MISSING DATA IMPUTATION AND THE INDUCTIVE MODELING

Miroslav Čepek[1], Pavel Kordík[1], Miroslav Šnorek[1]

[1]Czech Technical University in Prague, Faculty of Electrical Engineering
160 00 Prague, Technická 2, Czech republic

cepekm1@fel.cvut.cz(Miroslav Čepek)

Abstract

Missing data is a big problem in simulation for data mining and data analysis. Real world applications often contains missing data. Many data-mining methods is unable to create models from data which contains missing values. Traditional approach is to delete vectors with missing data. Unfortunately, this approach may lead to decreased accuracy of the models and in the worst case all data in dataset may be deleted. For this reason many different imputation techniques were developed and some are widely used. In this paper, we present a comparison of several well-known techniques for missing data imputation. Presented techniques includes imputation of mean value, zero, value from nearest input vector and few others. In this paper we show which techniques are the best in estimation of missing values. To test imputation methods we used several different datasets. We compare the imputation methods in two ways. The first is to compare imputed data with original data. The measure of similarity is RMS. The second test was to compare the accuracy of inductive models generated from datasets with missing values replaced by different imputation techniques. Results shows that no method can be chosen as the best becouse the performance of each method depends on characteristics of the data.

Keywords: Missing data, Missing data imputation, GAME Neural network, Inductive modeling method

Presenting Author's Biography

Miroslav Čepek is a PhD student at the Department of Computer Science and Engineering of the Czech Technical University in Prague. He graduated in 2006. He is interested in biological signal processing and data mining.

ISBN 978-3-901608-32-2

1 Introduction

The problem of missing (or incomplete) data is relatively common in many fields of research, and it may have different causes such as equipment malfunctions, unavailability of equipment, refusal of respondents to answer certain questions, etc...

The problem of missing data interests many researchers. Here we present few selected work which also concerns about missing values.

Huisman [1] uses simple imputation methods (like imputing mean value or hot-deck). He performed a simulation study based on responses to items forming a scale to measure a latent trait of the respondents.

Schafer and Graham [2] gives outstanding overview of advanced imputation methods, namely Multiple Imputation method and Maximum Likelihood. Also they summarize historical development and briefly describes missing data properties.

Holmes and Bilker [3] examined effect of missing data on Learning Classification System (LCS). They tested two LCS methods (EpiCS and See5) and found out that both are sensitive to missing data. Their classification accuracy decreased about 5% when rate of missing data changes from 0% to 25%.

In [4] Schafer presents one successful imputation method called Multiple Imputations and demonstrates its capabilities on data gathered for school anti-smoking programme. The Multiple Imputation method was introduced in [5] by Little and Rubin.

As may be seen there are many different imputation methods but we did not found any comparative study which compares influence of the imputation method to the modelling methods. In this work we are interested only in one particular modeling method called **Group of Adaptive Model Evolution** (GAME) which is developed in our department. The GAME creates models from complete training data. Here we test the ability of the GAME method to create successful model even when some values from the learning set are missing and are treated with several different imputation methods. For our experiment we use several different datasets. From each dataset we remove some values and replace missing data with several imputation techniques. The performance of imputation techniques will be compared in two different ways. At first the Root Mean Square (RMS) over all inputs will be computed and used as measure of quality. The RMS will be computed as root square of difference between original dataset and result of each imputation technique.

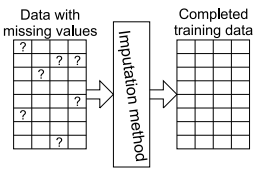

Fig. 1 Missing data imputation

The second criterion is the accuracy of the GAME models. For classification problems the classification accuracy of created models for original and imputed data is compared. For regression we compare RMS of model outputs for original and imputed data.

The general goal of our work is to explore behavior of GAME neural network with missing data and to find if suitable imputing method for specified type of data can be found. The results will be used for extending the automated FAKE GAME data mining tool.

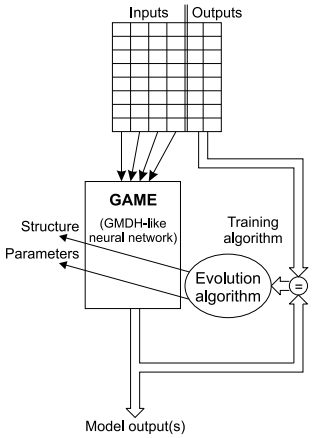

Fig. 2 Training of GAME artificial neural network

The structure of our article will be as follows: in section 2 we will describe missing values, their types and imputation methods, later in section 4 we will describe our inductive modeling method. In section 5 we will briefly describe dataset used for testing and in 6 we will describe the setup of the the experiment. Finally in section 7 we will present selected results.

2 Missing data

The key question for analysis of missing data is, what mechanism leads to the missingness of some data. This question is very hard to answer automatically. Often it is better if a human expert provides or estimates this information. In literature [2, 6, 5] different type of missing data are recognized.

- **MCAR** (Missing Completely At Random) – this means that probability of the missing value does not depend on any other value in the database. In other words: $P(\mathcal{M}|\mathbf{y}_O, \mathbf{y}_M) = P(\mathcal{M})$, where $\mathcal{M}$ is missing value, $\mathbf{y}_O$ are observed part of the data and $\mathbf{y}_M$ is missing part.
- **MAR** (Missing At Random) – in this case, values are not missing with the same probability and are depend on other values in the database. In other words: $P(\mathcal{M}|\mathbf{y}_O, \mathbf{y}_M) = P(\mathcal{M}|\mathbf{y}_O)$.
- **MNAR** (Missing Not At Random) – missing values are not random and completely depends on other values. This case is also called *non-ignorable* .

In this study we will focus on the first type - MCAR. The other two types will be subject of future research.

2.1 Missing data imputations method

Missing data can be treated with many methods. In our work we will use imputation methods. This means that correct value of missing item is predicted. Sande [7] discussed the problems an imputer is faced with, and concluded that a procedure is needed that:

1. will impute plausibly and consistently.
2. will reduce the bias and preserve the relationship between the items as far as possible.
3. will work for (almost) any pattern of missing items can be set up ahead of time.
4. can be evaluated in terms of impact on the bias and precision of the estimates.

Also different categories of imputation methods can be distinguished. First categorization is deterministic vs. stochastic – values imputed by deterministic methods are determined by the data. On the other hand stochastic methods impute random values and each run of such method will end with different imputed values.

The second categorization is explicit models versus implicit models. Little and Schenker [8] define explicit models as models which are usually discussed in mathematical statistics, for instance, normal linear regression models [6]. Implicit models are models which underlie procedures for fixing up data structures in practice and often have a nonparametric flavor [1].

Now we will present methods we used in our work.

2.1.1 Case Deletion (in text will be referred as DELETE)

This is probably the oldest and probably the most popular method among missing data threating methods. As name says we just drop the vector which contains missing data.

This approach have one big disadvantage - if missing values occurs too often, it is possible that all vectors will contain missing value and none vector will remain.

This approach is also possible only when values are missing completely at random. In other cases use of this method will change the bias (distribution and other properties) of the dataset.

2.1.2 Replace by zeros (in text will be referred as ZERO)

All missing values were replaced by zero.

2.1.3 Replace by mean value (MEAN)

For each input we calculated average value for all non missing data and all missing values in input is replaced by this average.

From statistical point of view this method have better properties than previous one.

2.1.4 Replace by value from random vector (RANDOM)

As title says for each missing value we randomly select non missing value from corresponding input.

2.1.5 Replace by value from nearest neighbor (NEARESTN)

For each input vector we determined the nearest vector and impute its value(s) at the place of the missing values.

The nearest vector is determined as the nearest in euclidean distance over all non missing dimensions.

$$\rho(x_1, x_2) = \sqrt{\sum_{i \in \mathcal{I}} (x_1 - x_2)^2}$$

where $\mathcal{I}$ is $\{i \in \mathcal{N} \| i \leq dim(x_1) = dim(x_2), x_1(i) \wedge x_2(i)$ not missing $\}$

Now the problem is that vectors with more non-missing values are disadvantaged. Therefore we decided to divide calculate ρ with number of dimensions where $x_1(i)$ and $x_2(i)$ are not missing.

2.1.6 Replace by average from values of five nearest neighbors (AVGNEAR)

In this method we extended the previous method and we calculated average value for five nearest vectors.

In future we will use more advanced methods like Multiple Imputation and EM.

3 Preliminary experiment

Prior to the work presented in this article we performed short introductory experiment. In this experiment we tested single dataset (Stock market prediction dataset) several slightly different methods – Case deletion, Replace missing values with zero and mean value, Nearest neighbor in eucleidian and dot-product distance and text match similarity. Text matching similarity means that if string representation of vector with missing values matches another complete vector, missing values are replaced from matching vector.

From complete dataset we created datasets with missing values. Portion of data missing were 5%, 10%, 50% and 80%. While removing data we maintain MCAR condition (see section 2 or [5] for explanation). Each of these datasets were completed with all above mentioned methods. Then we trained GAME ensembles on these corrected data sets. Finally, the error of ensembles on the original Stock market prediction dataset is shown in the Figure 3.

The results showed that replacement with zero is not suitable for the Stock market prediction data. Much better is to replace by the mean value. The leave out strategy is superior to other methods up to 20% of missing values. The imputing based on the Euclid distance has very promising results, specifically for high percentages of missing values in the data set.

ISBN 978-3-901608-32-2

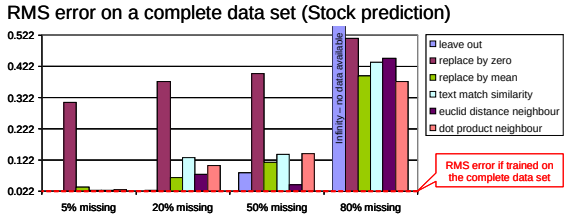

Fig. 3 The performance of imputing methods on the Stock market prediction data set with different volume of missing values.

4 Group of Adaptive Methods Evolution

In many applications it is important to find optimal model of unknown system (for example in classification, prediction, approximation, etc.). Such model can be found using two different approaches – deductive and inductive. The GAME artificial neural network (ANN) is based on inductive approach. This means that parameters and also structure of the ANN are parts of a learning process (the parameters are selected and the NN is constructed from some minimal blocks during the learning process).

The GAME ANN extends the concept of GMDH network [9, 10]. The GMDH allows only one type of minimal block (neurons with one transfer function). On the other hand in GAME ANN there are neurons with many different transfer functions (linear, sigmoid, polynomial, etc...). The GAME has a feed-forward structure [11] as illustrated in figure 4.

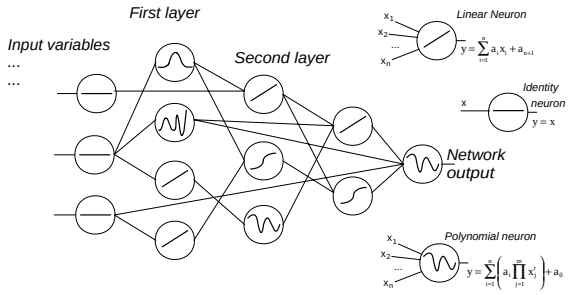

Fig. 4 Example structure of GAME artificial neural network with different types of neurons.

The structure of the network is determined by genetic algorithm which searches for optimal setup of the input connections for each unit and also optimal setup of inner parameters of each unit. More informations on GAME neural network may found in [11].

5 Data description

Here we briefly describe datasets we will use. In our experiments we use several datasets presenting wide range of problems. The reason is that we want to explore behavior of imputing methods and the GAME ANN for different types of data with missing values.

- ANTRO-CLASS – Classification problem. Data represent a set of observations the skeletal indicators studied for the proposal of the methods of age at death assessment from the human skeleton (see [12]). It is a results of the visual scoring of the morphological changes of the features in two pelvic joint surfaces defined and described by a text accompanied with photos. The material consists of 955 subjects from the 9 human skeletal series of subjects known age and sex. The age in the death of the individuals varies between 19 and 100 years.
- BANCROT – Classification problem. Task is to classify companies if they will bankrupt or not. Inputs are their financial results in previous time period.
- BUILDINGRAW – Prediction problem. The "Building data set" is frequently used for benchmarking modeling methods [13]. It predicts hot, cold water and energy consumption for the specific outside weather conditions – temperature, wind strength, etc... .
- MANDARIN – Regression problem. The Mandarin tree data set (provided by the Hort Research, New Zealand) describes water consumption of a mandarin tree.
- SPIRALS – Artificial benchmark data. Two spirals interwinded together.
- SPIRALS10 – Artificial benchmark data. Two spirals interwinded together.
- Following datasets from UCI machine learning repository [14, 15].
  - ADVERT – Classification problem.
  - BOSTONHOUSE – Regression problem. The Boston housing data set was taken from the StatLib data library. It concerns housing values in suburbs of Boston and its dependency on the house neighborhood.
  - CARS – Regression problem. Estimate value of the car using given attributes.
  - ECOLI – Classification problem. Predicts the localization of the proteins. Proteins are described in several different ways. The output is the area where protein should be located.
  - OCR – Classification problem. The task is to recognize hand-written digits.

6 Experiment Setup

In this section we describe exact setup of our experiments.

All datasets above are complete, this means that they have no missing values. Prior to any other action we divided each dataset into two parts – training and testing. The testing part we will leave as it is. We will use it to

compare the performance of created models in further sections.

From training part we artificially remove some data to simulate missing data problems. While removing values we maintained MCAR property. We removed data from training dataset several times to explore behavior of imputation methods and the GAME modeling method. Each time different portion of values is removed. In this experiment we decided to remove 1%, 5%. 10%, 25% and 50% of values.

After removing values each created dataset is passed to imputation methods described above.

First we measure the performance of imputation methods without any model. We use RMS value computed between original non-missing data and dataset with imputed values.

Second we measure the performance of imputation methods with the GAME modeling method. For each imputed dataset we create 20 different GAME models and each model is evaluated with separate testing set. This high number of generated models for each dataset is because of random nature of GAME method. Each created model have slightly different performance and we need to use statistical approach to formulate useful results.

7 Results

7.1 Data Distortion Results

The first method for comparison of results is not using any model. Imputing methods are compared according their RMS difference to original non imputed values. We use the RMS value as measure of distortion, the higher RMS value, the higher distortion of the imputed data.

The RMS can be computed as

$$RMS = \frac{1}{m}\sqrt{\sum (o_i^{(k)} - y_i^{(k)})^2}$$

, where $o_i^{(k)}$ is i-th value from k-th input from original dataset and $y_i^{(k)}$ is imputed value.

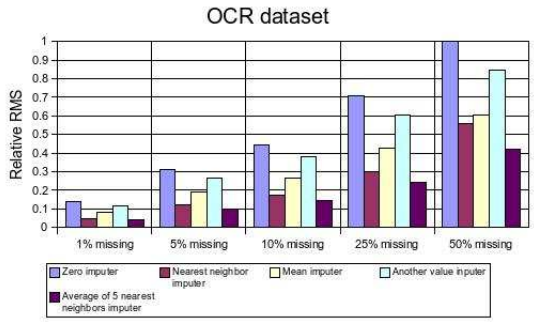

Fig. 5 RMS results for OCR dataset.

Result of one dataset is shown on Figure 5. This figure represents typical situation.

The first conclusion is that if there is higher number of missing values, there is also higher value of RMS. This is quite natural because no imputation method can produce exact values and more errors means higher RMS.

The second, not so trivial, conclusion is about accuracy of imputation methods. The worst in model-less comparison is ZERO method. This is not big surprise. Imputing still the same value, which is not the mean value, always produce high RMS error. In ideal case ZERO method may by only as good as MEAN method. The ideal case means that mean values of all datasets are equal to zero.

The RANDOM method also achieves very poor results. Reason is that it replace missing value with random non-missing value which may be very far from original value and thus it produce high RMS error. Because of random nature of this method, exact RMS value differs in each run, but general results are always the same.

The MEAN and NEARESTN methods seems to be on the same level. Their exact performance depends on properties of the dataset. When there is a low number of missing data, the NEARESTN is always better. The reason is that if two vectors are near in non-missing dimensions, their values are also near in dimension where some data are missing. But this principle works only if distance of vectors can be measured properly. That is reason why NEARESTN sometimes fails with datasets with more missing values. Some datasets have vectors which are equal or very close in few dimensions (inputs) and in other dimensions are completely different.

The relative success of MEAN method is quite clear – the mean value is the nearest constant to all values and therefore produce low RMS error.

The best method is the AVGNEAR method. It uses the same principle as the NEARESTN method do. But it uses five nearest neighbors. This eliminates disadvantages present in NEARESTN. Namely the match of irrelevant vectors. Better, among irrelevant vectors also truly near vectors are selected and the average of several values will push the result to correct value.

Results from datasets ANTRO-CLASS, ADVERT, BANCROT, BUILDINGRAW, CARS, ECOLI, MANDARIN, SPIRALS and SPIRALS10 looks similar to Figure 5 and supports results formulated here.

7.2 GAME Modeling Method Results

The second experiment is designed to find optimal imputation method for the GAME modeling method. The results presented here were obtained from two different datasets. For both datasets we compare their RMS values. The RMS error are computed as difference between originally observed outputs and model responses.

For each dataset we estimate mean value and deviance of models RMS and present them in form of boxplot graph on Figures 6 and 7. Each row presents one dataset (one imputation method). On the left side are names of the imputation methods used to create given dataset. Method name is followed by ratio of missing values.

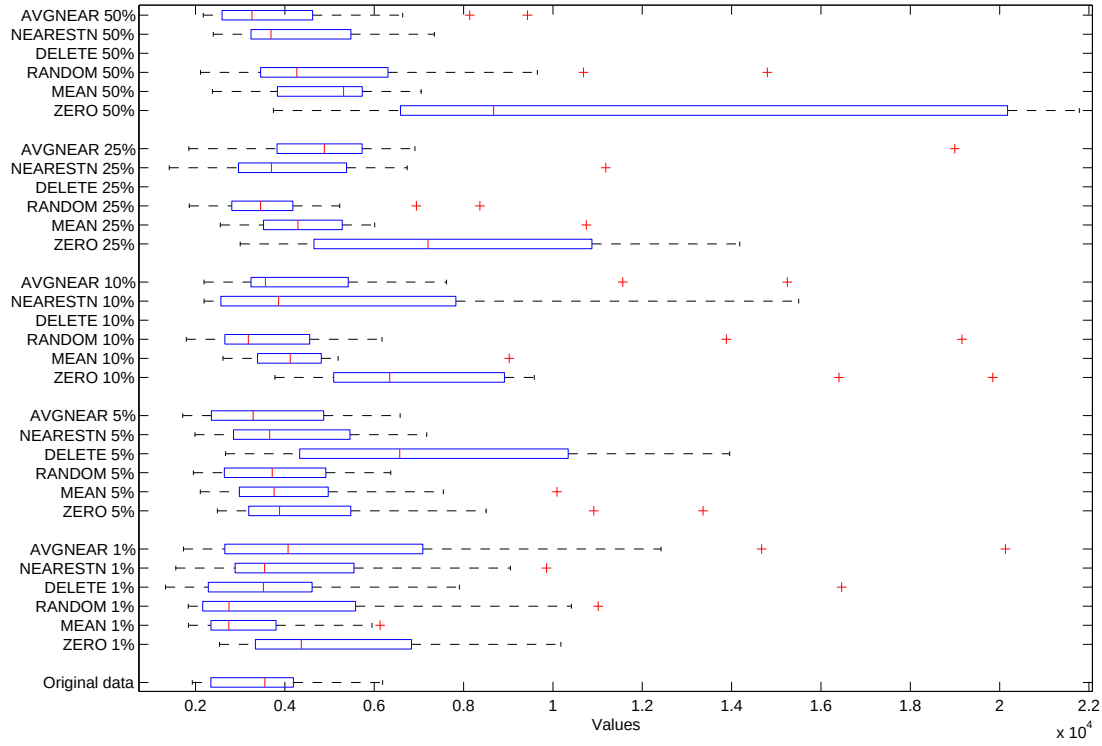

Fig. 6 GAME ANN classification RMS results as boxplot for CARS dataset. Imputation methods used to create given dataset is on the label on the left side, followed by ratio of missing values. Better values are on the left side of the figure.

First Figure 6 presents RMS error of GAME models. The first conclusion is that if there is low number of missing data (up to 5%), the GAME modeling method is able to create models comparable to models created from original non-missing data. And the performance does not dependent on the imputation method. For higher number of missing values the RMS error raises.

The lowest performance achieve GAME models for the DELETE imputation method. Performance is, even for 5% of missing data, statistically distinguishable from complete data models. For 25% and 50% of missing data we were unable to build any model because there were no training data left.

Also GAME models based on ZERO imputation method achieves high RMS errors for higher number of missing values. If there is more than 10% of missing values the GAME models performance of ZERO imputation methods is distinguishable from performance of GAME models over original data.

The performance of other methods seems to be comparable to GAME models generated over original data. In general both, the lower bound of confidence interval, mean value and deviance, slightly raises but still they can not be statistically distinguished.

The best method from previous experiment (AVGNEAR) is in this case on the same level as other methods (NEARESTN, RANDOM and MEAN). Our conclusion is that the GAME models are robust enough to handle noise and errors introduced by imputation these methods.

The second Figure 7 shows results of the GAME models with the ADVERT dataset. In this case the GAME models are comparable to the GAME model with original data up to 10% of missing values. More precisely results of GAME models for all imputation methods are not statistically distinguishable from the GAME model for original data up to 10% of missing data.

As in previous case, the performance of GAME models with the DELETE imputation method is quite poor and the error quickly raises.

The interesting difference between Figures 7 and 6 is the performance of ZERO and MEAN imputation method. In case of CARS dataset models with MEAN method are superior to models with ZERO method, but for ADVERT dataset the ZERO method is superior to MEAN method. This fact we find hard to interpret and we will perform more analysis to find out reasons.

The other imputation methods achieves the almost same results as in the previous cases.

ISBN 978-3-901608-32-2

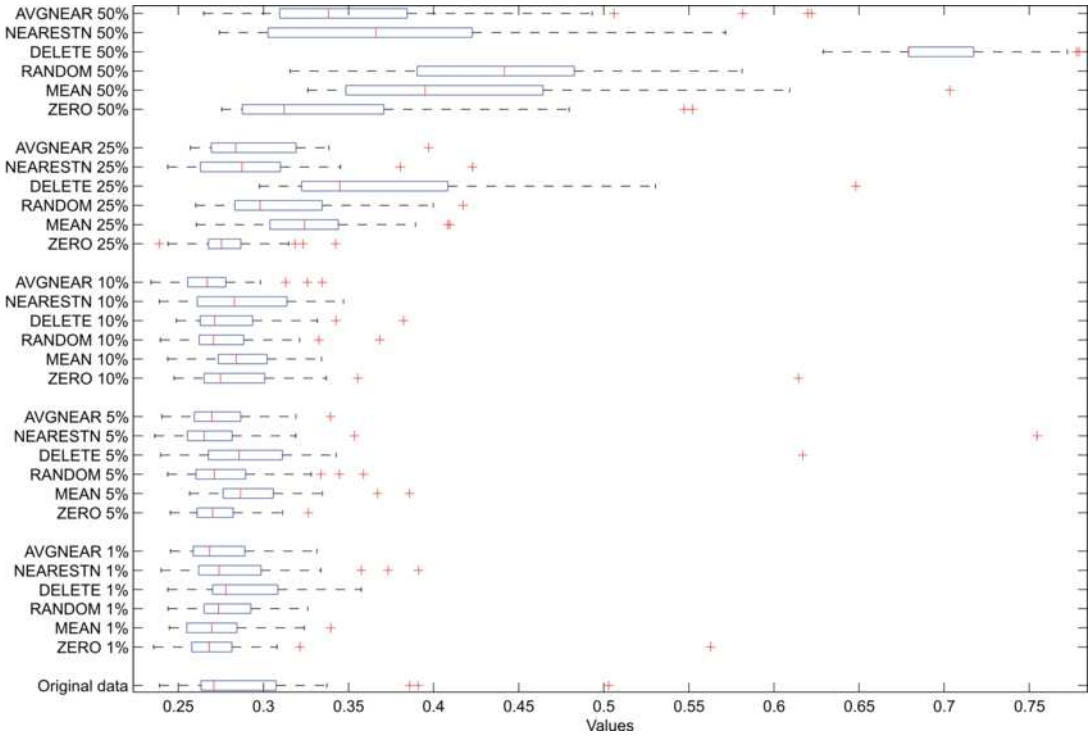

Fig. 7 GAME ANN RMS results as boxplot for Advert dataset. Imputation methods used to create given dataset is on the label on the left side, followed by ratio of missing values. Better values are on the left side of the figure.

For remaining datasets the imputation methods achives almost the same results as in above presented cases.

8 Conclusion

In this paper we compared several methods for imputing missing data. We tested imputation methods in two ways. In the first experiment we tested the data distortion introduced by imputation method. This we compared using RMS error computed between original and imputed data.

The method with the lowest data distortion we identified as the AVGNEAR method. This method imputes average of values of five nearest vectors. The AVGNEAR method is followed by MEAN and NEARESTN methods which achieve slightly higher errors. The worst method we identified as the ZERO method. This is because imputing still the same value, which is not the mean value, always produce high RMS error. The reason why the AVGNEAR method is superior to NEARESTN is because among irrelevant vectors also truly near vectors are selected and the average of several values will push the result to correct value.

In the second experiment we tested influence of imputation method to performance of the GAME modeling method. During this experiment we created 20 GAME models for each imputation method and we compared performance of these models. For regression and prediction problems we compared only RMS errors of the models. RMS error in this case is difference between value predicted by GAME model and value observed. Typical results are presented of Figures 7 and 6.

The main conclusion about GAME modeling method and imputation methods is that if there is low number of missing data (less then 10%) the performance of created GAME models do not depend on imputation method. In addition, their performance is not statistically distinguishable from performance of the GAME models with original data.

When there is more than 10% of missing data the performance of the GAME models degenerates and depends on imputation method. The worst results achieves DELETE imputation method. The reason is that it removes vectors from training set and the GAME model can not be created properly. In the worst case all vectors are removed from the training set and the model can not be created.

The MEAN and ZERO methods for some data work very well but sometimes they works quite bad. In addition, sometimes the MEAN method achieves lower error than ZERO method, but sometimes vice versa. The reason is not clear now and will be subject of further analysis. For this reason we cannot recommend these methods to be used with the GAME modeling method.

The same conclusion is for the RANDOM method, sometimes it achieves good result (for example CARS dataset) but sometimes its results are quite poor as in case of ADVERT dataset. Therefore we also can not recommend to use it with the GAME modeling method.

Remaining methods AVGNEAR and NEARESTN always achieves good results. Sometimes they are out-

ISBN 978-3-901608-32-2

performed by MEAN or ZERO imputation method but they always achieves good results and we did not noticed situations similar to failures of MEAN or ZERO methods. For this reason we recommend to use NEARESTN or AVGNEAR imputation method with the GAME modeling method.

9 Acknowledgment

This research is partially supported by the grant Automated Knowledge Extraction (KJB201210701) of the Grant Agency of the Academy of Science of the Czech Republic, Internal grant (CTU0706913) of Czech Technical University and the research program "Transdisciplinary Research in the Area of Biomedical Engineering II" (MSM6840770012) sponsored by the Ministry of Education, Youth and Sports of the Czech Republic.

10 References

[1] Mark Huisman. Imputation of missing item responses: Some simple techniques. *Quality and Quantity*, 34(4):331351, November 2000.

[2] Joseph L. Schafer and John W. Graham. Missing data: Our view of the state of the art. *Psychological Methdos*, 7(2):147177, 2002.

[3] John H. Holmes and Warren B. Bilker. *The Effect of Missing Data on Learning Classifier System Learning Rate and Classification Performance*, volume Learning Classifier Systems, chapter The Effect of Missing Data on Learning Classifier System Learning Rate and Classification Performance. Springer Berlin / Heidelberg, 2002.

[4] Joseph L. Schafer. Multiple imputation: a primer. *Statistical Methods in Medical Research*, 8:315, 1999.

[5] R. J. A. Little and D. B. Rubin. *Statistical analysis with missing data*. John Wiley, 1987.

[6] J. L. Schafer. *Analysis of Incomplete Multivatriate Data*. Chapman & Hall/CRC, 1997.

[7] I. G. Sande. Imputation in survays: Coping with reality. *The American Statistician*, 36:145152, 1982.

[8] R. J. A. Little and N. Schenker. *Missing data*, volume Handbook of Statistical Modeling for Social and Behavioral Sciences, chapter 3, page 3975. Plenum Press, 1995.

[9] J. A. Muller and F. Lemke. *Self-Organising Data Mining*. Berlin, 2000.

[10] H. Madala and A. Ivakhnenko. *Inductive Learning Algorithm for Complex System Modelling*. CRC Press, 1994.

[11] Pavel Kordk. *Fully Automated Knowledge Extraction using Group of AdaptiveModels Evolution*. PhD thesis, Czech Technical University, Prague, full text available at http://neuron.felk.cvut.cz/game/doc.html, March 2007.

[12] A. Schmitt, P. Murail, E. Cunha, and E. Rouge. Variability of the pattern of aging on the human skeleton: Evidence from bone indicators and implications on age at death estimation. *Journal of forensic Sciences*, 47:12031209, 2002.

[13] L. Prechelt. A set of neural network benchmark problems and rules. Technical Report 21/94, Karlsruhe, Germany, 1994.

[14] Uci machine learning repository, available at http://www.ics.uci.edu/ mlearn/mlrepository.html, September 2006.

[15] C.L. Blake D.J. Newman, S. Hettich and C.J. Merz. Repository of machine learning databases, 1998.

