

A comparison of some classification techniques

P. S. S. Coelho^{1,2} & N. F. F. Ebecken²

¹*Faculdades Ibmecc, Brazil*

²*COPPE – UFRJ, Brazil*

Abstract

The classification activity assigns labels, or classes, to differentiate object groups. In general, these labels are well known beforehand through objects already classified. In Data Mining tasks, the objects are records, i.e., they are described using a set of attributes. These attributes can have any nature (categorical or continuous). The objective is to establish models to characterize the classes of the records using its attributes (values, distribution, pattern, etc.). Many different techniques for the record classification task are available today. These techniques are differentiated by the heuristics they use. In this article a comparison is made of some of the most popular classification techniques. This includes Decision Trees, Bayesian Algorithms (Statistical Methods), and the Classification Based on Rule Induction, also Classification Based on Association Rules. To compare these techniques, the Predictive Accuracy Criteria was mainly used. The Speed, Robustness, Scalability and Interpretability Aspects are also argued, but they had not been quantified for a mathematical comparison. The classification models had been determined from two relational tables with real data. The first one of them is composite with data about meteorological conditions in the region of the International Airport of Rio de Janeiro. This table has 26482 records with 19 variables (one of them is the class label). The second one is about an insurance company, having 130143 registers with 63 independent variables (attributes) and one dependent variable (label of the class). These data tables were prepared earlier. The result of this comparison can be seen in some tables.

1 Introduction

It can be considered that the activities of Data Mining are concentrated in development of models that represent some knowledge contained in the data

where the activity (mining) was developed. These models can, in a more immediate analysis, be classified in three main groups: Association Rules, Classification and Prediction and Clustering. Historically the activities of Clustering are the oldest [5], and the Association Rules were discovered in the beginning of the 1990's [1].

There are two very distinctive forms to construct a classification model. It can be obtained by interviewing specialists – many systems based on knowledge are constructed in this way, despite the involved difficulties in this approach. Alternatively, recorded classifications might be automatically examined, and a model can be constructed inductively, generalizing these specific examples [9]. This second form to construct models is the activity that in Data Mining is called Classification.

Classification is a process carried through in two steps. In the first step, the model is constructed describing a predetermined set of classified data. This process is made through the analysis of the databases records, which are described by attributes. Each register is assumed to belong in a predefined class, that is one of its attributes, called class label attribute. In the context of the classification, records are related as samples, examples or objects. The registers analyzed for the construction of the model collectively form the training data set. Each element of this set is called training sample, training records, and are randomly selected from the population of records. As the class label of each training register is known, this step is also known as supervised learning.

In the second step, the constructed model is used for classification. Generally, the first classifications are used to estimate the accuracy of the model, through a test set. The test set has records with same structure (attributes amount and type) to the training ones. The test records are also randomly selected too, and are independent of the training ones (it is common to divide the sample population in two sets without intersection – disjoint). For each test sample, the known class label is compared with the label classified by the model. The accuracy is estimated as the percentage of test samples that are correctly classified.

The classification activities play an important role in Computer Science, because of its already known application in the Machine Learning/Artificial Intelligence tasks. For Data Mining activities, the applications include credit approval, medical diagnostics, forecast of performance and selective marketing, amongst others. There are many different techniques for the classification task: decision trees, bayesian classify and bayesian networks (belief networks), neural networks, rule induction, classification based in proximity (KNN – k-nearest neighbor), genetic algorithms, rough sets, and fuzzy logic, among others. New techniques have been developed from the merging of some classic techniques, mainly using the logic fuzzy (decision trees fuzzy, neuro-fuzzy networks, fuzzy rule induction).

Because of the many options, the choice of the best technique to solve the problem in question can be an arduous task, demanding the experiment of many possibilities before defining which type of model will have to be used. In this selection, questions as to accuracy, speed (computational cost), robustness

(ability ahead of noises), scalability (efficiency ahead of a great volume of data) and interpretability (understanding level) are crucial.

This article covers one brief argument to these questions, specifically the accuracy applied to some of the main techniques of Data Mining, which are briefly described in the section, Technical Overview. Some software has been used, as can be seen in the Experiments section. This section has a detailed description of the databases used, including its physical features and a historical procedures description. At the end is the section with a final summary and conclusions.

2 Technical Overview

One of the more time consuming stages of all the processes of acquisition of the knowledge contained in a database is the data preparation, or pre-processing stage. This stage has a series of procedures that can be crucial for the success of the main stage of Data Mining (modeling). A vast and more specific reading on the question of the data preparation can be seen in [8].

For the classification techniques, the necessary data preparation activities are cleanliness (reduction of noise and missing values), analysis of relevance (irrelevant or redundant attributes elimination) and data transformation (generalization, discretização, substitution of alphanumeric values and normalization). In the techniques described above, some of this data preparation is necessary, and others are optional.

The classification techniques that we use can be grouped into three main types: decision trees, bayesian classifiers (naïve) and based in neighborhood (statisticians). We will make a superficial description of each one of the techniques.

2.1 Decision Tree

Decision Trees are a powerful and popular tool for classification tasks. They have a simple form of knowledge representation in the form of rules. This make the inferred model relatively easy to understand by the user. The tree structure is like a flow chart, where each internal node represents a test in an attribute, each branch represents a result of the test, and each leaf represents label class or, possibly, a distribution of probabilities of the classes.

An example of decision tree can be seen in Figure 1, which classifies possible customers of a fictitious store, where the internal node is represented by rectangles, and leaf nodes are represented with circles. To classify an unknown record, its attributed values are tested in the tree. A path is defined from the root to a leaf that contains the label class prediction for this sample. It can be observed that many leaf nodes can make the same classification, but as each leaf has an exclusive path for the root (a tree is an acyclic graph) the classification has a specific set of conditions.

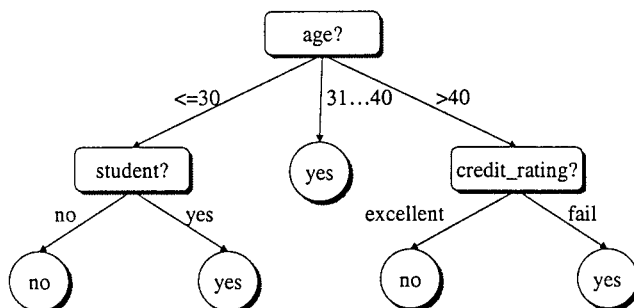


Figure 1: Decision Tree. Font: [3].

It can also be considered that a decision tree represents a series of questions. The reply of each question determines which will be the following question. If good questions were chosen, only one small amount of questions will be enough to classify a register. All attributes must be categorical, or have a small set of previously known discrete values,.

The basic strategy for construction of a tree is as follows [9]: the tree starts as a simple node representing all the samples of the training; if all the samples are of the same class, the node transforms into a leaf and this is labeled with the class; or (there are many classes in the sample set), the algorithm uses a measure based on entropy (information gain) as a heuristic to select the attribute that will better distinguish the samples in individual classes. This attribute becomes the “test” of the internal node. A branch is created for each known value of the attribute test, and the samples are divided between these branches; the algorithm processes the same recursively to construct a new (sub) tree from the samples obtained in each division. An attribute already used as a test cannot be considered again. The process finishes when one of the following conditions becomes true: all the samples in one determined node belong to the same class, or more attributes do not exist to divide. In this in case, the majority of samples in the branch can impose the classification.

The information gain measure is used to select the attribute to be used as a test in each node of the tree. The attribute with the biggest information gain, or with the best entropy reduction, is chosen to be tested in the current node. This attribute minimizes the amount of information necessary to classify the result sample.

During the tree construction, many branches will reflect anomalies in the training of the data in function of noises and outliers. The method of pruning the tree resolves the data overfitting problem. This method generally uses statistics measures to remove the uncertain branches, generally resulting in a faster classification and with bigger **probabilities** of correctly new data classifying.

The efficiency of algorithms of a decision tree, such as ID3 and C4.5[9], already are considered well established for small data sets[3]. However, when the work happens with databases of the real world, or with data bases with much information, aspects such as efficiency and scalability become problematic. This is because these algorithms have the restriction that the training samples are

usually in the main memory. This restriction limits the scalability of these algorithms, where the decision tree construction can become inefficient in its function of the data set handling.

2.2 Naïve Bayesian Classifier

Bayesian classifiers are statistics ones, which can predict the probability of the occurrence of each class as the probability of one given sample belonging to a specific class. This technique is based on Bayes Theorem. Studies comparing classification algorithms have pointed out that naïve bayesian classifier can be compared with the decision trees and neural network approaches in performance aspects [3]. One of the most important features of the Bayesians classifiers is the precision and speed when applied to large databases.

Naïve bayesian classifier assumes that the effect of the value of an attribute in one determined class is independent of the value of other attributes in the same class. This assumption is known as conditional independence of the classes and is assumed to reduce the computational effort.

The bayesian classifier works in the following way: each sample is represented as n -dimensional feature vector, $X = (x_1, x_2, \dots, x_n)$, reflecting n sample measures. These measures are made from attributes, $A_1, A_2, \dots, A_n$. Suppose to exist m classes, $C_1, C_2, \dots, C_m$. Given an unknown sample, X , the classifier will determine X to be a class with the biggest *a priori* probability conditioned by X . This is, the unknown sample X is classified as C_i if, and only if, $P(C_i | X) > P(C_j | X)$, for $1 \leq j \leq m, j \neq i$.

Thus, we maximize $P(C_i | X)$. According Bayes Theorem, $P(C_i | X) = \frac{P(X | C_i)P(C_i)}{P(X)}$. As $P(X)$ is constant for all classes, just $P(X | C_i)P(C_i)$ need be maximized. If the classes probabilities are unknown, it is possible to define $P(C_1) = P(C_2) = P(C_m)$, and would therefore maximize $P(X | C_i)$. The class prior probabilities may be estimated as $P(C_i) = \frac{s_i}{s}$, where s_i is the number of training samples classified as C_i , and s is the total training set size.

When the dataset has many attributes, it would be extremely computationally expensive to compute $P(X | C_i)$. To reduce this computational effort, it is common to make the assumption that the values of the attributes are conditionally independent of one another, given the class label of the sample, that is, there are no dependence relationships among the attributes. Thus,

$$P(X | C_i) = \prod_{k=1}^n P(x_k | C_i)$$

578 *Data Mining III*

The probabilities $P(x_1 | C_i)$, $P(x_2 | C_i)$, ..., $P(x_n | C_i)$ could be estimated from training samples:

- If A_k is categorical (discrete), then $P(x_k | C_i) = \frac{s_{ij}}{s_i}$, where s_{ik} is the number of samples belonging the class C_i , with the value x_k to A_k attribute, and s_i is the total training samples belonging the class C_i .
- If A_k is continuous, then is possible assume the attribute has a Gaussian distribution,

$$P(x_k | C_i) = g(x_k, \mu_{C_i}, \sigma_{C_i}) = \frac{1}{\sqrt{2\pi} \cdot \sigma_{C_i}} e^{-\frac{(x_k - \mu_{C_i})^2}{2\sigma_{C_i}^2}},$$

where $g(x_k, \mu_{C_i}, \sigma_{C_i})$ is a Gaussian Function to A_k attribute, and μ_{C_i} and σ_{C_i} are the mean and standard deviation given the values for attribute A_k for training samples of class C_i .

To classify an unknown sample X , $P(X | C_i) P(C_i)$ is estimated to each class C_i . The sample X is classified as C_i if, and only if,

$$P(X | C_i) \cdot P(C_i) > P(X | C_j) \cdot P(C_j), \text{ for } 1 \leq j \leq m, j \neq i.$$

Bayesians classifiers have, in general, lower error rates in matching with the other classifiers. While Naïve Bayesian classifier assumes independence of the classes, the Bayesian Belief Networks specify distributions of conditional probabilities. They allow us to define conditional independences of the classes through subgroups of variables.

2.3 K-Nearest Neighbor

These classifiers are based on learning for analogy. The training samples are described for a n -dimensional numerical vector. Each sample represents a point in an n -dimensional space, called pattern space. When a non classified sample is supplied, the classifier searches in pattern space for the k samples that are closest to the unknown sample. This k training sample is the k "nearest neighbors" of the unknown sample. Closeness is defined in terms of Euclidian distance, where the distance between two points $X = (x_1, x_2, \dots, x_n)$ e $Y = (y_1, y_2, \dots, y_n)$ in a n -dimensional space, is defined as $d(X, Y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$.

The unknown sample is associated with the most common class among its k nearest neighbors. When $k = 1$, the unknown sample is associated with the same class of the sample of the training that it is closest to it in pattern space. Neighbor Nearest classifiers are lazy learners, because all the samples of training are stored and no model is constructed until a new and unknown sample needs to be classified. These classifiers are very fast in the stage of training (only storage),

but time consumer in the classification task. Normally, efficient techniques of indexation are required.

3 Experiments

The previously described techniques were experimented upon using a total of 8 software programs, each having commercial or academic purpose. To do this, we chose two databases, of small and medium size. The smaller one is a meteorological database (called Meteo). The larger one is a database with profiles of customers of a German insurance agency (called Insur). Below is a brief historical description of the bases, since their collection and up until the phases of data preparation.

The Meteo base was created in the 1950's. During 10 years (from 1951 until 1960), the Institute of Meteorological Research of the UNESP – São Paulo State University, collected hourly, 39 sets of meteorological data in the International Airport of Rio de Janeiro. This collection had generating a data set with approximately 88000 records (and 39 attributes). Initially stored in punched cards, with a series of limitations, the database was organized and prepared in [4] and [2], resulting in a table with 19 columns (including the class label) and 26482 registers. The attributes include information of the hour, day, month, year and meteorological measurements. The class has 9 levels, indicating the present time. Of these 9 levels only 7 had been present.

The base Insur was presented to the academic world in 1998, when it had a competition proposal in a congress [6]. This case uses data extracted from a private company of life insurance, Data Warehouse. The original files given by the company describe the relationship between customers, contracts of insurance and components of tariff. The tables had been unified by [2], generating a base with 80 columns and 147478 records. These records describe the customers, and the columns characterize personal information and data on the policy. On many attributes, less or none information are divulged. After the data preparation, the database size becomes 64 columns and 130143 strings. The class label has only two labels.

To all, 5 softwares based in trees had been used, all commercial ones: Mineset (SGI), Intelligent Miner (IBM), CART and XpertRule Miner. The only representative of the Naïve Bayesian classifiers is Roc, an academic classifier that posses some features to manipulate missing values and to set a prior probability. Notice that these features had not been used. To represent the Belief Networks we have separated BNSoftware, the KDD-Cup 2001 winner, also academic. Finally, representing the statistical classifiers, we use the commercial software PRW.

In a general way, the software had presented a good interface, although in some cases, sufficiently peculiar. XPertRule Miner has a graphical design for the data input robust, allowing connection directly with a DBMS, but it is not very intuitive, and the final accuracy was not calculated! The PRW also presents an interface partially established in wizard.

580 *Data Mining III*

In assessing speed, XPertRule was very slow when compared with the others software, and CART was found to be very fast. Roc, which was expected to be fast, did not show to as much advantage of these bases, having a medium behavior.

About scalability, a great problem with BNSoftware. They couldn't work with the databases. Initial parameters were not explained to set memory size. With a 256MB Windows system, the software reports the status as 'out of memory'. Roc and Mineset also have problems. Roc only accepted input with approximately 65000 registers. Curiously, beyond this margin it presents an error on the type of internal data, instead of the information of software limitation. To obtain the execution, this was the only case where the test base had the same so great that the training set, with the Insur database. With the reduced training set, the software became unstable during the classification phase, presenting unexpected errors, such as division by zero. It can be observed in Table 1 that only one result was obtained with the Insur database, against four results obtained with the Meteo database. Mineset, which we use in MS-Windows environment cannot open the Insur database, and in IRIX environment (native), cannot make Cross-Validation or Boost, and also presents an error without clear explanation.

To measure the accuracy of the software, we extract a sample of each one of the bases that was not supplied during the training. The test bases have 30% of the total of the original bases. The sample of training is the same for all the software, except Mineset, which has made its proper sampling. The Insur base for Roc also had separate sampling, therefore it was necessary to reduce the size of the training base. The test set has 50% of the total base.

Table 1 shows a summary of the tested performances of the software. Some software had been tested with different parameters and appear more than in a string. In the case of Mineset, we use the option of Cross Validation. The CART was tested with some options, but the result was always exactly the same.

Table 1: Accuracy of tested softwares

Software and options	Meteo	Insur
Mineset wo/ Cross Validation	84.17% $\pm$ 0.39%	94.77% $\pm$ 0.11%
Mineset w/ Cross Validation	84.78% $\pm$ 0.22%	-
Intelligent Data Miner	83.37%	88.27%
CART	74.16%	93.14%
XPertRule Miner	79.36%	74.75%
Roc normalization 10x4	68.67% / 68.69%	70.82% / 70.90%
Roc normalization 30x8	67.60% / 67.58%	-
PRW	75,28% ($k = 150$)	76,54% ($k = 300$)
PRW	76,05% ($k = 75$)	79.59% ($k = 150$)
PRW	76,41% ($k = 50$)	-

XPertRule did not allow options. Roc alone works with normalizing options. During the input of the data it is possible to define to each categorical variable the maximum amount of different values to be considered, and the number of level values, case is necessary to discretize. The Default is 10 values to be considered as discrete variable and the number of states to be created is 4 (10x4). We also use the option 30x8.

The accuracy related to Mineset is accomplished by standard deviation. And, the Roc values are duplicated because in the testing phase it is possible use a Strong (first value) or Weak (second one) stochastic dominance.

These bases had also been used in some previous work. Table 2 shows the best results acquired. More details on the involved techniques can be seen in the cited references. In experiments related in [2] the accuracy was not the main objective. Because of this, reinforcement algorithms like Cross Validation were not used.

Table 2: Accuracy of early tests

Technique and Reference	Meteo	Insur
Neural Network [2]	84.02%	58.64%
Decision Tree [7]	73.25%	61.59%

4 Conclusion

The results point to greater success using decision trees software. This result, from the viewpoint of interpretability, is an excellent result, and of all software tested is the only one that allows an interpretation of the knowledge contained in the model. However, these results still leave the matter of choice as to the system employed, even though some intrinsic features of the databases can invert the picture.

A careful examination of the available techniques, considering mainly the speed, escalabilidade and accuracy issues, is always necessary. The results must be seen as an indicating of that, and the various software cannot be left out of the selection process.

Moreover, the results show, with the exception of two suites, IBM and SGI, that software is still not sufficiently developed for bigger bases. This seems to be an inconsistency, because Data Mining argues, basically, escalabilidade. Data in the real world is not limited to an Excel spreadsheet. The advance of the computational capacities (hardware) allows software to be more efficient in extreme situations such as those shown.

References

- [1] Agrawal, R., Imielinsky, T. & Swami, A. Mining Association Rules between Sets of Items in Large Databases. In: *Proceedings of the 1993 ACM SIGMOD Conference*. Washington DC, EUA, 1993.

- [2] Costa, M.C.A. *Data Mining on High Performance Computing using Neural Networks*. In portuguese. DSc Thesis, COPPE – UFRJ: Rio de Janeiro, 1999.
- [3] HAN, J. & Kamber, M. *Data Mining Concepts and Techniques*. Morgan Kaufmann Publishers: San Francisco, 2001.
- [4] Hruschka, E. R. *A study about rule extraction from Neural Networks in Data Mining applications*. In portuguese. MSc Thesis, COPPE – UFRJ: Rio de Janeiro, 1998.
- [5] Kaufman L., & Rousseeuw, P.J. *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley & Sons: New York, 1990.
- [6] KDD-Sisyphus. *KDD Sisyphus Workshop – Data Preparation, Preprocessing and Reasoning for Real-World Data Mining Applications*. Internet: <http://research.swisslife.ch/kdd-sisyphus>, 1998.
- [7] Onoda, M. *Study on a decision tree algorithm coupled to a relational database system*. In portuguese. MSc Thesis, COPPE – UFRJ: Rio de Janeiro, 2001.
- [8] Pyle, D. *Data Preparation for Data Mining*. Morgan Kaufmann Publishers: San Francisco, 1999.
- [9] Quinlan, J.R. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers: San Mateo, 1993.
- [10] Ramoni, M. & Sebastiani, P. *Robust Bayesian Classification*. Technical Report TR-68. Knowledge Media Institute, The Open University. 1999.
- [11] UNICA Technologies, Inc. *Pattern Recognition Workbench 2.1 User's Guide*. Unica Technologies Press: Lincoln, 1996.