A Study of K-Nearest Neighbour as an Imputation Method

Gustavo E. A. P. A. Batista and Maria Carolina Monard
University of São Paulo - USP
Institute of Mathematics and Computer Science - ICMC
Department of Computer Science and Statistics - SCE
Laboratory of Computational Intelligence - LABIC
P. O. Box 668, 13560-970 - São Carlos, SP, Brazil
{gbatista, mcmonard}@icmc.usp.br

Abstract. Data quality is a major concern in Machine Learning and other correlated areas such as Knowledge Discovery from Databases (KDD). As most Machine Learning algorithms induce knowledge strictly from data, the quality of the knowledge extracted is largely determined by the quality of the underlying data. One relevant problem in data quality is the presence of missing data. Despite the frequent occurrence of missing data, many Machine Learning algorithms handle missing data in a rather naive way. Missing data treatment should be carefully thought, otherwise bias might be introduced into the knowledge induced. In this work, we analyse the use of the k-nearest neighbour as an imputation method. Imputation is a term that denotes a procedure that replaces the missing values in a data set by some plausible values. Our analysis indicates that missing data imputation based on the k-nearest neighbour algorithm can outperform the internal methods used by C4.5 and CN2 to treat missing data.

1 Introduction

Data quality is a major concern in Machine Learning and other correlated areas such as Knowledge Discovery from Databases (KDD). As most Machine Learning algorithms induce knowledge strictly from data, the quality of the knowledge extracted is largely determined by the quality of the underlying data.

One relevant problem in data quality is the presence of missing data. Missing data may have different sources such as death of patients, equipment malfunctions, refusal of respondents to answer certain questions, and so on.

Despite the frequent occurrence of missing data, many Machine Learning algorithms handle missing data in a rather naive way. Missing data treatment should be carefully thought, otherwise bias might be introduced into the knowledge induced.

Imputation is a term that denotes a procedure that replaces the missing values in a data set by some plausible values. One advantage of this approach is that the missing data treatment is independent of the learning algorithm used. This allows the user to select the most suitable method for each situation.

The objective of this work is to analyse the performance of the k-nearest neighbour as an imputation method for missing data. The performance of this method is compared to the performance of two well known Machine Learning algorithm: CN2 [4] and C4.5 [12].

This work is organized as follows: Section 2 describes the taxonomy proposed by [10] to classify the degree of randomness of the missing data in a data set; Section 3 surveys the most used methods for missing data treatment; Section 4 is dedicated to a specific class of missing data treatment methods: imputation; Section 5 presents the k-nearest neighbour as an imputation method for treating missing values; Section 6 describes how the Machine Learning algorithms C4.5 and CN2 treat missing data internally; Section 7 performs a comparative study of the k-nearest neighbour algorithm as an imputation method with the internal methods used by C4.5 and CN2 to treat missing data; finally, Section 8 presents the conclusions of this work.

2 Randomness of Missing Data

Missing data randomness can be divided into three classes, as proposed by [10]:

1. *Missing completely at random (MCAR).* This is the highest level of randomness. It occurs when the probability of an instance (case) having a missing value for an attribute does not depend on either the known values or the missing data. In this level of randomness, any missing data treatment method can applied without risk of introducing bias on the data;

2. *Missing at random (MAR).* When the probability of an instance having a missing value for an attribute may depend on the known values, but not on the value of the missing data itself;

3. *Not missing at random (NMAR).* When the probability of an instance having a missing value for an attribute could depend on the value of that attribute.

3 Methods for Treating Missing Data

There are several methods for treating missing data available in the literature. Many of these methods, such as case substitution, were developed for dealing with missing data in sample surveys, and have some drawbacks when applied to the Data Mining context. Other methods, such as replacement of missing values by the attribute mean or mode, are very naive and should be carefully used to avoid insertion of bias.

In a general way, missing data treatment methods can be divided into three categories, as proposed in [10]:

1. *Ignoring and discarding data.* There are two main ways to discard data with missing values. The first one is known as *complete case analysis*, it is available in all statistical programs and is the default method in many programs. This method consists of discarding all instances (cases) with missing data. The second method is known as *discarding instances and/or attributes*. This method consists of determining the extent of missing data on each instance and attribute, and delete the instances and/or attributes with high levels of missing data. Before deleting any attribute, it is

necessary to evaluate its relevance to the analysis. Unfortunately, relevant attributes should be kept even with a high degree of missing values. Both methods, complete case analysis and discarding instances and/or attributes, should be applied only if missing data are MCAR, because missing data that are not MCAR have non-random elements that can bias the results;

2. *Parameter estimation.* Maximum likelihood procedures are used to estimate the parameters of a model defined for the complete data. Maximum likelihood procedures that use variants of the Expectation-Maximization algorithm [5] can handle parameter estimation in the presence of missing data;

3. *Imputation.* Imputation is a class of procedures that aims to fill in the missing values with estimated ones. The objective is to employ known relationships that can be identified in the valid values of the data set to assist in estimating the missing values. This papers focus on imputation of missing data. More details about this class of methods are described next.

4 Imputation Methods

Imputation methods involve replacing missing values with estimated ones based on some information available in the data set. There are many options varying from naive methods like mean imputation to some more robust methods based on relationships among attributes.

This section surveys some widely used imputation methods, although others forms of imputation are available.

1. *Case substitution.* This method is typically used in sample surveys. One instance with missing data (for example, a person that cannot be contacted) is replaced by another nonsampled instance;

2. *Mean and mode.* One of the most frequently used methods. This method consists of replacing the missing data for a given attribute by the mean (quantitative attribute) or mode (qualitative attribute) of all known values of that attribute;

3. *Hot deck and cold deck.* In the hot deck method, a missing attribute value is filled in with a value from an estimated distribution for the missing value from the current data. Hot deck is typically implemented into two stages. In the first stage, the data are partitioned into clusters. And, in the second stage, each instance with missing data is associated with one cluster. The complete cases in a cluster are used to fill in the missing values. This can be done by calculating the mean or mode of the attribute within a cluster. Cold deck imputation is similar to hot deck but the data source must be other than the current data source;

4. *Prediction model.* Prediction models are sophisticated procedures for handling missing data. These methods consist of creating a predictive model to estimate values that will substitute the missing data. The attribute with missing data is used as class-attribute, and the remaining attributes are used as input for the predictive

model. An important argument in favour of this approach is that, frequently, attributes have relationships (correlations) among themselves. In this way, those correlations could be used to create a predictive model for classification or regression (depending on the attribute type with missing data, being, respectively, nominal or continuous). Some of these relationships among the attributes may be maintained if they were captured by the predictive model. An important drawback of this approach is that the model estimated values are usually more well-behaved than the true values would be, *i.e.*, since the missing values are predicted from a set of attributes, the predicted values are likely to be more consistent with this set of attributes than the true (not known) value would be. A second drawback is the requirement for correlation among the attributes. If there are no relationships among one or more attributes in the data set and the attribute with missing data, then the model will not be precise to estimate the missing values.

5 Imputation with k-Nearest Neighbour

In this work we propose the use of k-nearest neighbour algorithm to estimate and substitute missing data. The main benefits of this approach are:

- k-nearest neighbour can predict both discrete attributes (the most frequent value among the k nearest neighbours) and continuous attributes (the mean among the k nearest neighbours);
- There is no necessity for creating a predictive model for each attribute with missing data. Actually, the k-nearest neighbour does not create explicit models (like a decision tree or a set of rules), once the data set is used as a "lazy" model. Thus, the k-nearest neighbour can be easily adapted to work with any attribute as class, by just modifying which attributes will be considered in the distance metric. Also, this approach can easily treat examples with multiple missing values.

The main drawback of this approach is:

- Whenever the k-nearest neighbour looks for the most similar instances, the algorithm searches through all the data set. This limitation can be very critical for KDD, since this research area has, as one of its main objectives, the analysis of large databases. Several works that aim to solve this limitation can be found in the literature. One method is the creation of a reduced training set for the k-nearest neighbour composed only by prototypical examples [13]. In this work we use an access method called M-tree [3], that we have implemented in our k-nearest neighbour algorithm. M-trees can organize and search data sets based on a generic metric space. M-trees can drastically reduce the number of distance computations in similarity queries.

6 How C4.5 and CN2 Treat Missing Data

C4.5 and CN2 are two well known Machine Learning Algorithms that induce propositional concepts: decision trees and rules, respectively. These algorithms were selected because they are considered two of the best Machine Learning algorithms. C4.5 seems to have a good internal algorithm to treat missing values, since a recent comparative

study, with other simple methods to treat missing values, concluded that it was one of the best methods [6]. On the other hand, CN2 seems to use a rather simple method to treat missing data.

C4.5 uses a probabilistic approach to handle missing data. Missing values can be present in any attribute, except the class attribute, in training and test files.

Given a training set, T, C4.5 finds a suitable test, based on a single attribute, that has one or more mutually exclusive outcomes $O_1, O_2, \ldots, O_n$. T is partitioned into subsets T_1, T_2, ..., T_n, where T_i contains all the instances in T that satisfy the test with outcome O_i. The same algorithm is applied to each subset T_i until a stop criteria is applied.

C4.5 uses the *information gain ratio* measure to choose a good test to partition the instances. If there exist missing values in an attribute X, C4.5 uses the subset with all known values of X to calculate the information gain.

Once a test based on an attribute X is chosen, C4.5 uses a probabilistic approach to partition the instances with missing values in X. When an instance in T with known value is assigned to a subset T_i, this indicates that the probability of that instance belonging to subset T_i is 1 and to all other subsets is 0. When the value is not known, only a weaker probabilistic statement can be made. C4.5 associates to each instance in T_i a *weight* representing the probability of that instance belonging to T_i. If the instance has a known value, and satisfies the test with outcome O_i, then this instance is assigned to T_i with weight 1; if the instance has an unknown value, this instance is assigned to all partitions with different weights for each one. The weight for the partition T_i is the probability of that instance belongs to T_i. This probability is estimated as the sum of the weights of instances in T known to satisfy the test with outcome O_i, divided by the sum of weights of the cases in T with known values on the attribute X.

The CN2 algorithm uses a rather simple imputation method to treat missing data. Every missing value is filled in with its attribute most common known value, before calculating the entropy measure [4].

7 Experimental Analysis

The main objective of the experiments conducted in this work is to evaluate the efficiency of the k-nearest neighbour algorithm as an imputation method to treat missing data, comparing its performance with the performance obtained by the internal algorithms used by C4.5 and CN2 to learn with missing data.

In our experiments, missing values were artificially implanted, in different rates and attributes, into the data sets. The performance of all three missing data treatments are compared using cross-validation estimated error rates. In particular, we are interested in analysing the behaviour of these treatments when the amount of missing data is high since some researchers have reported to find databases where more than 50% of the data were missing, for instance [8].

The experiments were carried using three data sets from UCI [11]: Bupa, Cmc and Pima. We chose these three data sets because they have no missing values. The main reason for this choice is that we want to have total control over the missing data in the data set. For instance, we would like that the test sets do not have any missing data. If some test set has missing data, then the inducer's ability to classify missing data

properly may influence on the result. This influence is undesirable since the objective of this work is to analyse the viability of the k-nearest neighbour as imputation method for missing data and the inducer learning ability when missing values are present. Table 1 summarizes the data sets employed in this study. It shows, for each data set, the number of instances (#Instances), number and percentage of duplicate (appearing more than once) or conflicting (same attribute-value but different class attribute) instances, number of attributes (#Attributes) quantitative and qualitative, class attribute distribution and the majority class error. These information has been obtained using the MLC++ *info* utility [7].

Data set	# Instances	#Duplicate or conflicting (%)	#Attributes (quanti., quali.)	Class	Class %	Majority Error
bupa	345	4 (1.16%)	6 (6,0)	1	42.03%	42.03%
				2	57.97%	on value 2
cmc	1473	115 (7.81%)	9 (2,7)	1	42.70%	57.30%
				2	22.61%	on value 1
				3	34.69%	
pima	769	1 (0.13%)	8 (8,0)	0	65.02%	34.98%
				1	34.98%	on value 0

Table 1: Data sets Summary Descriptions

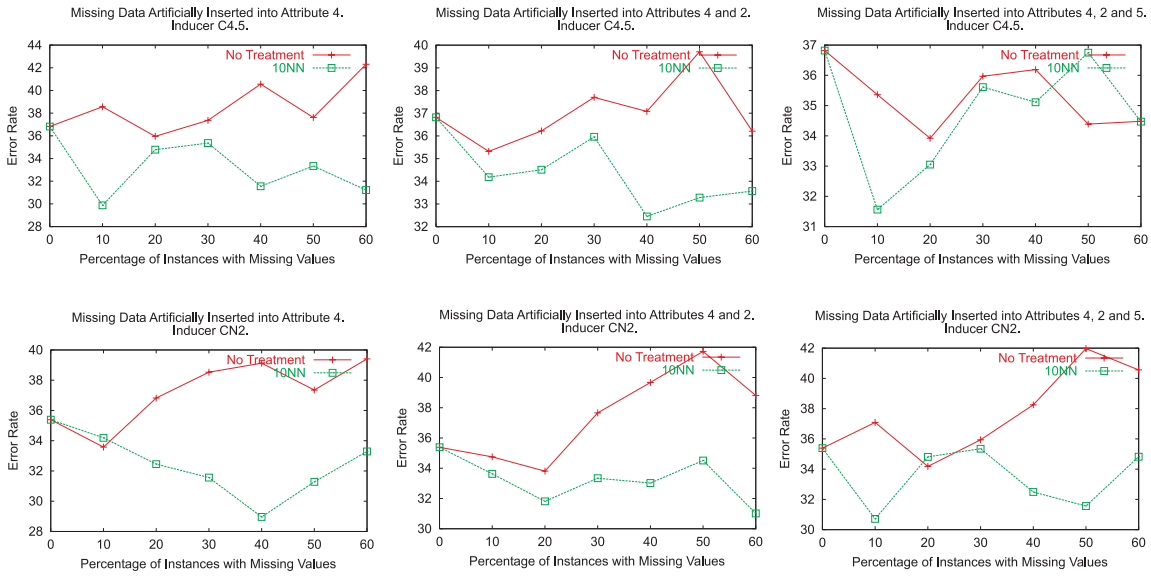

Figure 1: Comparative results for the Bupa data set.

Initially, the original data set is partitioned into 10 pairs of training and test sets through the application of 10-fold cross validation resampling method. Then, missing values are inserted into the training set. Four copies of this training set are used, two are given to C4.5 and CN2 without any missing data treatment. In the other two, the k-nearest neighbour is used to estimate and substitute missing values. After the treatment of missing data, the training sets are given to C4.5 and CN2. All classifiers, *i.e.* the two induced with untreated data and the other two induced with treated data, are used to classify the test set. At the end of 10 iterations, we can estimate the true error rate by calculating the mean of the error rates of each iteration. Finally, the performances of C4.5 and CN2 allied to the missing data treatment method can be analysed and

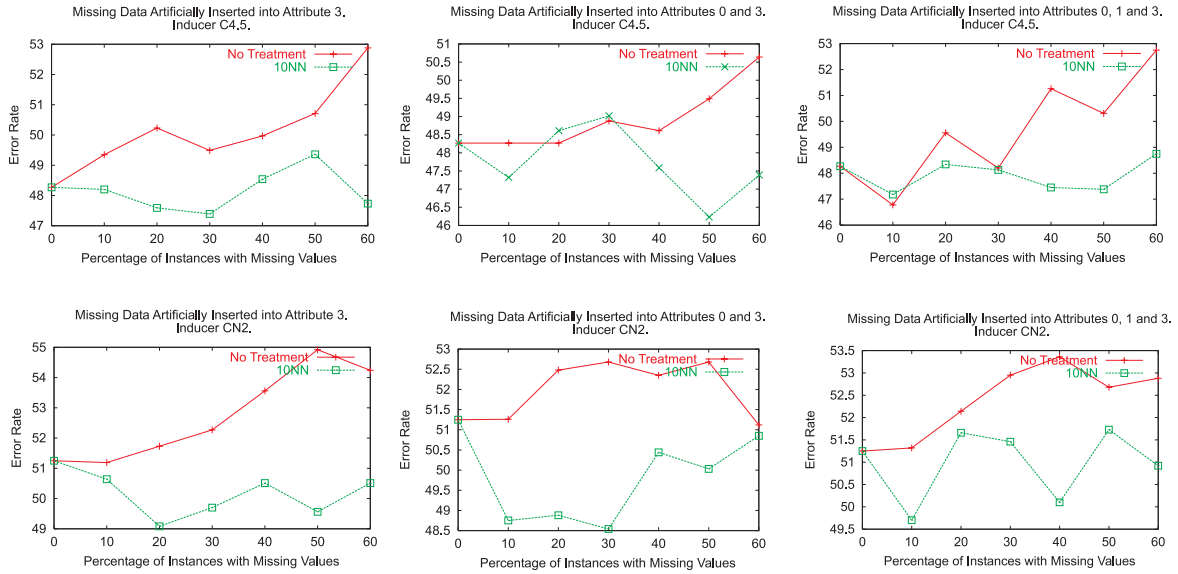

Figure 2: Comparative results for the Cmc data set.

compared to the performance of the method used internally by C4.5 and CN2 to learn when missing values are present.

In order to insert missing data into the training sets, some attributes have to be chosen, and some of their values modified to unknown. Which attributes will be chosen and how many of their values will be modified to unknown is an important decision. It is easy to see that the most representative attributes of the data set are a sensible choice for the attributes that should have their values modified to unknown. Otherwise, the analysis may be compromised by treating non-representative attributes that will not be incorporated into the classifier by the learning system. Since finding the most representative attributes of a data set is not a trivial task, we used the results of [9] to select the three most relevant attributes according to several feature subset selection method such as wrapper and filter.

There is no assurance that these attributes will be incorporated by C4.5 and CN2 into the classifiers induced in the experiments. The existence of an attribute with similar information (high correlation) with one of the selected attributes, can make the inducers choose not to use the selected attribute.

Related to the amount of missing data to be inserted into the training sets, we want to analyse the behaviour of the methods with different amounts of missing data. In this way, missing data was inserted completely at random (MCAR) in the following percentages: 10%, 20%, 30%, 40%, 50% and 60% of the total of instances.

The experiments were done with missing data inserted into one, two and all three attributes selected as the most representative. The missing values were replaced by estimated values using 1, 3, 5, 10, 20, 30, 50 and 100 nearest neighbours. Unfortunately, because of lack of space, only results with 10-nearest neighbour, identified as 10-NNI, will be showed in this work.

Considering the results shown in Figure 1, the performance of 10-NNI is superior to the performance of both C4.5 and CN2 algorithms for the Bupa data set. The only situation that C4.5 is competitive to 10-NNI is when missing values are inserted into the attributes 4, 2 and 5.

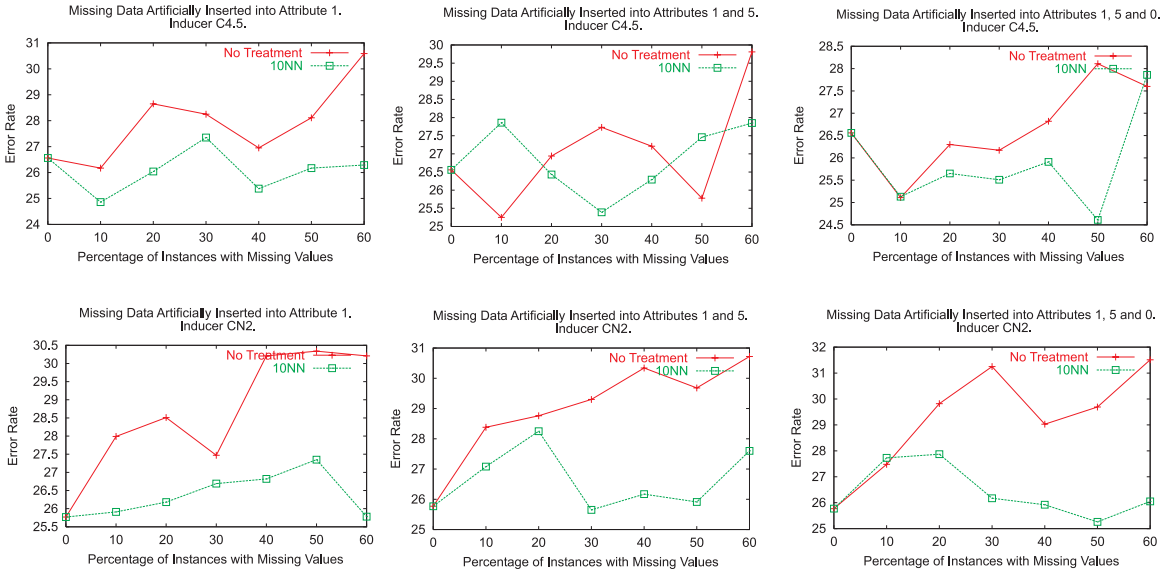

Figure 3: Comparative results for the Pima data set.

Similar results are shown in Figure 2. The performance of the imputation method 10-NNI is superior to the performance obtained, without missing data treatment, for both C4.5 and CN2 in the Cmc data set.

Finally, Figure 3 shows the comparative results for the Pima data set. In this data set, the method 10-NNI had a slightly superior performance compared with C4.5, and a superior performance compared with CN2.

Table 2 shows some numerical results of the graphs presented in Figures 1, 2 and 3. In this table are showed the error rates, standard deviations and 10-fold cross validation paired t-test results. More detailed results can be seen in [1].

It is important to say that the internal methods to treat missing data of C4.5 and CN2 had lower error rates compared to 10-NNI only in 11 of 108 measurements (8 for C4.5 and 3 for CN2). In none of these 11 measurements the internal methods had a statistically significant difference. On the other hand, 10-NNI had statistically significant difference in 35 measurements (13 of these measurements were high significant).

8 Conclusion

The main objective of this work is to present some results of a research that aims to analyse the benefits and drawbacks of missing data treatment methods [2]. In this work, we analyse the behaviour of three methods for missing data treatment: the 10-NNI method using a k-nearest neighbour algorithm for missing data imputation; and the internal algorithms used by C4.5 and CN2 to treat missing data. These methods were analysed inserting different percentages of missing data into different attributes. The results are very promising. The 10-NNI method provide very good results, even when the training sets had a large amount of missing data.

			C4.5			CN2		
		%?	No Imputation	t-test	10-NN	No Imputation	t-test	10-NN
Bupa	4	0%	36.82 ± 2.69	-	-	35.39 ± 2.47	-	-
		10%	38.56 ± 1.74	$\uparrow$ -3.01	29.87 ± 1.76	33.58 ± 1.94	0.22	34.19 ± 1.45
		20%	35.95 ± 1.24	-0.67	34.78 ± 2.43	36.82 ± 0.96	$\Uparrow$ -3.51	32.45 ± 0.95
		30%	37.36 ± 1.89	-0.72	35.36 ± 2.71	38.53 ± 2.16	$\Uparrow$ -4.02	31.56 ± 2.71
		40%	40.56 ± 2.05	$\uparrow$ -3.09	31.55 ± 1.86	39.13 ± 1.09	$\Uparrow$ -6.21	28.96 ± 2.24
		50%	37.62 ± 2.35	-1.16	33.34 ± 2.54	37.35 ± 2.74	-2.02	31.28 ± 1.91
		60%	42.31 ± 2.11	$\uparrow$ -2.62	31.22 ± 3.33	39.41 ± 1.20	$\uparrow$ -2.38	33.29 ± 2.64
	4 and 2	0%	36.82 ± 2.69	-	-	35.39 ± 2.47	-	-
		10%	35.32 ± 2.36	-0.52	34.18 ± 1.72	34.75 ± 2.01	-0.57	33.63 ± 1.77
		20%	36.22 ± 2.18	-0.57	34.51 ± 2.16	33.81 ± 3.23	-0.64	31.81 ± 2.65
		30%	37.70 ± 2.40	-0.90	35.96 ± 2.05	37.66 ± 1.48	$\uparrow$ -2.65	33.34 ± 1.88
		40%	37.08 ± 1.42	$\uparrow$ -2.51	32.45 ± 1.09	39.67 ± 1.98	-2.25	33.02 ± 2.44
		50%	39.71 ± 2.76	-1.85	33.28 ± 3.07	41.72 ± 1.38	$\uparrow$ -3.04	34.51 ± 2.40
		60%	36.21 ± 1.84	-1.09	33.57 ± 2.38	38.81 ± 1.58	$\Uparrow$ -3.90	31.01 ± 1.48
	4, 2 and 5	0%	36.82 ± 2.69	-	-	35.39 ± 2.47	-	-
		10%	35.36 ± 1.76	-1.51	31.56 ± 2.44	37.09 ± 2.55	$\uparrow$ -2.45	30.71 ± 2.47
		20%	33.92 ± 2.07	-0.36	33.05 ± 2.09	34.18 ± 2.03	0.38	34.81 ± 1.49
		30%	35.97 ± 2.90	-0.08	35.61 ± 3.00	35.94 ± 2.14	-0.24	35.35 ± 1.39
		40%	36.19 ± 2.39	-0.38	35.11 ± 2.14	38.25 ± 1.49	$\uparrow$ -3.09	32.49 ± 1.20
		50%	34.39 ± 2.84	0.97	36.75 ± 2.12	41.97 ± 1.58	$\Uparrow$ -5.34	31.56 ± 1.58
		60%	34.48 ± 1.77	-0.00	34.47 ± 3.02	40.56 ± 1.88	-2.05	34.82 ± 2.04
Cmc	3	0%	48.27 ± 0.83	-	-	51.25 ± 0.80	-	-
		10%	49.35 ± 1.14	-1.27	48.20 ± 1.16	51.19 ± 1.51	-0.48	50.64 ± 1.22
		20%	50.23 ± 1.12	$\uparrow$ -2.31	47.59 ± 0.98	51.73 ± 1.17	$\uparrow$ -2.61	49.08 ± 0.95
		30%	49.49 ± 0.95	-2.16	47.39 ± 1.48	52.27 ± 0.94	-1.96	49.70 ± 1.71
		40%	49.97 ± 0.87	-1.58	48.54 ± 1.12	53.56 ± 1.47	$\uparrow$ -2.28	50.51 ± 1.11
		50%	50.71 ± 1.11	-1.14	49.36 ± 0.91	54.92 ± 0.95	$\Uparrow$ -4.18	49.56 ± 1.74
		60%	52.88 ± 1.25	$\Uparrow$ -7.08	47.73 ± 0.95	54.24 ± 1.31	$\uparrow$ -2.63	50.51 ± 1.12
	3 and 0	0%	48.27 ± 0.83	-	-	51.25 ± 0.80	-	-
		10%	48.27 ± 0.67	-1.10	47.32 ± 1.30	51.26 ± 0.80	-2.16	48.75 ± 1.42
		20%	48.27 ± 0.99	0.30	48.61 ± 1.30	52.48 ± 1.51	$\uparrow$ -2.84	48.88 ± 1.46
		30%	48.88 ± 1.40	0.09	49.02 ± 1.36	52.68 ± 0.91	$\uparrow$ -2.77	48.54 ± 1.34
		40%	48.61 ± 1.20	-0.76	47.59 ± 1.53	52.35 ± 1.10	$\uparrow$ -2.88	50.44 ± 1.09
		50%	49.49 ± 0.84	$\uparrow$ -3.16	46.23 ± 1.06	52.68 ± 0.81	-1.60	50.03 ± 1.76
		60%	50.64 ± 1.16	-1.62	47.39 ± 1.87	51.12 ± 1.53	-0.28	50.85 ± 1.49
	3, 0 and 1	0%	48.27 ± 0.83	-	-	51.25 ± 0.80	-	-
		10%	46.78 ± 1.46	0.29	47.18 ± 1.19	51.32 ± 1.19	-1.13	49.70 ± 1.61
		20%	49.56 ± 1.34	-1.75	48.34 ± 1.29	52.14 ± 1.04	-0.39	51.66 ± 1.06
		30%	48.20 ± 1.19	-0.05	48.13 ± 1.51	52.95 ± 1.25	-1.55	51.46 ± 1.15
		40%	51.26 ± 1.33	$\uparrow$ -3.18	47.45 ± 1.46	53.36 ± 1.23	$\uparrow$ -2.91	50.10 ± 1.49
		50%	50.31 ± 1.23	$\uparrow$ -2.52	47.38 ± 1.74	52.68 ± 1.02	-0.55	51.73 ± 1.82
		60%	52.75 ± 1.16	$\Uparrow$ -3.49	48.75 ± 1.86	52.88 ± 0.76	-1.32	50.92 ± 1.35
Pima	1	0%	26.56 ± 1.16	-	-	25.77 ± 1.12	-	-
		10%	26.17 ± 1.03	-1.06	24.86 ± 0.88	27.99 ± 0.98	-2.23	25.91 ± 0.86
		20%	28.65 ± 1.15	-1.48	26.04 ± 1.68	28.51 ± 1.06	-1.80	26.18 ± 0.78
		30%	28.25 ± 1.85	-0.46	27.35 ± 1.03	27.47 ± 1.11	-0.41	26.69 ± 1.61
		40%	26.95 ± 1.67	-0.93	25.38 ± 1.15	30.21 ± 1.08	-2.00	26.82 ± 0.98
		50%	28.11 ± 1.14	-1.10	26.17 ± 1.11	30.34 ± 1.21	-1.54	27.35 ± 1.47
		60%	30.59 ± 1.13	-2.21	26.29 ± 1.90	30.21 ± 1.28	-2.21	25.78 ± 1.33
	1 and 5	0%	26.56 ± 1.16	-	-	25.77 ± 1.12	-	-
		10%	25.25 ± 1.10	2.00	27.86 ± 1.15	28.38 ± 0.87	-1.32	27.08 ± 0.98
		20%	26.94 ± 1.22	-0.39	26.43 ± 1.08	28.76 ± 1.51	-0.32	28.25 ± 1.09
		30%	27.73 ± 1.60	-1.26	25.39 ± 0.81	29.30 ± 1.23	$\uparrow$ -2.45	25.65 ± 1.13
		40%	27.21 ± 1.45	-0.51	26.29 ± 1.69	30.34 ± 1.59	$\uparrow$ -2.56	26.17 ± 1.07
		50%	25.78 ± 1.13	1.39	27.46 ± 1.16	29.68 ± 1.58	-2.02	25.91 ± 1.08
		60%	29.81 ± 1.43	-1.18	27.85 ± 1.51	30.72 ± 1.47	-1.18	27.60 ± 1.47
	1, 5 and 0	0%	26.56 ± 1.16	-	-	25.77 ± 1.12	-	-
		10%	25.11 ± 1.70	0.01	25.13 ± 0.90	27.48 ± 1.00	0.21	27.73 ± 0.68
		20%	26.30 ± 1.01	-0.66	25.65 ± 1.35	29.82 ± 0.82	-1.48	27.87 ± 1.26
		30%	26.17 ± 1.35	-0.38	25.51 ± 1.75	31.25 ± 0.89	$\Uparrow$ -3.55	26.17 ± 1.32
		40%	26.82 ± 1.28	-0.67	25.91 ± 1.44	29.03 ± 0.90	$\Uparrow$ -3.67	25.92 ± 1.32
		50%	28.11 ± 1.32	$\Uparrow$ -3.41	24.61 ± 1.16	29.69 ± 0.41	$\Uparrow$ -7.98	25.26 ± 0.68
		60%	27.60 ± 1.05	0.19	27.86 ± 1.55	31.51 ± 1.17	$\uparrow$ -3.05	26.05 ± 0.86

Table 2: Comparative results for all data sets. "$\uparrow$" means that the difference is statistically significant (95% confidence level). "$\Uparrow$" means that the difference is highly significant (99% confidence level).

In future works, the missing data treatment methods will be analyzed in other data sets. Furthermore, in this work missing values are being inserted completely at random (MCAR). In a future work, we will analyze the behaviour of these methods when missing values are not randomly distributed. In this case, there exist the possibility of invalid knowledge be created by the inducer. For an effective analysis, we will have to inspect not only the error rate, but also, the quality of the knowledge induced by the learning system.

Acknowledgements. This research is partially supported by Brazilian Research Councils CAPES and FINEP.

References

[1] G. E. A. P. A. Batista and M. C. Monard. K-Nearest Neighbour as Imputation Method: Experimental Results (in print). Technical report, ICMC-USP, 2002. ISSN-0103-2569.

[2] Gustavo E. A. P. A. Batista. Data Pre-processing for Supervised Learning (in Portuguese). Phd Qualifying Exam, ICMC-USP, March 2000.

[3] Paolo Ciaccia, Marco Patella, and Pavel Zezula. M-tree: An Efficient Access Method for Similarity Search in Metric Spaces. In *The VLDB Journal*, pages 426–435, 1997.

[4] P. Clark and T. Niblett. The CN2 Induction Algorithm. *Machine Learning*, 3(4):261–283, 1989.

[5] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm (with Discussion). *Journal of Royal Statistical Society*, B39:1–38, 1977.

[6] J. W. Grzymala-Busse and M. Hu. A Comparison of Several Approaches to Missing Attribute Values in Data Mining. In *Proceedings of the Second International Conference on Rough Sets and Current Trends in Computing RSCTC'2000*, pages 340–347, 2000.

[7] R. Kohavi, D. Sommerfield, and J. Dougherty. Data Mining using MLC++: A Machine Learning Library in C++. *Tools with Artificial Intelligence*, pages 234–245, 1996.

[8] K. Lakshminarayan, S. A. Harp, and T. Samad. Imputation of Missing Data in Industrial Databases. *Applied Intelligence*, 11:259–275, 1999.

[9] Huei Diana Lee, Maria Carolina Monard, and José Augusto Baranauskas. Empirical Comparison of Wrapper and Filter Approaches for Feature Subset Selection. Technical Report 94, ICMC - USP, São Carlos, SP, Oct 1999. ftp://ftp.icmc.sc.usp.br/pub/BIBLIOTECA/rel_tec/Rt_94.ps.zip.

[10] R. J. Little and D. B. Rubin. *Statistical Analysis with Missing Data*. John Wiley and Sons, New York, 1987.

[11] C. J. Merz and P. M. Murphy. UCI Repository of Machine Learning Datasets, 1998. http://www.ics.uci.edu/ mlearn/MLRepository.html.

[12] J. R. Quinlan. *C4.5 Programs for Machine Learning*. Morgan Kaufmann, CA, 1988.

[13] D. R. Wilson and T. R. Martinez. Reduction Techniques for Exemplar-Based Learning Algorithms. *Machine Learning*, 38(3):257–286, March 2000.