**Proceedings of the 2007 IEEE Symposium on
Computational Intelligence and Data Mining (CIDM 2007)**

Comparison of Classifiers Efficiency on Missing Values Recovering: Application in a Marketing Database with Massive Missing Data

Bruno M. Nogueira, Tadeu R. A. Santos and Luis E. Zárate
Applied Conputational Intelligece Laboratory - LICAP
Pontifical Catholic University of Minas Gerais
Belo Horizonte, Minas Gerais, Brazil
Email: bmnogueira@gmail.com, tadeu@javafree.com.br, zarate@pucminas.br

Abstract— **Missing data in databases are considered to be one of the biggest problems faced on Data Mining application. This problem can be aggravated when there is massive missing data in the presence of imbalanced databases. Several techniques as samples deletion, values imputation, values prediction through classifiers and approximation of patterns have been proposed and compared, but these comparisons do not consider adverse conditions found in real databases. In this work, it is presented a comparison of techniques used to classify records from a real imbalanced database with massive missing data, where the main objective is the database pre-processing to recover and select records completely filled for further techniques application. It was compared algorithms such as clustering, decision tree, artificial neural networks and Bayesian classifier, expressing their efficiency through ROC curves. Through the results, it can be verified that the problem characterization and database understanding are essential steps for a correct techniques comparison in a real problem. It was observed that artificial neural networks are an interesting alternative for this kind of problem since it was capable to obtain satisfactory results even when dealing with real-world problems.**

I. Introduction

On the last years, the KDD process (Knowledge Discovery in Databases) has been applied on several scientific and marketing areas in order to extract non-obvious knowledge and to support decision taking events. The KDD process can be interpreted as a set of steps which objective is to extract useful knowledge from data. Data Mining (DM) is the main step of the KDD process, in which some algorithms are applied in order to extract patterns from data through representative models [1].

Real-world databases frequently have some problems that impact on DM algorithms application [2]. Among these problems, it could be detached the occurrence of missing values occasioned by non-controlled circumstances. Missing data are the ones which values were not added to database but for which a real value exists on the ambient that they were extracted. The presence of missing data on databases is a common fact and could be distributed in various attributes, in a same instance (record) or randomly.

Missing values could generate serious problems on knowledge extraction and on Data Mining algorithms application. Missing values may hide important information about the dataset. Moreover, most of prediction methods used in Data Mining, such as Naïve Bayesian Classifier and Nearest Neighbor Classifier, can't deal with data that includes missing values [3].

There are several methods widely used to treat missing values [4]. The simplest of them is the deletion of samples or attributes that contain missing values from the database. This solution may cause loss of important information underlined on the present values and is only applicable in cases where missing values occurs in a small percentage of values or attributes.

Another possible solution is the imputation of values assisted by a domain expert. This is a valid option when the number of missing values and the database size are small. Other imputation method is the imputation based on database characteristics values, such as a global constant value, mean or mode and mean or mode per class. It is important to notice that all these imputation methods, even the most careful, may introduce distortion and generate distorted knowledge.

Finally, it is possible to estimate missing values building prediction models, using algorithms such as classifiers in order to estimate missing values. These models are constructed using the filled values of the database, establishing relations between variables and estimating the value of one variable in function of the other variables.

The restrictions caused by missing data can be aggravated in contexts which missing values occur massively. Some classification algorithms, for example, do not obtain satisfactory results when dealing with databases containing missing values in a large percentage [2], [5].

The present work presents a missing values recovering in a massive missing values context. It was compared the efficiency of four largely used classifiers: Artificial Neural Networks (ANN) trained with backpropagation [17], C4.5 Decision Tree algorithm [18], Naïve Bayesian Classifier and K-Means Clustering algorithm [20]. All the classifiers were applied on a same real-world imbalanced database that contains massive missing data, which has passed through a preprocessing stage that contains some activities to assure the correct missing value

prediction and, so, make available as much correct values as possible. In order to obtain a non-biased comparison, it was constructed ROC curves to each classifier, comparing their efficiency through the AUC measurement [6].

The database used in this work comes from a survey made with textile retail businessmen. Initially, each considered record is a store that answered the survey (634 stores) and each attribute is equivalent to a survey field (totalizing 71). Due to different reasons, some survey fields were not filled, generating a database with great percentage of missing data (23,5%) and 0% of records completely filled. Among these data, it was verified that the field related to the store annual <income> contains a great absence percentage (33%). As a result of the importance of this field information to obtain the stores profile, the classification process in this work intends to estimate the income interval to which its values belong.

II. UNDERSTANTING THE DATABASE

In [7] it was established that an ontological analysis through domain understanding and problem characterization are essential steps for a correct application of a KDD process and for an efficient comparison of techniques in a real problem. In this section the database analysis, assisted by a domain expert, is presented. The goal is to obtain a representative data structure of the domain, which allows it to obtain a data set completely filled without missing values. The usage of a domain expert is essential to obtain a deep comprehension about the data [8], [9]. Later, that set will be used by the data classification algorithms to estimate the income interval to which its values belong.

The database used on the present work comes from a survey made with textile retail businessmen. This database contains missing data in great proportions (23,5%) where 0% of the records are completely filled. Initially, each considered record is a store that answered the survey (634 stores) and each attribute is equivalent to a survey field (totalizing 71). Some main attributes of the database can be observed at Table I.

TABLE I
EXAMPLE DATABASE MAIN ATTRIBUTES

Attribute	Description
<com_name>	Store commercial name
<address>	Store address
<owner_gen>	Store owner gender
<prop_name>	Store owner name
<region_appeal>	Store location region advantages
<cli_stratum>	Clients stratum
<income>	Annual income
<appeal>	Region appeal
<difficulty>	Region difficulties
<parking>	Parking
<evening_freq>	Evening client frequency
<charge_period>	Charge period
<client_type>	Client type

The first step is to identify the attributes related to the problem domain through data structure determination. The data structure can be supported by propositional logic that permits reasoning about variables relation into a problem domain. This will be presented on the next subsection.

A. Specifying Properties Through Propositional Logic

This section describes how propositional logic can be used to identify attributes related to ẗextile market saleḋomain. Atomic propositions and their relationships are specified in order to define static structure (attributes) of the problem domain. The main idea is to find the set of attributes related to <*income*>.

In order to write specifications that describe context properties is necessary to define a set of *atomic propositions AP*. An atomic proposition is an expression that has the form $v\ op\ d$ where $v \in V$ - the set of all variables in the context, $d \in D$ - the domain of interpretation, and *op* is any relational operator. To describe sequences of transitions along time, Temporal Logic is a very useful formalism. With temporal logic it is possible to reason about the system in terms of occurrences of events. For example, it can be reason if a given event will *eventually* or *always* occur.

There are several propositions of temporal logic. These logics vary according temporal structure (*linear* or *branching-time*) and time characteristic (*continuous* or *discrete*). Temporal linear logics reason about the time as a chain of time instances. Branching-time logics reason about the time as having many possible futures at a given instance of time. Time is continuous if between two instances of time there is always another instance. Time is discrete if between two instances of time a third one cannot be determined.

For the considered database, it is used a branching-time and discrete logic known as Computation Tree Logic (CTL) to express properties of systems. CTL provides operators to be applied over computation paths. When these operators are specified in a formula they must appear in pairs: path quantifier followed by temporal operator. A *path quantifier* defines the scope of the paths over which a formula f must hold. There are two path quantifiers: **A**, meaning **all** paths; and **E**, meaning **some** paths. A temporal operator defines the appropriate temporal behavior that is supposed to happen along a path relating a formula *f*. For example:

- F ("in the future" or "eventually") - f holds in some point of the computation path;
- G ("globally" or "always") - f holds in all path;

A well formed CTL formula is defined as follows:

1. If $p \in AP$, then p is a CTL formula, such that AP is the set of atomic propositions;

2. If f and g are CTL formulas, then $\neg f$, $f \vee \_g$, $f \wedge g$, AFf, EFf, AGf, EGf, A[fRg], E[fRg], A[fUg], E[fUg], AXf, EXf, are CTL formulas.

The selection of a set of variables (attributes) relevant to the problem domain is entirely based on the variables relationships:

- AG((*Appeal* & ¬*Difficulty*) ‖ (*Parking* & *Transport*)) → AF(*Income*));
- EF(*Evening Frequency* → EF(*Income*));
- A[*Charge Period* ∪ *Income*];

- AG((*Client Type* & *Frequency*) || *Client Stratum*) → AF(*Income*)).

Some attributes considered relevant to estimate the <income> interval to which its values belong, obtained through propositional logic, can be observed at Table I.

After identifying the main attributes of the data structure the following step is the database pre-processing. The aim is to obtain the set of representative records completely filled to apply the recovering algorithms.

III. PRE-PROCESSING OF THE DATABASE

In this section, the pre-processing stages such as recovering missing values, attributes removal, inconsistence analysis and outliers analysis, applied to the considered database, will be discussed. All these steps were applied to the database in order to assure the correct values prediction, recovering the missing values.

A. Missing Values Recovery by Related Attributes and Characteristics Deletion

Sometimes, it is possible to recover values that are not present on the database but are available in an indirect way. In the considered database, two actions were done:

- Values recovered by related attributes: on the database, it was possible to find some attributes representing the same information. So, it could be used the information from one of them to recover information of another attribute. For example, <owner_gen> attribute was recovered by <prop_name> attribute through gender analysis.
- Values recovered by default replacement: not filled attributes, which consider default values on the survey form were replaced by these values.

Some attributes do not have sufficient information to be considered. Thus, three criterions were established to eliminate attributes and records.

- Elimination of attributes containing short information. The entropy concept was applied for this aim [10].

$$H(s_1, s_2, ..., s_m) = -\sum_{i=1}^{m} p_i \, log_2 \, p_i \qquad (1)$$

Where: p_i is the s_i value occurrence probability to an attribute. Thus, for fields with nearly constant values: $p_i \approx 1$, so $H(s_i) \approx 0$. On the considered database, seven attributes were eliminated in this step.

- Attributes with large percentage of missing values do not help to characterize the problem domain. So, a threshold has been established to remove an attribute. On the present work, the threshold value was 25%, resulting on a sole attribute <cli_stratum> being removed.
- As done with attributes, records containing great quantity of missing data have to be eliminated. On the database considered, a threshold value of 25% was established to eliminate records. Fortysix records were removed from database.

In databases with a great number of attributes, some of them can be eliminated after a relevance classification process.

- Facts are those attributes which importance to the problem domain is considered to be more relevant because they have essential information. On the other hand, judgments are lower importance attributes, from which it could not be extracted relevant information during data analysis and, though, can be unconsidered. The distribution of relevance degree contained at an attribute is shown in Fig. 1. After a classification of the attributes, assisted by a problem domain expert, attributes considered Highly-Fact (HF), Fact (F) and Judgment-Fact (JF) are considered necessary, while Judgment (J) and Highly-Judgment (HJ) are discarded.

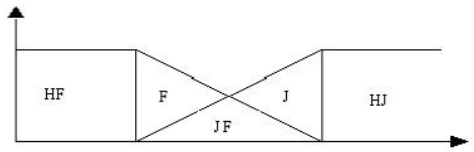

Fig. 1. Attribute importance classification

The removal is done in agreement with the following algorithm. On the example database a total of 27 attributes were removed.

```
FOR all attributes DO
     IF inf_atrib ==Judgment OR
        inf_atrib == Highly-Judgment
     THEN remove attribute
```

B. Attributes Transformation

In this section, it is presented a description of the transformations applied to the considered database, which contains dichotomic, nominal, categorical and ordinal attributes:

- The dichotomic fields were replaced by the equivalent binary number (0 or 1).
- The composite data <address> was transformed in data on the form (Longitude, Latitude) through GIS (Geographic Information System). To make possible a representative numerical classification to the addresses, the clustering technique was applied to the group the records (stores). The K-Means algorithm was chosen, defining 10 clusters a priori. Fig. 2 shows the stores grouping by localization.

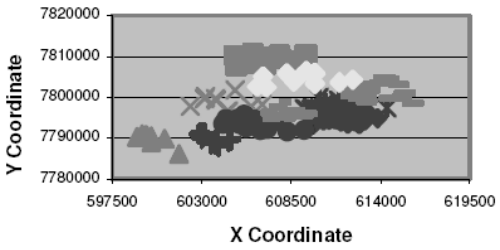

Fig. 2. Stores grouping by localization

- For attributes with multiple non-associated options, for example <region_appeal>, each option was considered an independent attribute, being transformed to dichotomic form (marking = 1, non-marking = 0). So, the number of database attributes was expanded to 81. To other cases that consist in multiple choices of associated options it is recommended the treatment of these data as circular data [11].
- Attributes separated by value ranges on the survey form or that consist on a simple option choice had arbitrary number associated to these values ranges or options. Four intervals were established to the attribute <income>, (2).

$$\begin{aligned} \text{Income } &\leq 61000 \in Inc_1 \\ 61000 < \text{Income} &\leq 123000 \in Inc_2 \\ 123000 < \text{Income} &\leq 377000 \in Inc_3 \\ \text{Income} &> 377000 \in Inc_4 \end{aligned} \quad (2)$$

Due to the problem domain characteristics, on which the information falsity (annual income) is widely latent, the four income intervals were reduced to two intervals through the domain expert experience. The tacit knowledge based rule applied on this database is expressed as follows:

Considering the limit set as:

$$\begin{aligned} Limits = \{LimI_i \in \Re, LimS_i \mid LimI_i < LimS_i, \\ i = 1..N, \text{ com } LimI_{i+1} = LimS_i\} \end{aligned} \quad (3)$$

and the income interval ϕ taken as:

$$\Phi = \{F_i; LimI_i \leq F_i < LimS_i, i = 1..N\} \quad (4)$$

The informed income value is defined as:

$$VInf = \{x \in F_k; p(x \in F_k) = p_k\} \quad (5)$$

where p_k is the x $\in$ F_k probability.
If x $\in$ F_k is a false information, so:

$$\begin{aligned} p(LimI_{k+1} \leq x < LimS_n) > p(x \in F_k) > \\ p(LimI_1 \leq x < LimS_{k-1}) \end{aligned} \quad (6)$$

In other words:

$$\begin{aligned} p(x < LimS_N) - p(x \leq LimS_{k-1}) > p_k > \\ p(x < LimS_{k-1}) - p(x \leq LimI_1) \end{aligned} \quad (7)$$

Thus, it is possible to reduce the income intervals without significant information loss.The new set of income intervals ϕ* may be expressed as:

$$\Phi* = \{F_i; LimI_i \leq F_i < LimS_i,\ i = 1..k, N\} \quad (8)$$

From which:

$$\begin{aligned} Income \leq 61000 \in Inc_1 * \\ 61000 < Income \in Inc_2 * \end{aligned} \quad (9)$$

With the objective of compare the potentiality of the classification techniques, in this analysis were considered two and four intervals to the attribute <income>.

C. Inconsistent Data Identification and Outliers Analysis

Due to the marketing nature of the database, the problem domain expert alerted that false information is a common fact, generating inconsistencies mainly on the fields related to the stores income, becoming more difficult the correct information recovery process. In order to detect these records inconsistencies, the clustering technique K-Means was applied separately under the pre-classified income groups.

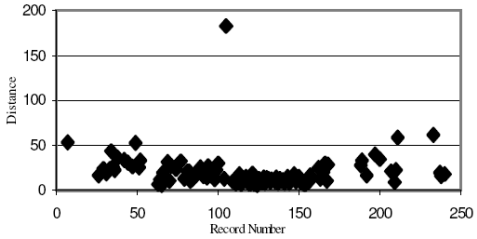

Fig. 3. Cluster 1 of income interval 1

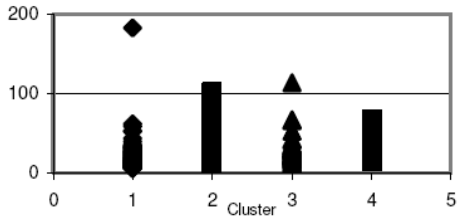

Fig. 4. Income interval 1 (4 clusters)

Fig. 3 shows the Cluster 1 for Inc_1 records distribution on the example database. Fig. 4 shows the distances between records and correspondent centroids to each interval. In both figures it is possible to observe the presence of discrepant elements (outliers), which were removed from the Inc_1 interval instances.

Objects removed from each interval obey the following rules:

$$RS_i = O_{jk} \in F_i | li_i \leq dist(O_{jk}, cent_{ki}) \leq ls_i \quad (10)$$

Where: RS is the selected record set for the income interval "i"; F_i is the income interval number "i"; O_{jk} is the object "j" attached to cluster k; $Cent_{jk}$ is centroid number for income interval number "i"; li_i is the inferior distance limit to the income interval number "i"; ls_i is the superior distance limit to the income interval number "i";

Each income interval Inc_i, li_i and ls_i is shown in Table II.

TABLE II
LIMIT DISTANCES TO RESPECTIVE CLUSTERS CENTROIDS

Inc_i, for i =	li_i	ls_i
1	0	60
2	10	20
3	20	50
4	20	30

IV. OBTAINING A SET OF REPRESENTATIVE RECORDS

On the considered database, after the pre-processing, 257 records (40% of the original database) and 81 attributes completely filled were obtained. As the information being recovered in that database is the one related to the stores annual income, the attribute <income> was taken as the goal of the classifiers algorithms, according to (2) and (3).

The records set was divided in two subsets: a training subset, which is presented to the classifiers in order to obtain a classification model, and a validation subset, which is used on the classifier accuracy. However, it is important to notice that the considered database contains much more records from one class than the other classes, generating an imbalanced domain that can induce some classifiers to error [12], [13], [14], [15]. Fig. 5 shows the heterogeneous distribution of the amount of records for the four income intervals.

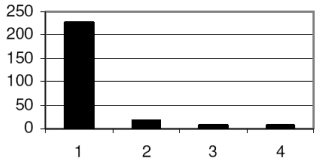

Fig. 5. Income interval x Frequency

After application of Consistent Sub-set Algorithm (CSS) proposed in [16], for artificial balancing of the databases, 110 records were selected as a set of representative records. The tested conditions are shown in Table III:

TABLE III
CONDITIONS TESTED

Test	Number of Intervals	Training Records Number	Validation Records Number	Selection Method
T1	2	110	147	Random
T2	4	110	147	Random
T3	2	110	147	CSS
T4	4	110	147	CSS

V. TECHNIQUES OF CLASSIFICATION

In this section, four representative classification techniques were analyzed: Clustering (K-Means), Decision Tree (C4.5), Artificial Neural Networks (Multi Layer Perceptron trained by backpropagation) and Bayesian Classifier (Naïve Bayesian Classifier).

A. Artificial Neural Networks Applications

To the considered database, two Multilayer Neural Networks [17] were trained. The networks were composed by perceptron neurons disposed in one hidden layer and one output layer totally connected. The used networks have 81 inputs each, with 4 and 2 outputs and 60 neurons on the hidden layer. The sigmoid function was chosen as activation function. All considered inputs represented by the E set (Table I) were mapped on 4 and 2 binary outputs mutually exclusive on the ANN (11)

$$f(E) \overset{ANN}{\rightarrow} (Inc_1, Inc_2, Inc_3, Inc_4) \tag{11}$$

For the neural networks training process, the numerical input data have to be submitted to a normalization process:

- Intending to improve the training process convergence, the data have to obey the normalization interval [0.2, 0.8];
- Data were normalized following the expressions:

$$\begin{aligned} f^a(L_0) &= L_n = (L_0 - L_{min})/(L_{max} - L_{min}) \\ f^b(L_n) &= L_0 = L_n * L_{max} + (1 - L_n) * L_{min} \end{aligned} \tag{12}$$

L_{min} and L_{max} were computed as follows

$$\begin{aligned} L_{min} &= L_{sup} - (N_s/(N_i - N_s)) * (L_{inf} - L_{sup}) \\ L_{max} &= ((L_{inf} - L_{sup})/(N_i - N_s)) + L_{min} \end{aligned} \tag{13}$$

where: L_{sup} is a variable maximum value, L_{inf} is the minimum value and N_i and N_s are the normalization limits (in this case, N_i = 0.2 and N_s = 0.8).

To begin the training process, random values between -1 and 1 were set on the connections weight.

As soon as the training process is finished, this training has to be validated. In this sense, the validation set (see Table III) was applied onto the neural network and its generated outputs were analyzed. For the tests T1 and T2, considering two and four income intervals, the accuracy rate was 89.88% and 87.82% respectively. For the tests T3 and T4, with artificial balancing of the databases, the accuracy rate was 98.05% and 91.82% respectively.

B. Decision tree application

Decision tree is one data-mining technique applied in many real-world applications as a powerful solution to classification problems. They use supervised learning methods that construct decision trees from a set of inputoutput samples. The algorithm used on the present work was the largely used C4.5 algorithm.

After the decision trees were generated, validation sets were presented to them in order to test the efficiency on the income classification. Considering the tests T1 and T2, the accuracy rate was 89.1% and 87.15%. For the tests T3 and T4, with artificial balancing of the databases, the accuracy rate was 91.05% and 91.43% respectively.

C. Cluster analysis application

The other technique used on the records classification was the classification pos - clustering. In order to group similar records, it was applied the K-Means algorithm, the simplest and most commonly used algorithm adopting the Euclidean distance criterion.

Using this technique for the tests T1 and T2, the accuracy rate was 91.43% and 30.35%. For the tests T3 and T4, with artificial balancing of the databases, the accuracy rate was 91.82% and 91.05% respectively.

D. Bayesian Classifier Application

Bayesian classification is based on the Bayes Theorem, which consists in a mathematical formula used to conditional probabilities calculus. The Naïve Bayesian Classifier, or Simple Bayesian Classifier, was used in this work. It assumes the existence of conditional independence between attributes and calculates the occurrence probability of a data sample given on a class. This classifier can be formally expressed as:

$$P(X/C_i) = \prod_{t=1}^{n} P(x_T/C_i) \quad (14)$$

Where:
X is a data sample whose class label is unknown;
C_i is a class value;
x_i are values for attributes in X;
P(X / C_i) is the occurrence probability of the sample X given the class C_i;
P(x_t / C_i) is the occurrence probability of the value x given the class C, that can be extracted from training dataset.

Thus, a new record can be classified by calculating the occurrence probability of the given data sample to each class, and assuming that the record belongs to the class with the major occurrence probability.

For the tests T1 and T2, considering two and four income intervals, the accuracy rate was 86.38% and 78.6% respectively. For the tests T3 and T4, with artificial balancing of the databases, the accuracy rate was 72.09% and 84.04% respectively.

VI. RESULTS COMPARISON

The results of techniques accuracy comparison shown in Fig. 6 and Fig. 7 show that Artificial Neural Networks had obtained much stable results, dealing well with the problem of imbalanced databases with two or four income intervals, obtaining the expressive recovering rate of 98.05% of the missing data in the experiments with the records set obtained through balancing algorithm.

It can be observed that the clustering technique was highly sensible to the effects of imbalanced database, caused by the records of the major class.

Other important fact that can be noticed is the instability of the Bayesian classifier when it is compared with the other techniques. This occurs because the algorithm needs that the records in the training set have large variety on its field values.

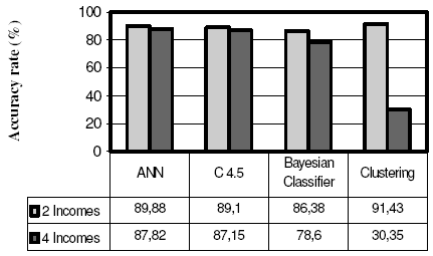

Fig. 6. Results obtained using the randomly selected training set (T1 and T2)

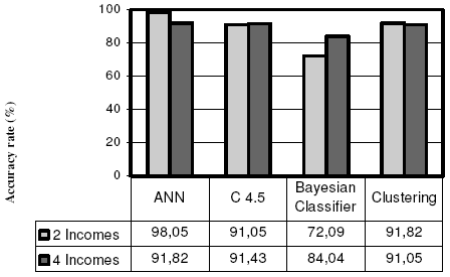

Fig. 7. Results obtained using the balanced training set (T3 and T4)

If it happens, the classifier cannot calculate all the possible values probability and it will not classify a certain record that have a field value that was not present in the training set.

A. ROC Curves Application

The efficiency of classifiers can be evaluated through the determination of the rates of true positives (elements correctly classified as positive class) and false positives (elements incorrectly classified as positive class) [19].

A ROC graph can be used to analyze the relation between sensitivity (efficiency when classifying positives) and specificity (efficiency when classifying negatives). It consists on the set of points generated by the classifiers validation statistics. Given the ROC curve, it is possible to obtain the accuracy rate, through the *Area Under Curve* (AUC) measurement [6]. The AUC value represents the fraction of the total area that is under the ROC curve. In this work, in order to obtain a trustworthy comparison that it was build one ROC curve to each one of the compared classifers considering the most favorable situation, with balanced database and two incomes interval (T2). Fig 8 shows the results obtained.

The AUC results obtained for each classifier can be observed in Table IV. It is possible to see differences between AUC values in Table IV and accuracy rates shown in Fig. 7. This is due to the fact that AUC measurement is capable to detect biased models which can correctly classify just one class or a set of classes. It is possible to see that C4.5 was very biased, presenting great percentage of false positives.

VII. CONCLUSIONS

The presence of missing values on a database is a common fact and can generate serious problems on the knowledge

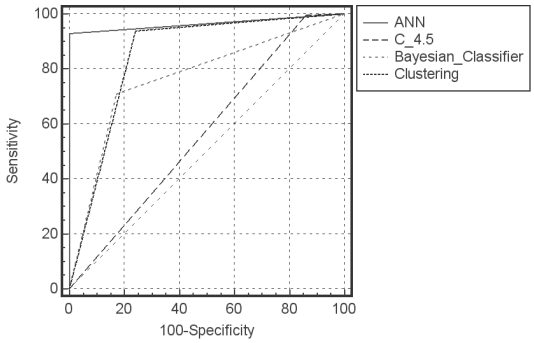

Fig. 8. Results obtained using ROC curves

TABLE IV
AUC VALUES

Classifier	AUC Value
ANN	96.5%
K-Means	84.9%
Bayesian Cl.	76.9%
C4.5	56.7%

extraction and on the Data Mining algorithms application. On the other hand, the elimination of instances / attributes with missing data may cause the information loss and replacement by a default value may introduce distorted information on the base, which do not belong to the events and circumstances that generated it.

There are some techniques that can face missing data, but most of them fail when it exists a massive data absence. In this work the considered database has 23.5% of missing value and 0% of completely filled records. Due to this reason, it was necessary the construction of classification models from the completely filled records, to be able to recover the attribute store income interval.

Experiments show how the domain expert helps to define the data structure for an effective recovering process.

Comparing the accuracy of the classifiers applied to imbalanced databases with the results of the same techniques applied to balanced databases, it can be observed the importance of the database balancing stage in the KDD preprocessing activity, especially when dealing with realworld databases. The preprocessing activity is considered to be one of the most important of the KDD process, where generally it is required much attention to guarantee a good final result. This importance can be noticed in the homogeneity of the accuracy rate for the classifiers which was trained with the balanced data set.

However, when analyzing the AUC values obtained through ROC curves construction, it was possible to measure a more realistic accuracy rate for the classifiers. This rate was capable to detect that the C4.5 decision tree was deeply biased during the training process. It can also be observed that ANN was highly efficient in the task of value recovering.

In future works, an robustness analyzis of the classifiers when trained with a minor data set will be done. It also will be constructed ROC curves with more validation data, which could result in a curve with more points.

VIII. ACKNOWLEDGMENTS

The authors gratefully acknowledge the financial support of CNPq (National Councill for Scientific and Technologic Development- Brazil).

REFERENCES

[1] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth and R. Uthurusamy, *Advances in Knowledge Discovery and Data Mining*, MIT Press, USA, 1996.

[2] L. Lei, N. Wu and P. Liu, "Applying Sensitivity Analysis to Missing Data in Classifiers", *in ICSSSM05: Proceedings of the 2005 International Conference on Services Systems and Services Management*, China, 2005.

[3] Y. Fujikawa, "Efficient Algorithms for Dealing with Missing values in Knowledge Discovery", *School of Knowledge Science - Japan Advanced Institute of Science and Technology*. Japan, 2001.

[4] D. Pyle, *Data Preparation for Data Mining*, Morgan Kaufmann, USA, 1999.

[5] P. Liu, L. Lei and N. Wu, "A quantitative study of the effect of missing data in classifiers", *in CIT'05: Proceedings of the 2005 Fifth international conference on Computer and Information Technology*.

[6] A. P. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms," *in Pattern Recognition, vol. 30, pp. 1145-1159*, 1997.

[7] P. Gottgtroy, N. Kasabov and S. Mcdonell, "An Ontology engineering approach for knowledge discovery from data in evolving domains", *Knowledge Engineering and Discovery Institute*, Auckaland University of Technology - New Zealand.

[8] R. J. Brachman and T. Anand. "The process of knowledge discovery in databases: A human centered approach", *in U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, Advances in Knowledge Discovery and Data Mining, chapter 2, pages 37-57*. AAAI/MIT Press, 1996.

[9] M. Hofmann and B. Tierney, "The involvement of human resources in large scale data mining projects", *in Proceedings of the 1st international symposium on Information and communication technologies*, Ireland, 2003, pp. 103 - 109.

[10] C.E. Shannon, "The Mathematical Theory of Communication", *in Bell System Technical Journal*, 1948.

[11] N.I. Fisher, *Statistical Analysis of Circular Data*, Cambridge Univesity Press, Australia, 1995.

[12] G. Batista, R.C. Patri, M.C. Monard. "A Study of the Behaviour of Several Methods for Balancing Machine Learning Training Data", in *SIGKDD Explorations*, June 2004, Vol. 6 Issue 1, pp.20-29.

[13] H. Guo and H.L. Viktor, "Learning from Imbalanced Data Sets with Boosting and Data Generation: The DataBoost-IM Approach", *in SIGKDD Explorations*, June 2004, Vol. 6 Issue 1, pp.30-39.

[14] N. Japkowicz, "Concept-Learning in the Presence of Between- Class and Within-Class Imbalances", *in Proceedings of the Fourteenth Conference of the Canadian Society for Computational Studies of Intelligence (AI'2001)*, Canada, 2001.

[15] G.M. Weiss, "Mining with Rarity: A Unifying Framework", *in SIGKDD Explorations*, June 2004, Vol. 6 Issue 1, pp.7-19.

[16] R.C. Prati, G. Batista and M.C. Monard, "Uma Experiência no Balanceamento Artificial de Conjuntos de Dados para Aprendizado com Classes Desbalanceadas utilizando Análise ROC", *in Proceedings of IV Workshop on Advances & Trends in AI for Problem Solving*, Chilán, 2003.

[17] S. Haykin, *Redes Neurais - Princípios e Práticas*,Bookman, Brazil, 2001.

[18] J. R. Quinlan, *C4.5: Programs for machine learning*, Morgan Kaufmann, San Mateo, CA, 1993.

[19] K. Woods and K. W. Bowyer, "Generating ROC Curves for Artificial Neural Networks", *in IEEE Transactions on Medical Imaging, volume 16, pages 329-337*, 1997.

[20] M. Kantardzic, *Data Mining - Concepts, Models, Methods and Algorithms*, IEEE Press, USA, 2003.