Evaluating a Nearest-Neighbor Method to Substitute Continuous Missing Values

Eduardo R. Hruschka[1], Estevam R. Hruschka Jr.[2], and Nelson F. F. Ebecken[3]

[1] Universidade Católica de Santos (Unisantos)
Rua Carvalho de Mendonça, nº 144, CEP 11.070-906, Santos, SP, Brasil
erh@unisantos.br
[2] COPPE / Universidade Federal do Rio de Janeiro
Bloco B, Sala 100, Caixa Postal 68506, CEP 21945-970, Rio de Janeiro, RJ, Brasil
estevamr@terra.com.br
[3] COPPE / Universidade Federal do Rio de Janeiro
Bloco B, Sala 100, Caixa Postal 68506, CEP 21945-970, Rio de Janeiro, RJ, Brasil
nelson@ntt.ufrj.br

Abstract. This work proposes and evaluates a Nearest-Neighbor Method to substitute missing values in datasets formed by continuous attributes. In the substitution process, each instance containing missing values is compared with complete instances, and the closest instance is used to assign the attribute missing value. We evaluate this method in simulations performed in four datasets that are usually employed as benchmarks for data mining methods - Iris Plants, Wisconsin Breast Cancer, Pima Indians Diabetes and Wine Recognition. First, we consider the substitution process as a prediction task. In this sense, we employ two metrics (Euclidean and Manhattan) to simulate substitutions both in original and normalized datasets. The obtained results were compared to those provided by a usually employed method to perform this task, i.e. substitution by the mean value. Based on these simulations, we propose a substitution procedure for the well-known K-Means Clustering Algorithm. Then, we perform clustering simulations, comparing the results obtained in the original datasets with the substituted ones. These results indicate that the proposed method is a suitable estimator for substituting missing values, i.e. it preserves the relationships between variables in the clustering process. Therefore, the proposed Nearest-Neighbor Method is an appropriate data preparation tool for the K-Means Clustering Algorithm.

1 Introduction

Knowledge discovery in databases (KDD) is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data [1]. In this context, data mining is a step in this process that centers on the automated discov-

T.D. Gedeon and L.C.C. Fung (Eds.): AI 2003, LNAI 2903, pp. 723-734, 2003.

ery of new facts and relationships in data and it consists of three basic steps: data preparation, information discovery and analysis of the mining algorithm output [2]. The data preparation step has a major importance in the whole process and it is used as a tool to adjust the databases to the information discovery step. Thus, when it is performed in a suitable way higher quality data are produced, and the KDD outcomes can be improved. In spite of its importance, the data preparation process became an effervescent research area only in the last few years.

The substitution of missing values is an important subtask in the data preparation step. The absence of values in a dataset is a common fact in real world applications and, further than, it may generate bias in the data, affecting the quality of the KDD process. One of the most used methods to deal with the missing values problem is the mean or mode imputation [3], but this method can bring bias in the data and is not adequate in all situations. This work shows a missing value substitution method using an algorithm based on the instance-based learning method [4]. More specifically, we propose and evaluate a missing values substitution method, based on the nearest-neighbor approach, for the well-known K-Means Clustering Algorithm.

The next section presents works related to the missing values problem, whereas Section 3 presents our proposed substitution method. This method is evaluated in simulations performed in four datasets that are benchmarks for data mining methods - Iris Plants, Wisconsin Breast Cancer, Pima Indians Diabetes and Wine Recognition. The prediction simulations, described in Section 4, allow us to compare the performance of two different distance metrics – Euclidean and Manhattan. Besides, we also evaluate the influence of normalized datasets in the substitution process and all these results helped us to parameterize the Nearest-Neighbor Method, which is then evaluated as a data preparation tool for the K-Means Algorithm - Section 5. Finally, Section 6 describes the conclusions and points out some future works.

2 Related Work

The missing values problem is an important issue in data mining. Thereby there are many approaches to deal with it [5]: i) Ignore objects containing missing values; ii) Fill the gaps manually; iii) Substitute the missing values by a constant; iv) Use the mean of the objects in the same class as a substitution value; and v) Get the most probable value to fill the missing values. The first approach usually wastes too much data, whereas the second one is unfeasible in data mining tasks. The third approach assumes that all missing values represent the same value, probably leading to considerable distortions. The substitution by the mean value is common and sometimes can even lead to reasonable results. However, we believe that the best approach involves trying to fill missing values with the suitable, most probable ones.

The literature that deals with the problem of missing values describes several works. For example, when working with decision trees, some practical results can be found in [6], which just ignore the objects with missing values. Another approach involves replacing the missing values by the most frequent value [7] - *majority method.* In the *probability method* [8], a decision tree is constructed to determine the missing values of each attribute, using the information contained in other attributes

and ignoring the class. The *dynamic path generation* [6] and the *Lazy decision tree approach* [9] do not generate the whole tree, but only the most promising path.

Some works about missing value classification tasks, using committee learning approach, are *Boosting* [10], *Bagging* [11], *Sasc* (Stochastic attribute selection committee) [12] and *SascMB* (Stochastic attribute selection committee with Multiple Boosting) [13], which applies decision trees to the learning task.

Considering Bayesian methods and supposing a missing at random data [16], a way to find the posterior distribution of the variable joint probabilities and the marginal probability of the variable having missing values is treating the missing values as unknown parameters, applying a Monte Carlo Markov Chain method [17]. When the missing data mechanism is not ignorable [15], an *imputation-based* analysis can be used [16]. However, these methods have some disadvantages [14]. First, they need information about the missing values mechanism. Second, the sampling variability and the non-response variability are mixed. Third, they have a high computational cost. Some approaches that try to solve these problems can be found in [18, 19].

In multivariate analysis, some works apply the Multiple Imputation (MI) [20] method to handle missing data. MI methods provide good estimations of the sample standard errors, and several analyses can be applied. However, the data must be missing at random in order to generate a general-purpose imputation.

The EM (Expectation-Maximization) algorithm [21] can be applied when the model belongs to an exponential family, but it has a slow convergence rate. For example, the MS-EM (Model Selection – Expectation Maximization) [22], which plays relatively few iterations to find the best network with incomplete data, implements a version of the EM and uses a metric to choose the best Bayesian model. Other methods applying the EM algorithm can be seen in [15, 23].

Instance-based (IB) learning methods [4] are part of another class of algorithms that can be applied to the missing values substitution process. There are some classical IB learning algorithm classes as the *nearest neighbor* (k-NN) [3], the *locally weighted linear regression* [24], and the *case based reasoning* (CBR) [25]. One of the most important characteristics of these methods is that they do not generate a model to describe the data. In other words, they do not have the training step as the other learning methods do. Thus, instead of consulting a generated model to estimate the best value to substitute the missing one (for each substitution), these algorithms search the whole dataset to find the best instance to be used. This characteristic produces a high computational cost when working with many attributes. On the other hand, as the learning process is specific to each query, it may be more accurate.

Few works deal with the problem of substituting missing values in clustering problems. This paper proposes and evaluates a Nearest Neighbor Method to substitute missing values - similar to that employed in [3,31,32] for algorithms C4.5 and CN2 – to be employed as a data preparation tool for the K-Means Algorithm.

3 Nearest-Neighbor Method

The proposed substitution method considers that missing values can be substituted by the corresponding attribute value of the most similar complete instance (object) in the

dataset. In other words, we employ a K-nearest-neighbor method [4], using K=1 and two different distance functions (e.g. Euclidean and Manhattan). More specifically, let us consider two objects *i* and *j*, both described by a set of N continuous attributes $\{x_1,x_2,...,x_N\}$. The distance between objects *i* and *j* will be here called d(i,j). Besides, let us suppose that the *k*-th attribute value ($1 \leq k \leq N$) of the object *m* is missing. Thus, the Nearest-Neighbor Method (NNM) will compute the distances d(m,i), for all $i \neq m$, according to the Euclidean or the Manhattan distance, i.e. respectively:

$$d(m,i)_E = \sqrt{(x_1^m - x_1^i)^2 + ... + (x_{k-1}^m - x_{k-1}^i)^2 + (x_{k+1}^m - x_{k+1}^i)^2 + ... + (x_N^m - x_N^i)^2} \quad . \qquad (1)$$

$$d(m,i)_M = \left|x_1^m - x_1^i\right| + ... + \left|x_{k-1}^m - x_{k-1}^i\right| + \left|x_{k+1}^m - x_{k+1}^i\right| + ... + \left|x_N^m - x_N^i\right| . \qquad (2)$$

One observes that we are not taking into account the attribute x_k, because it is missing. After computing all distances, we choose the smallest one, which refers to the most similar object in respect to *m*. This object is here called *s,* which is the nearest neighbor. In this way, one observes that d(m,s) = min d(m,i) for all $i \neq m$, and x_k^m is substituted by x_k^s. The proposed method can be easily adapted to datasets formed by discrete attributes. To do so, one can just substitute the Euclidean/Manhattan distance function by the Simple Matching Approach [26].

4 Simulation Results in a Prediction Task

We performed simulations in four datasets that are benchmarks for data mining methods: Iris Plants, Wisconsin Breast Cancer, Pima Indians Diabetes, and Wine Recognition. These datasets were chosen because they are formed only by numeric attributes and because they are available at the UCI Machine Learning Repository [27]. All these datasets describe classification problems. However, we are mainly interested in evaluating the proposed method in the context of clustering problems, i.e. as a preprocessing tool for clustering algorithms. This fact leads us to investigate the performance of the proposed method in an *unsupervised way*, i.e. applying the method in the dataset formed by examples of all classes. In this way we can simulate the substitution process in the dataset to be clustered. Thus, in all the experiments we applied the substitution process in the datasets formed just by the attribute values (without the *class value*). Besides, we also compare these simulation results with those obtained by means of a simple and usual substitution method, which consists in substituting the missing values by the mean of the attribute values.

Basically, our simulations consider that there is just one missing value at a time. Let us consider that one has a dataset formed by L objects $i=\left(x_1^i, x_2^i, ..., x_N^i\right)$. First, we simulate that x_1^1 is missing and it is consequently substituted. Second, x_2^1 is missing and it is consequently substituted. This process is repeated until x_N^1 is substituted. After that, we simulate that x_1^2 is missing and it is consequently substituted. In sum-

mary, this procedure is repeated for all x_k^i (i=1,...,L; k=1,...,N). In this way the simulations can be easily reproduced, i.e. they are not influenced by the choice of random samples. Besides, if there is a set of objects whose distances d(m,i) are equal the substituted value comes from the first object of this subset, in the sense that the algorithm starts from the first object in the dataset and goes until the last one. After the substitution process, one has two datasets (the original one and the substituted one) and it is possible to verify how similar the substituted values are compared to the original ones. In this sense, we calculate the absolute difference between the substituted value and the original one, obtaining an average error for each attribute – considering all possible substitutions. These errors are shown by means of graphics.

Obviously the method is sensitive to the distance function choice. Therefore, in this paper we compare the results obtained by two distance functions that are commonly used: the Euclidean (1) and the Manhattan (2). Another important issue is about normalization, which is merely an option that may or may not be useful in a given application [26]. In order to evaluate its influence in the substitution process, we compare the results obtained in the original values with those obtained with the normalized ones. In this sense, we convert all the attribute values into the range [0,1], using a linear interpolation.

4.1 Iris Plants

This database consists of 3 classes (Setosa, Versicolour and Virginica), each one formed by 50 examples of plants. There are 4 attributes (sepal and petal length and width). The class Setosa is linearly separable from the others, whereas the classes Versicolour and Virginica are not linearly separable from each other. Figures 1 and 2 show that, in all experiments, the Nearest-Neighbor Method (NNM) provided lower prediction errors than the substitution by the mean. Considering the normalized data, the Euclidean distance provided better results than the Manhattan distance in attributes A3 and A4, whereas in the original data the Euclidean distance provided better results in attributes A1 and A3. If one compares the results obtained by means of normalized and original data, the normalized data provided better results in attributes A1 and A2 (Euclidean), as well as in A3 and A4 (Manhattan).

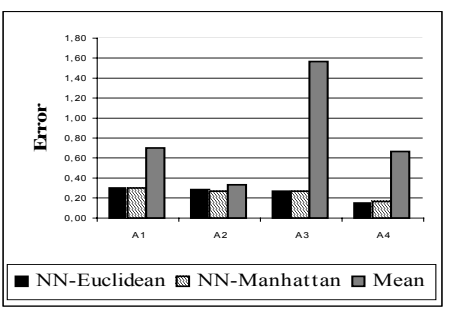

Fig. 1. Iris Plants Normalized

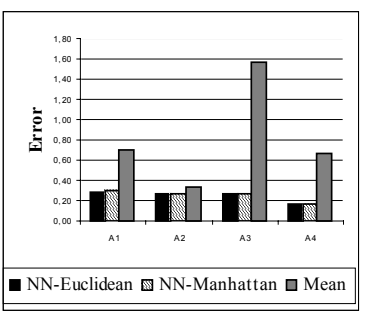

Fig. 2. Iris Plants Original

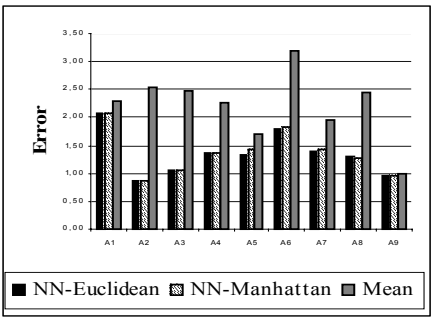

Fig. 3. Wisconsin Normalized.

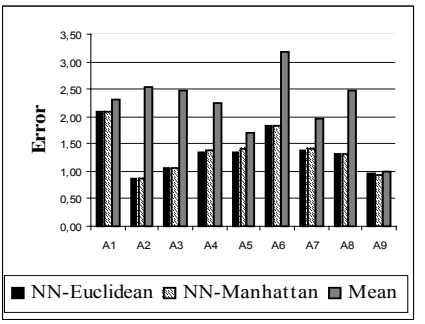

Fig. 4. Wisconsin Original.

4.2 Wisconsin Breast Cancer

In this database each object has 9 attributes and an associated class label (benign or malignant). The two classes are known to be linearly inseparable. The total number of objects is 699 (458 benign and 241 malignant), of which 16 have a single missing feature. We removed those 16 objects and used the remaining ones to simulate the substitution of missing values. Figures 3 and 4 show that, in all experiments, the NNM provided better results (lower prediction errors) than the substitution by the mean. Considering the normalized data, the Euclidean distance provided better results than the Manhattan distance in attributes A5, A6 and A7, whereas in the original data the Euclidean distance provided better results in attributes A1, A4, A5, A6 and A7**.** If one compares the results obtained by means of normalized and original data, the normalized data provided better results in attributes A3, A5, A6 and A8 (Euclidean), as well as in A1, A3, A4, A6, A7 and A8 (Manhattan). It is also necessary to say that, in theory, the results obtained in the original and in the normalized dataset should be equal, because all attribute values in this dataset belong to the set {1, 2,…, 9}. However, small differences were observed due to *rounding errors* that occur in the normalization process. In fact, the NNM can also be employed in datasets formed by discrete ordinal attributes [26], and the simulations performed in the Wisconsin Breast Cancer dataset illustrate this property.

4.3 Pima Indians

This example represents a complex classification problem. The dataset contains 768 examples – 500 meaning negative conditions for diabetes (class 1) and 268 showing positive conditions of diabetes (class 2). Each example contains 8 attributes plus the class label. Figure 5 shows that the NNM provided better results than the substitution by the mean in attributes A1, A4, A5 and A8. Figure 6 shows that the NNM provided lower average prediction errors than the substitution by the mean in attributes A4, A5 and A8 (just for Manhattan distance). The substitution errors for A7 do not appear in any figure because they are very low (less than 0.33). Considering the normalized data, the Euclidean distance provided better results than the Manhattan distance in

attributes A1, A4 and A7, whereas in the original data the Euclidean distance provided better results in attributes A3-7**.** If one compares the results obtained by normalized and original data, the normalized data provided better results in A1, A2, A4, A7 and A8 (Euclidean), as well as in A1, A3-8 (Manhattan).

4.4 Wine Recognition

In this database each object has 13 attributes and an associated class label (1, 2 or 3). The total number of objects is 178 (59 – class 1, 71 – class 2, 48 – class 3). Figure 7 shows that the NNM provided better results than the substitution by the mean in attributes A1-2, A4 (just for Manhattan distance), A5-13. Figure 8 shows that the NNM provided lower average errors than the substitution by the mean in attributes A1 (just for Manhattan distance), A6-7, A10-11 (just for Manhattan distance), A12-13. Considering the normalized data, the Euclidean distance provided better results than the Manhattan distance in attributes A1, A5-7, A10-13, whereas in the original data the Euclidean distance provided better results just in attribute A2**.** If one compares the results obtained by means of normalized and original data, the normalized data provided better results in all attributes when the Euclidean distance is applied, as well as in 12 attributes (less A13) when the Manhattan distance is applied.

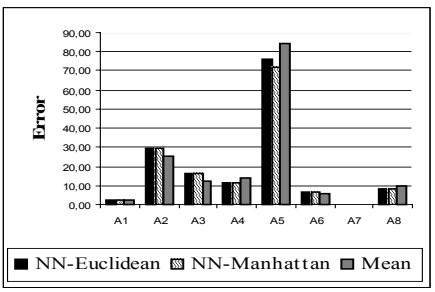

Fig. 5. Pima Indians Normalized

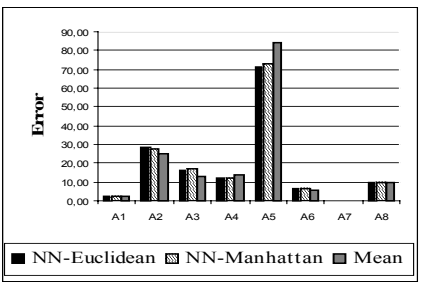

Fig. 6. Pima Indians Original

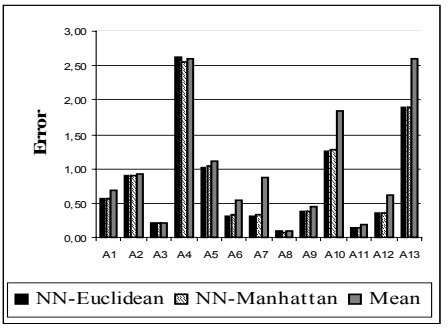

Fig. 7. Wine Recognition Normalized

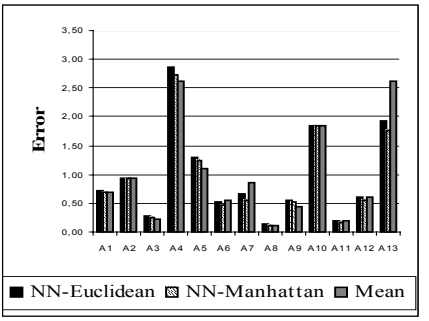

Fig. 8. Wine Recognition Original

4.5 Discussion

In general, the prediction simulation results showed that the NNM provided better results than the use of the mean. In fact, the NNM provided better results in all experiments involving the datasets Iris Plants and Wisconsin Breast Cancer. In the Wine Recognition dataset the NNM provided lower average errors in 65.38% of the experiments, whereas in the Pima Indians Diabetes the NNM provided lower average errors in just 34.38% of the experiments. Considering all the performed experiments (for each attribute, each distance metric and each dataset - normalized and original), the NNM provided better results than the substitution by the mean in 71.32% of the cases.

Another aspect that was investigated deals with the use of two different distance metrics: Euclidean and Manhattan. In order to compare the obtained results, let us consider that we performed four experiments for each dataset (two for normalized and two for original). Thus, we can compute the number of times, for each attribute, in which one metric surpass the other one. Doing so, we observed that in 57.35% of the experiments the Manhattan Distance provided lower errors than the Euclidean one. Besides, the Manhattan distance is computationally more efficient and should be preferred. Finally, we also compared the results obtained in normalized datasets with those obtained in the original datasets and we observed that normalized datasets provided better results in 75% of the experiments.

In summary, we verified that in most simulations: (i) the NNM provided better results than the substitution by the mean; (ii) the Manhattan distance is a better metric that the Euclidean one; (iii) it is better to employ normalized datasets. Although the prediction evaluation is relevant and valid, it is not the only important issue to be analyzed. In this sense, one of the most important aspects is that the substitution method must generate values that least distort the original characteristics of the original sample [28]. This being the case, we are also interested in evaluating if the substitution process preserves the between-variable relationships, which, in our study, are defined by the clustering process. This aspect is approached in the next section.

5 Simulation Results in a Clustering Process

The substituted values should preserve the between-variable relationships. In a clustering process, it means that the *natural* clusters should be preserved, i.e. the imputed values should not change the group that the object *really* belongs to. In order to evaluate this aspect, one has to suppose that the *natural* clusters are *a priori* known. If real-world datasets are considered, it is a hard task to find the *natural, correct* clusters. As previously mentioned, we have employed classification datasets in our simulations, considering that the classes form the *natural* clusters. Thus, one can verify to what extent the K-Means is capable of finding the correct clusters, which are defined by each class. In this sense, we propose to compare the A*verage Classification Rates (ACRs)* obtained by the K-Means Algorithm in the original dataset with those obtained in the substituted datasets, which are formed by the procedure described in Section 4. When the NNM is concerned, we have employed the Manhattan distance and normalized datasets, which, in our experiments, provided better results when one evaluates the substitution method as a prediction task.

Our clustering simulations were performed by means of the WEKA System [29], using the K-Means Algorithm with the default parameters and considering that the number of clusters is equal to the number of classes, defined by each employed dataset. Table 1 shows the simulation results both in the original and in the substituted datasets. In most cases, the ACRs obtained in the original datasets were very similar to the ones obtained in the substituted datasets, what indicates that the NNM provides unbiased estimates of the missing values. Bad results were just observed in the Pima Indians Dataset, which represents a very difficult classification problem [30]. However, our main goal was not to evaluate the performance of the K-Means Algorithm, which is well known. Instead, our objective was to evaluate how suitable is the NNM – as a data preparation tool - to the K-Means Algorithm. In this sense, one important aspect to be observed is the ACR difference between the original and the substituted dataset - 14.46%, which can be considered an estimate of the *inserted bias* in this dataset.

6 Conclusions and Future Work

This paper described and evaluated a Nearest-neighbor Method (NNM) to substitute missing values in datasets formed by continuous attributes. The proposed method compares each instance containing missing values with complete instances, using a distance metric, and the closest complete instance is used to assign the missing attribute value.

We evaluated the proposed method by means of simulations performed in four datasets that are benchmarks for data mining methods. First, we considered the substitution process as a prediction task. In that manner, we employed two metrics (Euclidean and Manhattan) to simulate substitutions both in original and normalized datasets. These substitutions were compared with a usually employed method, i.e. using the mean value. Besides, we evaluated the efficacy of the proposed method both in origi-

nal and normalized datasets. In summary, these simulations showed that in most cases: (i) the NNM provided better results than the substitution by the mean; (ii) the Manhattan distance is a better metric that the Euclidean one; (iii) it is really better to employ normalized datasets. Thus, our simulations suggest that a NNM based on the Manhattan distance and on normalized datasets provide better results. In this sense, the Euclidean metric is more influenced by outliers than the Manhattan one, and some attributes can have undue weights if the dataset is not normalized.

Table 1. Average Classification Rates (ACR): K-Means Algorithm

Dataset	ACR (%) Original	ACR (%) Substituted	(Original – Substituted)
Iris Plants	88.00	89.33	-1.33 %
Wisconsin Breast Cancer	96.19	95.90	+0.29 %
Pima Indians Diabetes	66.80	52.34	+14.46 %
Wine Recognition	94.38	95.51	-1.13 %

Although the prediction results are relevant, they are not the only important issue to be analyzed. In fact, one of the most important aspects is that the substitution method must generate values that least distort the original characteristics of the original sample [28], i.e. the substituted values should preserve the between-variable relationships. In our work, we evaluated this aspect in the context of the K-Means Algorithm, performing clustering simulations and comparing the results obtained in the original datasets with the substituted ones - using the Manhattan distance and normalized datasets. These results indicated that the proposed method is a suitable estimator for missing values, i.e. it preserves the relationships between variables in the clustering process. Therefore, the NNM is a suitable data preparation tool for the K-Means Algorithm.

Considering our future work, there are many aspects that can be further investigated. One important issue is to evaluate the best number of neighbors, i.e. the best K value in the K-nearest-neighbor method. In this sense, we are also going to evaluate the substitution process in the context of other learning algorithms. Besides, we are going to evaluate the efficacy of the proposed method in datasets with more missing values, comparing the NNM results with those obtained with other substitution methods. Another important aspect is to investigate the results of the substitution process when the method is applied in the examples of each class separately, because this methodology can be useful in classification tasks. Besides, we are going to evaluate the NNM in discrete datasets, applying the Simple Matching Approach [26] as the distance metric.

References

[1] Fayyad, U. M., Shapiro, G. P., Smyth, P. “From Data Mining to Knowledge Discovery : An Overview”. In: Advances in Knowledge Discovery and Data Mining, Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P., Uthurusamy, R., Editors, MIT Press, pp. 1-37, 1996.

[2] Bigus, J. P., Data Mining with Neural Networks , First edition, USA, McGraw-Hill, 1996.
[3] Batista, G. E. A. P. & Monard, M. C., An Analysis of Four Missing Data Treatment Methods for Supervised Learning, Proceedings of the First International Workshop on Data Cleaning and Preprocessing, IEEE International Conference on Data Mining, Maebashi, Japan, 2002.
[4] Mitchell, T. M. Machine Learning. McGraw-Hill, 1997.
[5] Han, J. & Kamber, M. - Data Mining: Concepts and Techniques. Morgan Kaufmann, 2001.
[6] Quinlan, J., Induction of Decision Trees. Machine Learning, 1,81-106, 1986.
[7] Kononenko, I., Bratko, I. & Roskar, E., Experiments in Automatic Learning of Medical Diagnostic Rules. Technical Report, Jozef Stefan Institute, Ljubjana, Yogoslavia, 1984.
[8] Quinlan, J. R. Unknown Attribute Values in Induction. Proceedings of 6th International Workshop on Machine Learning, 164-168, Ithaca, NY, 1989.
[9] Friedman, H. F., Kohavi, R. & Yun, Y., Lazy Decision Trees. In Proceedings of the 13th National Conference on Artificial Intelligence, pp. 717-724, AAAI Press/MIT Press, 1996.
[10] Schapire, R. E., Freund, Y., Barlett, P. & Lee, W. S., Boosting the Margin: A New Explanation for the Effectiveness of Voting Methods. Proceedings of the 14th International Conference on Machine Learning, Morgan Kaufmann, pp. 352-330, 1997.
[11] Breiman, L. Bagging Predictors. Machine Learning, 24, 123-140, 1996.
[12] Zheng, Z. & Webb, G. I., Stochastic Attribute Selection Committees. In Proceedings of the 10th Australian Joint Conference of Artificial Intelligence. Berlin: Springer-Verlag, 1998.
[13] Zheng, Z. & Webb, G. I., Stochastic Attribute Selection Committees with Multiple Boosting: Learning more Accurate and more Stable Classifier Committees. In Proceedings of the 3rd Pacific-Asia Conference on Knowledge Discovery and Data Mining, Berlin: Springer-Verlag, 1999.
[14] Gilks W. R., Richardson, S. & Spiegehalter, D. J., Markov Chain Monte Carlo in Practice. Chapman and Hall, London, 1996.
[15] Gelman, A., Carlin, J. B., Stern, H. S. & Rubin, D. B., Bayesian Data Analysis. Chapman and Hall, London, 1995.
[16] Sebastiani, P. & Ramoni, M., Bayesian Inference with Missing Data Using Bound and Collapse. Technical Report KMI-TR-58, KMI, Open University, 1997.
[17] Little, R. & Rubin, D. B., Statistical Analysis with Missing Data. Wiley, New York, 1987.
[18] Hruschka Júnior, E. R., Ebecken, N. F. F. Missing Values prediction with K2. Intelligent Data Analysis.(IDA).IOS Press, Netherlands: , v.6, n.6, 2002.
[19] Hruschka Júnior, E. R., Hruschka, E. R., Ebecken, N. F. F. A Data Preparation Bayesian Approach for a Clustering Genetic Algorithm, In: Frontiers in Artificial Intelligence and Applications, A. Abraham et al. (Eds), Soft Computing Systems: Design, Management and Applications, pp. 453-461, IOS Press, 2002.

[20] Rubin, D. B., Multiple Imputation for non Responses in Surveys. New York, John Wiley & Sons, 1987.
[21] Dempster, A. P., Laird, N. M. & Rubin, D. B., Maximum Likelihood from Incomplete Data via the EM algorithm. Journal of the Royal Statistical Society B, 39, 1-39, 1977.
[22] Friedman, N., Learning Belief Networks in the presence of Missing Values and Hidden Variables. Proceedings of the 14th International Conference on Machine Learning, 1997.
[23] Lauritzen, S. L., The EM Algorithm for Graphical Association Models with Missing Data. Computational Statistics and Data Analysis, 19, 191-201, 1995.
[24] Atkeson, C. G., Moore, A. W., Schaal, S. A., Locally Weighted Learning for Control. AI Review, 1997.
[25] Aamodt, A. & Plazas, E., Case-Based Reasoning: Methodological Variations, and System Approaches. AI Communications, 7(1), 39-52, 1994.
[26] Kaufman, L., Rousseeuw, P. J., Finding Groups in Data – An Introduction to Cluster Analysis, Wiley Series in Probability and Mathematical Statistics, 1990.
[27] Merz, C.J., Murphy, P.M., UCI Repository of Machine Learning Databases, http://www.ics.uci.edu, Irvine, CA, University of California, Department of Information and Computer Science.
[28] Pyle, D., Data Preparation for Data Mining. Academic Press, 1999.
[29] Witten, I. H., Frank, E., Data Mining – Practical Machine Learning Tools and Techniques with Java Implementations, Morgan Kaufmann Publishers, USA, 2000.
[30] Nakhaeizadeh, G., Kressel, U., Keh, S. et al., Dataset Descriptions and Results, In: Machine Learning Neural and Statistical Classification, D. Michie, D.J. Spiegelhalter and C.C. Taylor editors, pp. 131-174, Ellis Horwood Series in Artificial Intelligence, Bookcraft, Midsomer Norton, 1994.
[31] Batista, G.E.A.P.A. and Monard, M.C., A Study of K-Nearest Neighbor as an Imputation Method. In Second International Conference on Hybrid Intelligent Systems, Santiago, Chile, Soft Computing Systems: Design, Management and Applications, pp. 251-260, IOS Press, 2002.
[32] Batista, G.E.A.P.A. and Monard, M.C., A Study of K-Nearest Neighbor as a Model-Based Method to Treat Missing Data, Proceedings of Argentine Symposium on Artificial Intelligence, v. 30, pp. 1-9, Buenos Aires, 2001.