 | 13 | 100.00 | 57.4 | |
| smoke-mlc | | | | | | |
| (ci-all,C4.5-rules) | 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 | 15 | 15 | 100.00 | 64.6 | |
| (ci-all,CN2) | 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 | 15 | 15 | 100.00 | 84.7 | |
| (ci-new,C4.5-rules) | 0 1 9 10 11 12 13 14 | 8 | 8 | 100.00 | 58.4 | 41 |
| (ci-new,CN2) | 0 1 9 10 11 12 13 14 | 8 | 8 | 100.00 | 67.3 | |
| smoke01 | | | | | | |
| (ci-all,C4.5-rules) | 0 1 2 3 4 5 6 7 8 9 10 11 12 | 14 | 13 | 92.86 | 64.8 | |
| (ci-all,CN2) | 0 1 2 3 4 5 6 7 8 9 10 11 12 13 | 14 | 14 | 100.00 | 84.7 | |
| (ci-new,C4.5-rules) | 0 1 2 3 4 9 10 11 12 13 | 10 | 10 | 100.00 | 47.1 | 29 |
| (ci-new,CN2) | 0 1 2 3 4 9 10 11 12 13 | 10 | 10 | 100.00 | 65.8 | |
| smoke02 | | | | | | |
| (ci-all,C4.5-rules) | 0 1 2 3 4 5 6 7 8 9 10 11 12 14 | 14 | 14 | 100.00 | 65.2 | |
| (ci-all,CN2) | 0 1 2 3 4 5 6 7 8 9 10 | | | | | |

*continued on next page*

*continued from previous page*

| | Selected Features | Total # F | # Selected F | %F | Time (s) | #Duplicate or Conflicting Instances |
|---|---|---|---|---|---|---|
| | 11 12 14 | 14 | 14 | 100.00 | 85.4 | |
| (ci-new,C4.5-rules) | 0 1 5 6 7 8 9 10 11 12 14 | 11 | 11 | 100.00 | 60.7 | 41 |
| (ci-new,CN2) | 0 1 5 6 7 8 9 10 11 12 14 | 11 | 11 | 100.00 | 68.9 | |

Table 6.4.3: Smoke and Derived Datasets – Selected Features, Time for Selecting Features and Number of Duplicate or Conflicting Instances

| Dataset | smoke | | smoke-mlc | | | | smoke01 | | | | smoke02 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Feature Number | (ori-all, C4.5r) | (ori-all, $\mathcal{CN}2$) | (ci-all, C4.5r) | (ci-all, $\mathcal{CN}2$) | (ci-new, C4.5r) | (ci-new, $\mathcal{CN}2$) | (ci-all, C4.5r) | (ci-all, $\mathcal{CN}2$) | (ci-new, C4.5r) | (ci-new, $\mathcal{CN}2$) | (ci-all, C4.5r) | (ci-all, $\mathcal{CN}2$) | (ci-new, C4.5r) | (ci-new, $\mathcal{CN}2$) |
| #0 | o | o | o | o | o | o | o | o | o | o | o | o | o | o |
| #1 | o | o | o | o | o | o | o | o | o | o | o | o | o | o |
| #2 | o | o | o | o | | | o | o | o | o | o | o | | |
| #3 | o | o | o | o | | | o | o | o | o | o | o | | |
| #4 | o | o | o | o | | | o | o | o | o | o | o | | |
| #5 | o | o | o | o | | | o | o | | | o | o | o | o |
| #6 | o | o | o | o | | | o | o | | | o | o | o | o |
| #7 | | o | o | o | | | o | o | | | o | o | o | o |
| #8 | o | o | o | o | | | o | o | | | o | o | o | o |
| #9 | o | o | o | o | o | o | o | o | o | o | o | o | o | o |
| #10 | o | o | o | o | o | o | o | o | o | o | o | o | o | o |
| #11 | o | o | o | o | o | o | o | o | o | o | o | o | o | o |
| #12 | o | o | o | o | o | o | o | o | o | o | o | o | o | o |
| #13 | | | • | • | • | • | | • | • | • | | | | |
| #14 | | | • | • | • | • | | | | | • | • | • | • |
| # Selected F | 12 | 13 | 15 | 15 | 8 | 8 | 13 | 14 | 10 | 10 | 14 | 14 | 11 | 11 |
| Total # F | 13 | 13 | 15 | 15 | 8 | 8 | 14 | 14 | 10 | 10 | 14 | 14 | 11 | 11 |
| % F | 92.31% | 57.40% | 100% | 100% | 100% | 100% | 92.86% | 100% | 100% | 100% | 100% | 100% | 100% | 100% |

Table 6.4.4: Smoke Before and After Constructive Induction – Selected Features

| | C4.5rules | $\mathcal{CN}2$ |
|---|---|---|
| smoke 10-strat-cv | | |
| ori-all | 32,71±0,65 | 31,87±0,35 |
| smoke-mlc 10-strat-cv | | |
| ci-all | 32,58±0,46 | 31,59±0,50 |
| ci-new | 32,72±0,38 | 32,09±0,40 |
| smoke01 10-strat-cv | | |
| ci-all | 33,28±0,80 | 31,56±0,45 |
| ci-new | 33,24±0,37 | 32,25±0,88 |
| smoke02 10-strat-cv | | |
| ci-all | 32,93±0,49 | 31,49±0,45 |
| ci-new | 32,37±0,46 | 31,56±0,46 |

Table 6.4.5: Smoke and Derived Datasets – Error Rate

| | C4.5rules | $\mathcal{CN}2$ |
|---|---|---|
| smoke-mlc | | |
| ci-all | -0,23 | -0,65 |
| ci-new | 0,02 | 0,59 |
| smoke01 | | |
| ci-all | 0,78 | -0,77 |
| ci-new | 1,00 | 0,57 |
| smoke02 | | |
| ci-all | 0,38 | -0,94 |
| ci-new | -0,60 | -0,76 |

Table 6.4.6: Difference in Standard Deviations Between Smoke and Derived Datasets

## 6.5 Hepatitis and Derived Datasets

One new feature was constructed for this dataset with the help of the specialist. Its structure is shown bellow.

- f1(New-fc01): indicates if the pacient probably will live or die. It combines three primitive features: *liver-firm*, *ascites* and *varices*.

```
if (liver-firm=yes and ascites=yes and varices=yes)    % pacient has liver-firm,
                                                       % ascites and varices
   then new-fc01=0                                     % probably will die
   else new-fc01=1                                     % probably will live
```

| Feature Number | Feature Name | #Distinct Values | | |
|---|---|---|---|---|
| | | possible | actual | type |
| #0 | age | - | 49 | continuous |
| #1 | female | 2 | 2 | nominal |
| #2 | steroid | 2 | 3 | nominal |
| #3 | antivirals | 2 | 2 | nominal |
| #4 | fatigue | 2 | 3 | nominal |
| #5 | malaise | 2 | 3 | nominal |
| #6 | anorexia | 2 | 3 | nominal |
| #7 | liver-big | 2 | 3 | nominal |
| #8 | liver-firm | 2 | 3 | nominal |
| #9 | spleen-palpable | 2 | 3 | nominal |
| #10 | spiders | 2 | 3 | nominal |
| #11 | ascites | 2 | 3 | nominal |
| #12 | varices | 2 | 3 | nominal |
| #13 | bilirubin | - | 34 | continuous |
| #14 | alk-phosphate | - | 83 | continuous |
| #15 | sgot | - | 84 | continuous |
| #16 | albumin | - | 29 | continuous |
| #17 | protime | - | 44 | continuous |
| #18 | histology | 2 | 2 | nominal |
| new#19 | f1(*new-fc01*) | - | 2 | nominal |

Table 6.5.1: Hepatitis and Derived Datasets – Feature Description

| Feature Number | Datasets | |
|---|---|---|
| | hepatitis | hepatitis01 |
| #0 | ◇ | ◇ |
| #1 | ◇ | ◇ |
| #2 | ◇ | ◇ |
| #3 | ◇ | ◇ |
| #4 | ◇ | ◇ |
| #5 | ◇ | ◇ |
| #6 | ◇ | ◇ |
| #7 | ◇ | ◇ |
| #8 | ◇ | ◇ |
| #9 | ◇ | ◇ |
| #10 | ◇ | ◇ |
| #11 | ◇ | ◇ |
| #12 | ◇ | ◇ |
| #13 | ◇ | ◇ |
| #14 | ◇ | ◇ |
| #15 | ◇ | ◇ |
| #16 | ◇ | ◇ |
| #17 | ◇ | ◇ |

*continued on next page*

*continued from previous page*

| Feature Number | Datasets | |
|---|---|---|
| | hepatitis | hepatitis01 |
| #18 | ◇ | ◇ |
| new#19 | | ◇ |

Table 6.5.2: Hepatitis and Derived Datasets – Original and Constructed Features

| | Selected Features | Total # F | # Selected F | %F | Time (s) | #Duplicate or Conflicting Instances |
|---|---|---|---|---|---|---|
| hepatitis | | | | | | |
| (ori-all,C4.5-rules) | 0 1 3 4 5 7 8 10 11 15 16 17 | 19 | 12 | 63.16 | 0.1 | 0 |
| (ori-all,CN2) | | 19 | | | | |
| hepatitis01 | | | | | | |
| (ci-all,C4.5-rules) | 0 1 4 5 7 8 10 11 15 16 | 20 | 10 | 50.00 | 0.9 | |
| (ci-all,CN2) | 0 1 7 10 11 12 13 14 15 16 17 18 19 | 20 | 13 | 65.00 | 1.2 | |
| (ci-new,C4.5-rules) | 2 4 5 6 7 9 14 15 16 17 | 17 | 10 | 58.82 | 0.0 | 0 |
| (ci-new,CN2) | 0 4 7 9 10 13 14 15 16 17 18 19 | 17 | 12 | 70.59 | 1.1 | |

Table 6.5.3: Hepatitis and Derived Datasets – Selected Features, Time for Selecting Features and Number of Duplicate or Conflicting Instances

| Dataset | hepatitis | | hepatitis01 | | | |
|---|---|---|---|---|---|---|
| Feature Number | (ori-all, $\mathcal{C}$4.5r) | (ori-all, $\mathcal{CN}$2) | (ci-all, $\mathcal{C}$4.5r) | (ci-all, $\mathcal{CN}$2) | (ci-new, $\mathcal{C}$4.5r) | (ci-new, $\mathcal{CN}$2) |
| #0 | ○ | | ○ | ○ | | ○ |
| #1 | ○ | | ○ | ○ | | |
| #2 | | | | | ○ | |
| #3 | ○ | | | | | |
| #4 | ○ | | ○ | | ○ | ○ |
| #5 | ○ | | ○ | | ○ | |
| #6 | | | | | ○ | |
| #7 | ○ | | ○ | ○ | ○ | ○ |
| #8 | ○ | | ○ | | | |
| #9 | | | | | ○ | ○ |
| #10 | ○ | | ○ | ○ | | ○ |
| #11 | ○ | | ○ | ○ | | |
| #12 | | | | ○ | | |
| #13 | | | | ○ | | ○ |
| #14 | | | | ○ | ○ | ○ |
| #15 | ○ | | ○ | ○ | ○ | ○ |
| #16 | ○ | | ○ | ○ | ○ | ○ |
| #17 | ○ | | | ○ | ○ | ○ |
| #18 | | | | ○ | | ○ |
| #19 | | | | • | | • |
| # Selected F | 12 | | 10 | 13 | 10 | 12 |
| Total # F | 19 | 19 | 20 | 20 | 17 | 17 |
| % F | 63.16% | | 50.00% | 65.00% | 58.82% | 70.59% |

Table 6.5.4: Hepatitis Before and After Constructive Induction – Selected Features

| | $\mathcal{C}$4.5rules | $\mathcal{CN}$2 |
|---|---|---|
| hepatiti 10-strat-cv | | |
| ori-all | 21,29±2,99 | 18,25±3,83 |
| hepatiti01 10-strat-cv | | |
| ci-all | 18,00±3,74 | 17,50±2,04 |
| ci-new | 25,83±3,33 | 17,51±1,49 |

Table 6.5.5: Hepatitis and Derived Datasets – Error Rate

| | $\mathcal{C}$4.5rules | $\mathcal{CN}2$ |
|---|---|---|
| hepatiti01 | | |
| ci-all | -0,97 | -0,24 |
| ci-new | 1,43 | -0,25 |

Table 6.5.6: Difference in Standard Deviations Between Hepatitis and Derived Datasets

# 7 Some Considerations

It is interesting to observe the numbers in the third table related to the column *Duplicate or Conflicting Instances* of each dataset — Sections 6.2 to 6.5. Note that for datasets cmc and smoke, in some cases these numbers changed (cmc: from 115 to 167 and 240; smoke: from 29 to 41) when the primitive features that were used to construct new features were removed.

In all cases considered, this number has grown up. A possible explanation for this is that the removed primitive feature probably is the only feature that has different values for one or more instances, and as a consequence of its removing, more instances became conflicting.

For example, consider the set of instances given by Table 7.7 were a new feature $X_{new}$, defined as:

$$\texttt{if } X_1 = X_3 \texttt{ then } X_{new} = \texttt{True else } X_{new} = \texttt{False}$$

has been constructed.

| Instances | $X_1$ | $X_2$ | $X_3$ | $X_4$ | $X_{new}$ | Class |
|---|---|---|---|---|---|---|
| $I_1$ | ♣ | ♢ | ♣ | ♡ | True | + |
| $I_2$ | ♢ | ♣ | ♡ | ♠ | False | + |
| $I_3$ | ♡ | ♢ | ♡ | ♡ | True | − |
| $I_4$ | ♡ | ♠ | ♠ | ♢ | False | − |

Table 7.7: Example of Duplicate or Conflicting Instances

Now, if we remove the primitive features that were used to construct this $X_{new}$ feature, the obtained set of instances is given by Table 7.8.

| Instances | $X_2$ | $X_4$ | $X_{new}$ | Class |
|---|---|---|---|---|
| $I_1$ | ♢ | ♡ | True | + |
| $I_2$ | ♣ | ♠ | False | + |
| $I_3$ | ♢ | ♡ | True | − |
| $I_4$ | ♠ | ♢ | False | − |

Table 7.8: Example of Duplicate or Conflicting Instances After Removing Primitive Features

Note that removing these two primitive features made instances $I_1$ and $I_3$ become conflicting, *i.e.* they both have the same values for all the features, but belong to different classes.

Table 7.9 presents a summary of the results obtained through the three steps performed in the experiments reported in this work. This table shows, for each one of the ten augmented datasets, the following information:

- A - the names of the datasets

- B - total number of features in the dataset
- C - the numbers which identify the primitive features (PrimF) used to construct the new one (NewF)
- D - features used by $\mathcal{C}$4.5rules
- E - features used by $\mathcal{CN}2$
- F - if all features of the dataset were selected
- G - if any of the new constructed features were selected
- H - if accuracies measured by 10-fold stratified cross-validation improved, using $\mathcal{C}$4.5rules and $\mathcal{CN}2$ on the augmented datasets. If so, this is indicated by the inducer that had the accuracies improved
- I -features used by $\mathcal{C}$4.5rules from the reduced datasets — Step 3
- J -features used by $\mathcal{CN}2$ from the reduced datasets — Step 3
- K -if accuracies measured by 10-fold stratified cross-validation improved, using $\mathcal{C}$4.5rules and $\mathcal{CN}2$ on the reduced datasets. If so, this is indicated by the inducer that had the accuracies improved

Note that features in underlined bold style correspond to the new constructed features.

As it can be seen, at the end of the third step, only two datasets showed an improvement of accuracies without the primitive features, which were used to compose the new constructed features. These two datasets are:

1. smoke using f2(Place)
2. hepatitis using f1(New-fc01).

| | Step 1 | | | | | | Step 2 | Step 3 | | |
|---|---|---|---|---|---|---|---|---|---|---|
| A | B | C<br>NewF → PrimF | D | E | F | G | H | I | J | K |
| pima-mlc | 10 | 3<br>8 → 1 2<br>9 → 1 4 | 1 2 4 5<br>6 7 | 0 1 2 3<br>4 5 6 7 **9** | | | | | | |
| pima01 | 9 | 2<br>8 → 1 2 | 1 2 5 6<br>7 **8** | 1 2 3 4<br>5 6 7 | | Yes | $\mathcal{C}$4.5rules | 0 4 5 6<br>7 **8** | 0 3 4 5<br>6 7 **8** | |
| pima02 | 9 | 2<br>9 → 1 4 | 1 2 4 5<br>6 7 **9** | 0 1 2 3<br>4 5 6 7<br>**9** | Yes | Yes | | | | |
| cmc-mlc | 11 | 3<br>9 → 1 2<br>10 →1 7 | 0 1 2 3<br>4 5 6 7<br>8 | 0 1 2 3<br>4 5 6 7<br>8 **9 10** | Yes | Yes | | | | |
| cmc01 | 10 | 2<br>9 → 1 2 | 0 1 2 3<br>4 5 6 7<br>8 | 0 1 2 3<br>4 5 6 7<br>8 **9** | | Yes | $\mathcal{CN}2$ | 0 3 4 5<br>6 7 8 **9** | 0 3 4 5<br>6 7 8 **9** | |
| cmc02 | 10 | 2<br>10 →1 7 | 0 1 2 3<br>4 5 6 7<br>8 **10** | 0 1 2 3<br>4 5 6 7<br>8 **10** | Yes | Yes | $\mathcal{C}$4.5rules | 0 2 3 4<br>5 6 8 **10** | 0 2 3 4<br>5 6 8 **10** | |
| smoke-mlc | 15 | 7<br>13 → 5 6 7 8<br>14 → 2 3 4 | 0 1 2 3<br>4 5 6 7<br>8 9 10<br>11 12<br>**13 14** | 0 1 2 3<br>4 5 6 7<br>8 9 10<br>11 12<br>**13 14** | Yes | Yes | $\mathcal{C}$4.5rules<br>$\mathcal{CN}2$ | 0 1 9 10<br>11 12<br>**13 14** | 0 1 9 10<br>11 12<br>**13 14** | |
| smoke01 | 14 | 4<br>13 → 5 6 7 8 | 0 1 2 3<br>4 5 6 7<br>8 9 10<br>11 12 | 0 1 2 3<br>4 5 6 7<br>8 9 10<br>11 12 **13** | Yes | Yes | $\mathcal{CN}2$ | 0 1 2 3<br>4 9 10 11<br>12 **13** | 0 1 2 3<br>4 9 10 11<br>12 **13** | |
| smoke02 | 14 | 3<br>14 → 2 3 4 | 0 1 2 3<br>4 5 6 7<br>8 9 10 11<br>12 **14** | 0 1 2 3<br>4 5 6 7<br>8 9 10 11<br>12 **14** | Yes | Yes | $\mathcal{CN}2$ | 0 1 5 6<br>7 8 9 10<br>11 12 **14** | 0 1 5 6<br>7 8 9 10<br>11 12 **14** | $\mathcal{C}$4.5rules<br>$\mathcal{CN}2$ |
| hepatiti01 | 20 | 3<br>19 → 8 11 12 | 0 1 4 5<br>7 8 10<br>11 15 16 | 0 1 7 10<br>11 12 13<br>14 15 16<br>17 18 **19** | Yes | Yes | $\mathcal{C}$4.5rules<br>$\mathcal{CN}2$ | 2 4 5 6<br>7 9 14<br>15 16 17 | 0 4 7 9<br>10 13 14<br>15 16 17<br>18 **19** | $\mathcal{CN}2$ |

Table 7.9: Results Summary

A dataset showing an improvement of the accuracy at the end of the third step, means that this dataset not only has a better performance using a new constructed feature, but also that when the primitive features used to create this new feature were removed, the accuracy still remained better than the one obtained using just the original set of features. Although there were improvements in accuracy, the results would only fit into the perfect situation if during the first step these primitive features, used to construct the new ones, were not selected by the inducers. As said before, two possible reasons could be pointed out to justify this:

- the constructed features do not capture perfectly the information embedded in each individual feature
- the datasets used in this work are such that the original features on its own are relevant for learning.

Results summarized in Table 7.9 also show that, in almost all the cases considered in the third step, features used by the inducers to express the learned concept are exactly all the given features in this step, *i.e.* the same ones used in the first step without the primitive features used to create the new constructed ones. Only datasets pima01 and hepatitis01 have a set of features selected in the third step which are different from the first step for both inducers. Only for $\mathcal{C}$4.5rules, the dataset cmc01 has a set of features selected in the third step which are different from the ones selected on the first step. This may indicate that the new constructed features, in these two cases, played a more important role in the concept learning task.

Note that when the primitive features, that were used to create the new features, were removed, the set of features selected by $\mathcal{C}$4.5rules for pima01 has two different features: 0 and 4. Considering $\mathcal{CN}2$, during the first step, the new constructed feature was not used, but when the reduced dataset was given to this same inducer, besides feature 8 (new constructed feature), $\mathcal{CN}2$ also selected feature 0.

For dataset cmc01, only $\mathcal{C}$4.5rules showed a different set of features selected in the first step from the set selected in the third step. In this case, the new constructed feature was not used in the first step but only in the third one.

Finally, the only dataset that exhibited different sets of features from the first step to the third one and an improvement in accuracies was the dataset hepatitis01. For $\mathcal{C}$4.5rules, in the third step, the inducer used features 2, 9 and 17 that were not present in the first step. For $\mathcal{CN}2$, in the third step, the inducer used features 4 and 9 that were not present in the first step.

Note that dataset smoke02 has accuracy improved for both inducers given only the set of features composed by the new constructed features and the primitive features that were not used to create these new features.

# 8 Conclusions

This work shows some empirical results of Knowledge-driven Constructive Induction. The Constructive Induction approach is based on domain knowledge provided by a user/specialist; given the primitive features of the original datasets, the user/specialist suggested freely the construction of some new features. Accuracy and the set of features selected were measured when given different sets of features to the two inducers $\mathcal{C}$4.5rules and $\mathcal{CN}2$. A feature is considered relevant for the learning task if it is used by these algorithms to induce the rules.

Results show that, in spite of having a user/specialist help, it is difficult to construct new features that are really relevant to learn the concept embedded in these datasets. It could be that the original features of these datasets are relevant on its own to learn the concept.

**Acknowledgments:** We are grateful to Dr. Wu Feng Chung for the valious help with the construction of new features for datasets pima, hepatitis and suggestions for dataset cmc. We also wish to thank José Augusto Baranauskas for many helpful comments on the draft of this report.

## References


Aha, D. W. (1997). Lazy learning. *Artificial Intelligence Review*, 11:7–10.

Baranauskas, J. A. and Monard, M. C. (1999). The $\mathcal{MLC}$++ wrapper for feature subset selection using decision tree, production rule, instance-based and statistical inducers: Some experimental results. Technical Report 87, ICMC-USP, São Carlos, SP. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_87.ps.zip.

Baranauskas, J. A. and Monard, M. C. (2000a). Reviewing some machine learning concepts and methods. Technical Report 102, ICMC-USP, São Carlos, SP. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_102.ps.zip.

Baranauskas, J. A. and Monard, M. C. (2000b). An unified overview of six supervised symbolic machine learning inducers. Technical Report 103, ICMC-USP, São Carlos, SP. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_103.ps.zip.

Baranauskas, J. A., Monard, M. C., and Horst, P. S. (1999). Evaluation of feature selection by wrapping around the CN2 inducer. In *Proceedings Argentine Symposium on Artificial Intelligence - ASAI'99*, pages 141–154.

Blake, C., Keogh, E., and Merz, C. (1998). UCI Irvine Repository of Machine Learning Databases. http://www.ics.uci.edu/~mlearn/MLRepository.html.

Bloedorn, E. and Michalski, R. S. (1998). Data-Driven Construtive Induction. *IEEE Intelligent Systems*, 13(2):30–37. March/April 1998.

Bull, S. (1994). Analysis of attitudes toward workplace smoking restrictions. *Case Studies in Biometry*, pages 249–271.

Clark, P. and Boswell, R. (1991). Rule induction with CN2: Some recent improvements. In Kodratoff, Y., editor, *Proceedings of the 5th European Conference EWSL 91*, pages 151–163. Springer-Verlag.

Clark, P. and Niblett, T. (1987). Induction in noise domains. In Bratko, I. and Lavrač, N., editors, *Proceedings of the 2nd European Working Session on Learning*, pages 11–30, Wilmslow, UK. Sigma.

Clark, P. and Niblett, T. (1989). The CN2 induction algorithm. *Machine Learning*, 3(4):261–283.

Felix, L. C. M., Rezende, S., Doi, C. Y., de Paula, M. F., and Romanato, M. J. (1998). $\mathcal{MLC}$++ biblioteca de aprendizado de máquina em $\mathcal{C}$++. Technical Report 72, ICMC-USP, São Carlos, SP. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_72.ps.zip.

Kohavi, R., Sommerfield, D., and Dougherty, J. (1994). *$\mathcal{MLC}$++: A Machine Learning Library in C++*. IEEE Computer Society Press.

Kohavi, R., Sommerfield, D., and Dougherty, J. (1996). Data mining using $\mathcal{MLC}$++: A amchine learning library in $\mathcal{C}$++. *Tools with IA*, pages 234–245.

Lee, H. D. (1999). Seleção e construção de features relavantes para o aprendizado de máquina. Monografia apresentada para o exame de qualificação, ICMC-USP.

Lee, H. D., Monard, M. C., and Baranauskas, J. A. (1999). Empirical comparison of wrapper and filter approaches for feature subset selection. Technical Report 94, ICMC - USP, São Carlos, SP. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_94.ps.zip.

Michalski, R. (1978). Pattern recognition as knowledge-guided computer induction. Technical Report 927, Department of Computer Science – University of Illinois, Urbana-Champaign, Ill.

Prati, R. C., Baranauskas, J. A., and Monard, M. C. (1999). BibView: Um sistema para auxiliar a manutenção de registros para o BibTex. Technical Report 95, ICMC-USP. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_95.ps.zip.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, 1(1):81–106.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann. San Francisco, CA.

Wnek, E. B. J. and Michalski, R. S. (1993). Multistrategy constructive induction. In Kaufmann, M., editor, *Proceedings of the Second International Workshop Machine Learning – ML93*, pages 188–203, San Francisco.

Wnek, J. and Michalski, R. S. (1994). Hypothesis-Driven Constructive Induction in AQ17-HCI: A Method and Experiments. *Machine Learning*, 14(2):139–168.