Pattern Recognition 41 (2008) 3692 – 3705

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Impact of imputation of missing values on classification error for discrete data

Alireza Farhangfar[a], Lukasz Kurgan[b,*], Jennifer Dy[c]

[a]Department of Computing Sciences, University of Alberta, Edmonton, Alberta, Canada
[b]Department of Electrical and Computer Engineering, ECERF, 9107-116 Street, University of Alberta, Edmonton, Alberta, Canada T6G 2V4
[c]Department of Electrical and Computer Engineering, Northeastern University, Boston, MA 02115, USA

ARTICLE INFO

Article history:
Received 9 November 2006
Received in revised form 15 February 2008
Accepted 18 May 2008

Keywords:
Missing values
Classification
Imputation of missing values
Single imputation
Multiple imputations

ABSTRACT

Numerous industrial and research databases include missing values. It is not uncommon to encounter databases that have up to a half of the entries missing, making it very difficult to mine them using data analysis methods that can work only with complete data. A common way of dealing with this problem is to impute (fill-in) the missing values. This paper evaluates how the choice of different imputation methods affects the performance of classifiers that are subsequently used with the imputed data. The experiments here focus on discrete data. This paper studies the effect of missing data imputation using five single imputation methods (a mean method, a Hot deck method, a Naïve-Bayes method, and the latter two methods with a recently proposed imputation framework) and one multiple imputation method (a polytomous regression based method) on classification accuracy for six popular classifiers (RIPPER, C4.5, K-nearest-neighbor, support vector machine with polynomial and RBF kernels, and Naïve-Bayes) on 15 datasets. This experimental study shows that imputation with the tested methods on average improves classification accuracy when compared to classification without imputation. Although the results show that there is no universally best imputation method, Naïve-Bayes imputation is shown to give the best results for the RIPPER classifier for datasets with high amount (i.e., 40% and 50%) of missing data, polytomous regression imputation is shown to be the best for support vector machine classifier with polynomial kernel, and the application of the imputation framework is shown to be superior for the support vector machine with RBF kernel and K-nearest-neighbor. The analysis of the quality of the imputation with respect to varying amounts of missing data (i.e., between 5% and 50%) shows that all imputation methods, except for the mean imputation, improve classification error for data with more than 10% of missing data. Finally, some classifiers such as C4.5 and Naïve-Bayes were found to be missing data resistant, i.e., they can produce accurate classification in the presence of missing data, while other classifiers such as K-nearest-neighbor, SVMs and RIPPER benefit from the imputation.

1. Introduction

Many existing industrial and research datasets contain missing values. They are introduced due to various reasons, such as manual data entry procedures, equipment errors, and incorrect measurements. The simplest way of dealing with missing values is to discard the examples that contain the missing values. However, this method is practical only when the data contain relatively small number of examples with missing values and when analysis of the complete examples will not lead to serious bias during the inference. Another approach is to convert the missing values into a new value (encode them into a new numerical value), but such simplistic method was shown to lead to serious inference problems [1]. On the other hand, if a significant number of examples contain missing values for relatively small number of attributes, it may be beneficial to perform imputation (filling-in) of the missing values. Imputation methods are traditionally developed based on statistical algorithms that can be subdivided into two categories: (1) model based and (2) quasi-randomization inference based (data driven) [2–4]. Model based methods assume that the population quantities of interests are outcomes of random attributes (variables), indexed by unknown population parameters. Quasi-randomization procedures, on the other hand, assume that the population values are fixed, i.e., they are not governed by unknown parameters, and therefore are not the outcomes of random attributes. Statistical methods range from simple data driven methods such as mean imputation to complex model based methods that perform parameter estimation. Two popular model based imputation algorithms, i.e., regression and likelihood based, are described in Refs. [3,5]. In regression based imputation,

[*] Corresponding author. Tel.: +1 780 452 5562; fax: +1 780 492 1811.
E-mail addresses: farhang@cs.ualberta.ca (A. Farhangfar), lkurgan@ece.ualberta.ca (L. Kurgan), jdy@ece.neu.edu (J. Dy).

0031-3203/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2008.05.019

the missing values are predicted by a regression of the unobserved attribute values based on the observed values for a given example. The likelihood based methods work based on parameter estimation in the presence of missing data, i.e., the data are described based on the models and their parameters are estimated by maximum likelihood or maximum a posteriori procedures that use variants of the expectation maximization algorithm [3,6].

In recent years, machine learning (ML) algorithms were introduced to develop imputation methods [2,7–12]. In contrast to statistical methods, ML algorithms generate a data model from data that contain missing values, and next the model is used to perform classification that imputes the missing values. Several different types of ML algorithms were used, such as decision trees [2,10], probabilistic [7,9], and rule-based methods [9,11], however the underlying methodology was the same. One of the most recent advancements was a missing data imputation framework that was developed to improve the quality of imputation methods [8]. This framework serves as a wrapper that can be applied with most existing imputation methods (referred to as base methods) to improve their accuracy of imputation while preserving the asymptotic computational complexity of the base method. Section 2.1 provides details about the imputation methods that are used in this paper.

Once the missing data are imputed, it is crucial to evaluate the performance of the imputation method through determining the effect of imputation on subsequently performed classification. Prior studies on the impact of imputation on classification accuracy did not provide a comprehensive analysis and conclusions. We note that numerous databases that include significant, up to 50%, amount of missing data are in use, e.g., industrial database maintained by Honeywell [2] and a medical cystic fibrosis database [13], and that various classification algorithms can be used on these data. Despite these facts, the prior works failed to perform test with data that includes high number of missing data and to provide a systematic analysis of the quality of imputation with respect to different classifiers and amounts of missing data. The goal of this paper is to perform a comprehensive analysis of the impact of the imputation on classification accuracy. It performs experiments on 15 datasets, with six commonly used classification algorithms, six imputation methods, and six different amounts of missing data between 5% and 50%.

1.1. Related work

Four recent studies that investigated the impact of imputation on the accuracy of the subsequently performed classification are:

1. Acuna and Rodriquez [14] have investigated the effect of four methods that deal with missing values. These methods include case deletion, and three imputation methods: mean imputation, median imputation, and K-nearest-neighbor (KNN). The classification was performed using two methods: linear discriminant analysis (LDA) and KNN. Their results show that imputation does not have a significant effect on the accuracy of classification, which agree with relatively older results by Dixon [15]. Three main drawbacks of their study are: (1) only very basic single imputation methods were used, (2) relatively small amounts of missing data (i.e., between 1% and 20%) were considered, and (3) each dataset had a different amount of missing data, which makes it impossible to assess how the imputation affects classification across a range of different amounts of missing values.
2. Batista and Monard [16] tested the classification accuracy of two popular classifiers, i.e., C4.5 decision tree [17] and CN2 rule induction algorithm [18], and three imputation methods, namely, mean, mode, and KNN. The missing data were introduced only in a few selected attributes. The results show that KNN imputation results in good accuracy, but only when the attributes are not highly correlated to each other. The main drawbacks of their study are: (1) only very basic single imputation methods were used, (2) relatively small amounts of missing data, i.e., between 1% and 20%, were considered, (3) each dataset had a different amount of missing data, which makes it impossible to perform a comprehensive analysis, and (4) only four relatively small (up to 1500 examples) datasets were used in the tests.
3. Grzymala-Busse and Hu [11] investigated the accuracy of classification with learning from examples based on rough sets (LERS) classifier [19] and five imputation methods that include mode, C4.5 and LERS ML based imputations, and two other non-traditional methods. The results of classification on 10 datasets show that on average imputation helps to improve classification accuracy, and the best imputation was achieved with the C4.5 ML based method. The main drawbacks are: (1) only one classifier (in two versions) was used to test the accuracy, (2) low amounts of missing data, i.e., between 1% and 13%, were considered, and (3) each dataset included a different amount of missing data, which makes it difficult to draw comprehensive conclusions.
4. Mundfrom and Whitcomb [20] used two classifiers: linear discriminant function and logistic regression, with just one dataset to test the impact of three imputation methods (mean, Hot deck, and regression methods). The classification accuracy on the imputed data shows thesuperiority of the mean and Hot deck imputations. However, this was by far the smallest study, which included only one small dataset and considered low, about 11%, amount of missing data, and thus these results may not generalize to other settings.

The above studies are summarized in Table 1. The relevant imputation methods are described in Section 2.

1.2. Proposed work

An analysis of Table 1 reveals the lack of a truly comprehensive study. To this end, this paper provides:

- The largest number of popular and modern classifiers, namely, RIPPER [21,22], C4.5 [17], KNN, support vector machine [23,24], and Naïve-Bayes. In this study we selected representative classifiers that belong to major families of ML algorithms: C4.5 is a decision tree, KNN is an instance-based method, RIPPER is a rule-based classifier, Naïve-Bayes is a probabilistic method, and support vector machine is a kernel-based classifier.
- The largest number of imputation methods (i.e., six methods that include both single and multiple imputations) and a newly proposed framework for improving the quality of imputation [8].
- The widest range of consistent amounts of missing data (i.e., missing data amounts of 5%, 10%, 20%, 30%, 40%, and 50%) for all datasets. This allows for a wide range of evaluation of the quality of imputation with respect to the amount of missing data.
- The largest number of test datasets—15 datasets ranging between 47 and 28,000 examples, 7 and 61 attributes, and 2 and 17 classes.

The goal of this paper is to present a comprehensive study of the impact of imputation of missing values on classification accuracy of several leading classifiers and for varying amounts of missing data. We note that although some of these classifiers including C4.5 have their own internal approaches of handling the missing values, it is not clear how they would react to external imputation methods.

The paper is organized as follows. Section 2 provides a background on imputation and describes imputation methods that are used in this paper. Section 3 explains details of the experimental study, and presents and analyzes the results. Finally, Section 4 summarizes the paper.

Table 1
Summary of the related contributions

Reference		Acuna et al. [14]	Batista et al. [16]	Grzymala-Busse et al. [11]	Mundfrom et al. [20]	This paper
Objective		Evaluation of the relative performance of imputation methods by comparison of the classification error on the imputed data				
Imputation method	Number	3	3	5	3	6
	Method names	1—Mean 2—Median 3—K-nearest-neighbor	1—Mean 2—Mode 3—K-nearest-neighbor	1—Mode and supervised (uses class attrib.) mode 2—C4.5 based 3—Based on assigning all possible values of the attribute 4—Event covering 5—LEM2 based	1—Mean 2—Regression based 3—Hot deck	1—Hot deck 2—Imputation framework with Hot deck 3—Naïve-Bayes 4—Imputation framework with Naïve-Bayes 5—Polynomial multiple regression 6-Mean imputation
Classifiers	Number	2	2	2	2	6
	Classifier names	1—Linear discriminant analysis 2—K-nearest-neighbor	1—C4.5 decision tree 2—CN2	1—New LERS 2—Naïve LERS	1—Linear discriminant analysis 2—Logistic regression	1—RIPPER 2—C4.5 3—SVM (RBF kernel) 4—SVM (polynomial kernel) 5-K-nearest-neighbor 6-Naïve-Bayes
#Test datasets		12	4	10	1	15
Amount of missing values		Varies between 1% to 20%	Varies between 1% to 20%	Varies between 1% to 13%	11%	5%, 10%, 20%, 30%, 40%, 50%
Comments		The amount of missing values introduced in each dataset was different, which makes it impossible to compare classification performance across varying amounts. Only very basic imputation methods are used	Missing data was introduced only in three attributes in each dataset, while datasets have different total number of attributes, which makes it impossible to perform a comprehensive analysis. Four small datasets (up to 1500 samples) were used. Only very basic imputation methods are used	The amount of missing values introduced in each dataset was different, which makes it impossible to perform a comprehensive analysis. Only one classifier (in two versions) was used. Non-traditional imputation methods were tested	Values for one attribute at a time were deleted for each of the 99 examples (patients). Only one small dataset was used. Only very basic imputation methods are used	Wide range and consistent, across large number of datasets, amounts of missing values are used. Both, single and multiple imputation methods are used. The largest number of well-known different types of classifiers is used and compared to previous works

2. Background

Three different mechanisms, which lead to the introduction of missing values, can be categorized as [2,3] follows:

1. Missing completely at random (MCAR), when the distribution of an example having a missing value for an attribute does not depend on either the observed data or the missing data. For example, in a dataset that includes student marks, a student's final grade is missing, and this does not depend on his/her status (for instance if this is a graduate or undergraduate student) or final grade of other students (for instance, if the other complete final marks are low or high).
2. Missing at random (MAR), when the distribution of an example having a missing value for an attribute depends on the observed data, but does not depend on the missing data. For example, student's final mark is missing, and this does depend on his/her status, but it does not depend on the final grade. Therefore, the missing final marks can be filled-in (predicted) using information about the student's status.
3. Not missing at random (NMAR), when the distribution of an example having a missing value for an attribute depends on the missing values. For instance, student's final grade is missing, and this does depend on the final grade (i.e., only grades in a special range, say 80–90%, are missing). This way, the missing value can be filled-in using the complete final marks of the other students.

In case of the MCAR mode, the assumption is that the distributions of missing and complete data are the same, while for MAR mode they are different, and the missing data can be predicted by using the complete data [3]. MCAR mechanism is assumed by most of the existing missing data imputation methods [1,2,25–27], and thus it is also assumed in this paper. Furthermore, considering the three mechanisms, it is only in the MCAR mechanism case where the analysis of the remaining complete data could give a valid inference (classification) due to the assumption of equal distributions. Both of the other mechanisms could potentially lead to information loss that would lead to the generation of a biased/incorrect classifier (classifier based on a different distribution). In the case of NMAR and MAR, if we have prior information about the mechanism that leads to the introduction of the missing values then we can use this background knowledge to directly impute the missing data. For instance in NMAR (following the example above), if we know that only grades in a special range, say 80–90%, are missing then we would impute all missing marks as 85%. In the case where the underlying mechanism is unknown, the user can perform a statistical test by Chen and Little [28] that can determine whether the missing values were introduced as MCAR. The NMAR mechanism is rarely applicable in practice.

2.1. Imputation methods used in this paper

This paper investigates representative methods from the three imputation mainstreams: a statistical model based regression imputation, a statistical data driven (quasi-randomization) Hot deck imputation, and an ML based method based on the probabilistic Naïve-Bayes algorithm. The first method is a multiple imputation algorithm, while the latter two are single imputation algorithms.

2.1.1. Single imputation algorithms

In single imputation methods, a missing value is imputed by a single value. The single imputation methods used in this paper are:

I. *Hot deck*: In Hot deck imputation, for each example that contains missing values, the most similar example is found, and the missing values are imputed from that example. If the most similar example also contains missing values for the same attributes as the missing information in the original example, then it is discarded and a second closest example is found. This is repeated until all the missing values are successfully imputed or the entire database are searched. In case when there is no similar example with the required values filled in, the closest example with the minimum number of missing values is chosen to impute the missing values. There are several ways of finding the most similar example to the example with missing values [3,10,29]. In this study, the distance function, which is used to measure the dissimilarity between two examples, assumes distance of 0 between two attributes if both have the same numerical or nominal values, otherwise the distance is set to 1, and these distances are summed over all attributes. A distance of 1 is also assumed for an attribute, for which any of the two examples has a missing value [8].

II. *Mean*: In mean imputation, the missing values are imputed with the mean for continuous data or the most frequent value (mode) for discrete data of the corresponding attribute.

III. *Naïve-Bayes*: Naïve-Bayes is a simple probabilistic classification technique [30]. The Naïve-Bayes classifier applies the simplistic assumption that the feature or attribute values are conditionally independent given the class, $P(a_1, a_2, \dots, a_d|c) = \prod_{i=1}^{d} P(a_i|c)$, where a_i is the ith attribute, c represents the class, and d the number of attributes. This assumption does not hold true in most real datasets (thus, the term naïve), however it has been shown to work well in practice. Naïve-Bayes requires only one pass through the training dataset, which makes it computationally efficient. This algorithm is applied to perform imputation in the following manner. During training, the conditional probabilities $P(a_i|c)$ and the prior probabilities $P(c)$ are estimated. We then classify each new instance by: $\arg\max_c P(c|a_1, a_2, \dots, a_d) = \arg\max_c P(c) \prod_{i=1}^{d} P(a_i|c)$. To perform imputation, we treat each attribute that contains missing values as the class attribute, then fill each missing value for the selected class attribute with the class predicted from the conditional probabilities established during training.

2.1.2. Multiple imputation algorithms

In multiple imputation methods, several, usually likelihood, ordered choices for imputing the missing value are computed [1,31,32]. This way, several complete databases are created by imputing different values to reflect uncertainty about the right values to impute. At the next step, each of the databases is analyzed by standard procedures specific for handling complete data, and the analyses for each database are combined into a final result [5,33]. Several multivariate multiple imputation methods were developed by different researchers. Li [34] and Rubin and Schafer [35] used probabilistic Bayesian models that compute imputations from the posterior probabilities of the missing data based on the complete data. The Rubin–Schafer method assumes multivariate normal distribution of the data and the MAR mechanism. On the other hand, Alzola and Harrell [36] introduced a function that imputes each incomplete attribute by cubic spline regression given all the other attributes, without assuming that the data must be modeled by the multivariate distribution. A multiple imputation environment called multivariate imputation by chained equations (MICE), which provides a full spectrum of conditional distributions and related regression based methods, was developed by Buuren and Oudshoorn [33,37]. MICE incorporates logistic regression, polytomous regression, and linear regression, uses Gibbs sampler [38] to generate multiple imputation, and is furnished with a comprehensive, state-of-the-art missing data imputation software package. For imputation of numerical attributes, MICE offers Bayesian linear regression imputation with normal errors, predictive mean matching, and unconditional mean imputation. For categorical attributes, MICE provides logistic

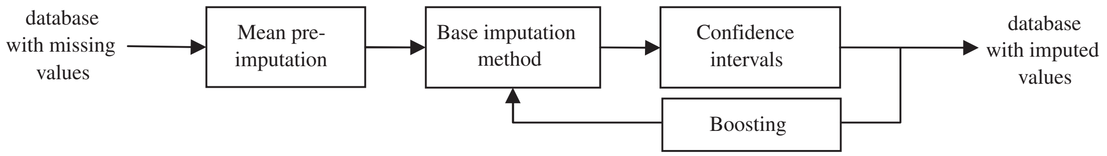

Fig. 1. Architecture of the applied imputation framework.

Table 2
Summary of the imputation methods used in this paper

Name of the method	Algorithm	Works with discrete attributes?	Works with continuous attributes?
‘Naïve-Bayes	Bayesian algorithm	Yes	No
Framework with Naïve-Bayes	Bayesian algorithm	Yes	No
Hot deck	Nearest neighbor	Yes	Yes
Framework with Hot deck	Nearest neighbor	Yes	Yes
Polytomous regression	Regression	Yes; logistic regression (binary)/polytomous logistic regression (categorical)	Yes
Mean	Statistical	Yes	Yes

regression (for binary attributes), polytomous logistic regression, and LDA. It also provides a simple random imputation, which may be useful for missing data in the MCAR mode. This paper uses a MICE's multiple imputation based on regression. We use logistic regression for imputation of binary attributes and polytomous logistic regression for the discrete categorical attributes.

Logistic and polytomous logistic regressions: In this case, imputation is performed by regression of the missing values using the complete values for a given example [39]. Several regression models can be used, including linear, logistic, polytomous, etc. To define the logistic regression model we apply the following notation: example x is described by attributes a_i, where $i = 1, 2, \ldots, d$ and d is the number of attributes, class attribute is denoted as c_j, where $j = 1, 2, \ldots, m$ and m is the number of classes, such that $c_j = 1$ if x corresponds to an example that belongs to class j. Using the polytomous logistic regression model (also known as the multinomial logistic regression model), the probability that x belongs to class j is defined as $P(c_j = 1|x, w) = \frac{e^{w_j x}}{\sum_{k=1}^{m}(e^{w_k x})}$, where w_j is a weight vector corresponding to class j. The parameters w_j, $j = 1, 2, \ldots, m$ are estimated from the training data using maximum likelihood estimation. Because of the normalization condition $\sum_{j=1}^{m} P(c_j = 1|x, w) = 1$, the weight vector for one of the classes need not be estimated. For binary class attributes, i.e., m=2, the above model reduces to the logistic regression model. Logistic regression is applied by assuming binary attributes with missing values as the class attributes, and polytomous regression is used for discrete attributes with missing values.

2.1.3. *Imputation framework*

In this paper we use a recently proposed imputation framework [8]. This framework is designed to improve the performance of single imputation methods like Hot deck and Naïve-Bayes. As a result of applying the framework to Hot deck and Naïve-Bayes, two more single imputation methods called "framework with Hot deck (FHD)" and "framework with Naïve-Bayes (FNB)" are added to this study. The framework consists of four modules: mean pre-imputation, base imputation, confidence intervals, and boosting (see Fig. 1). The effect of each module on the overall performance of the imputation performed with the framework is further investigated in Ref. [8].

I. *Mean pre-imputation*: The first component of the framework is the mean pre-imputation. In this module, the missing values are temporarily imputed with the mean of the corresponding attribute.
II. *Base imputation method*: Next, each missing pre-imputed value is imputed using a base imputation method, which in our case is either Hot deck or Naïve-Bayes method, and the imputed value is filtered by using the confidence intervals component.
III. *Confidence intervals*: Confidence intervals are used to select the most probable imputed values, while rejecting possible outlier imputations. This filter is based on the premise that imputed values, which are close to the mean (for numerical attributes) or mode (for nominal attributes) of an attribute, have the highest probability of being correct. They are defined as an interval estimate for the mean of a corresponding attribute [27]. Based on Ref. [8], the filter removes imputed values with a frequency lower than the average for the attribute (the confidence intervals are computed individually for each of the classes), while the remaining imputed values are kept. The frequency is computed as the number of times a given value is found in the dataset, which is normalized by the number of values of the most frequent value.
IV. *Boosting*: Once all the values are imputed and filtered, each of them is assigned with a weight that quantifies its quality. This weight might be expressed as a probability or a similarity measure. The imputed values are accepted or rejected by the boosting component based on their weight and some threshold. In the case of the Naïve-Bayes base imputation method, the weights are defined as the probabilities of the selected class attributes, i.e., the posterior class probability for the imputed value. We set the threshold to be the mean value of the selected class probability for all imputed values. All values with weights above the threshold are accepted, while the remaining values are rejected. Similarly, for the Hot deck base imputation method, the weights are defined as the distance between the example with the currently imputed value and the example from which the imputed value was taken. The threshold is set as the average distance between the examples with missing data and their closest examples for all the imputed values. The imputed values are accepted when their weights are less than the threshold and rejected otherwise. As a result, a partially imputed database is created and fed back to the base imputation algorithm, and the process repeats. The boosting component, similar to a boosting algorithm that inspired its design, aims to improve the accuracy of the imputation by accepting only the high quality imputed values and using them, i.e., the additional and reliable information, to impute the remaining values. After 10 boosting iterations, all remaining imputed values are accepted, and the algorithm outputs the imputed data. More details about the imputation framework can be found in Ref. [8].

Table 2 summarizes the imputation methods, which are used in this study. Some of the considered imputation methods work only with discrete values. Therefore, to include all these methods and assure comprehensiveness in terms of the number and types of imputation and classification algorithms, we focus our study on discrete data. The study on continuous data constitutes our future work.

3. Experiments and results

The main objective of the experiments is to empirically evaluate the effect of missing data imputation on the accuracy of subsequent classification. We start by describing the datasets and experimental setup, and follow up with experimental results and their analysis.

3.1. Datasets

The experiments were performed using 15 benchmark datasets selected from the UCI ML repository [40] and the KDD repository [41]. Each dataset is described by a set of characteristics such as number of data samples, attributes and classes. We provide this information in Table 3. The selected datasets include only discrete data (i.e., discrete numerical and categorical data) and cover a full spectrum of values for each of the characteristics.

Missing data were introduced randomly, using the MCAR mechanism, into each of the datasets. The missing values were introduced into all attributes in all datasets in the following six amounts: 5%, 10%, 20%, 30%, 40%, and 50%.

3.2. Experimental setup

In general, the experiments were performed as follows. Each dataset was first randomly divided into equal sized training and test subsets and the six different amounts of missing values were introduced in the training subset. Next, the missing values in the training set were imputed using the six single and multiple imputation methods and the resulting datasets were used with the six classifiers: RIPPER, C4.5, SVM with polynomial kernel, SVM with RBF kernel, Naïve-Bayes, and KNN. Finally, the classification accuracy of the classifiers was evaluated by applying the corresponding classification model on the test set, as shown in Fig. 2 (*classification with imputation* procedure). The results of the above experiments were compared with two experimental setups, in which the imputation was not performed: when the classifiers were trained on data with missing values, and when the classification was performed on the complete data. The former experiment, as shown in Fig. 3 (*classification without imputation* procedure), establishes a lower limit on accuracy, which should be improved by imputation, and which was possible since the selected classifiers can generate models directly from the data with the missing values. The latter experiment, as shown in Fig. 4 (*classification with complete data* procedure), gives an upper limit on accuracy and was possible since the considered datasets were originally complete.

The executed experiments, which include 15 datasets, six amounts of missing values, six imputation methods and one set of experiments without imputation, and six classifiers, gives us a total of $15 * 7 * 6 * 6 = 3780$ experiments. Additionally, $15 * 6 = 90$ experiments were performed with the complete data.

3.3. Experimental results

The experiments report the classification error rate against the six amounts of missing values, for six different classifiers namely

Table 3
Description of the datasets used in the experiments

Name	Abbreviation	#Examples	#Attributes	#Classes
Soybean (small)	Soy	47	36	4
Postoperative patient data	Pos	87	9	3
Promoters	Pro	106	58	2
Monks 1	Mk1	432	7	2
Monks 2	Mk2	432	7	2
Monks 3	Mk3	432	7	2
Balance	Bal	625	5	3
Tic-tac-toe	Tic	958	10	2
CMC	Cmc	1473	10	3
Car	Car	1728	7	4
Splice	Spl	3190	61	3
Kr-vs-kp	Krs	3196	36	2
LED	Led	6000	8	10
Nursery	Nrs	12960	9	5
Kr-V-K	Krv	28056	7	17

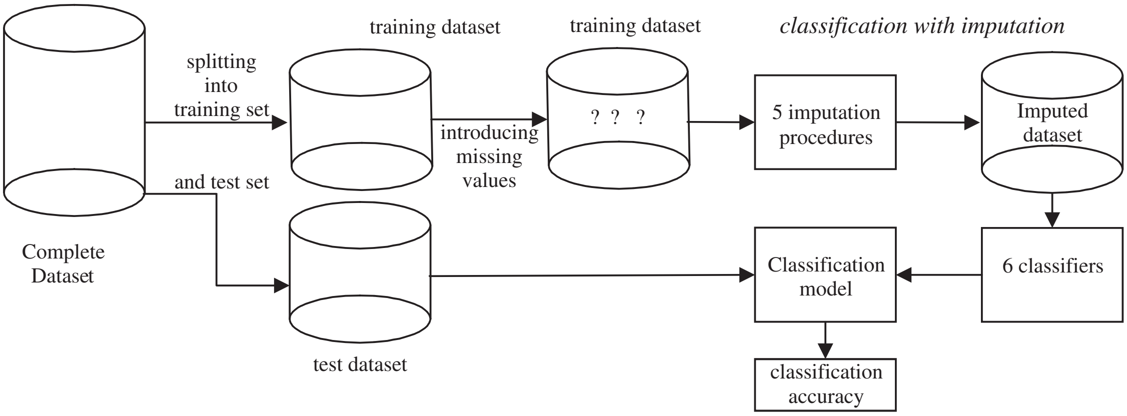

Fig. 2. Experimental procedure that includes classification with imputation (? denotes missing values).

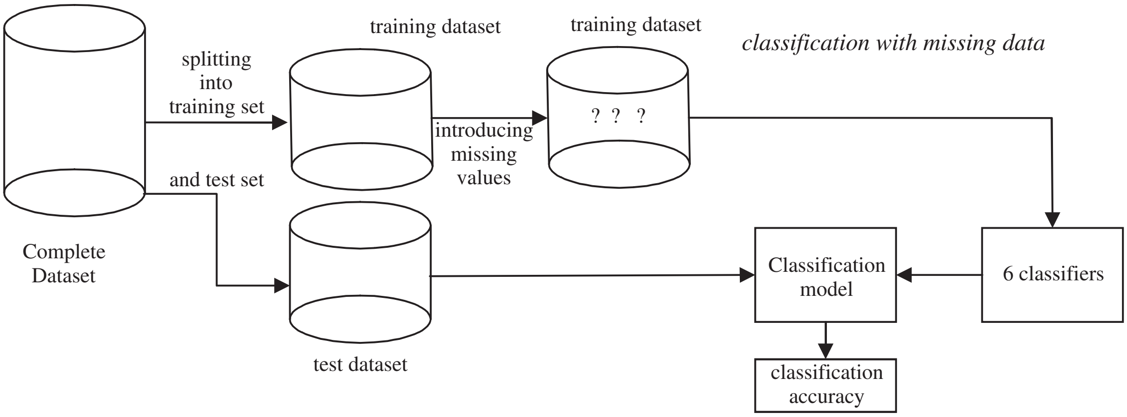

Fig. 3. Experimental procedure that includes classification without imputation (? denotes missing values).

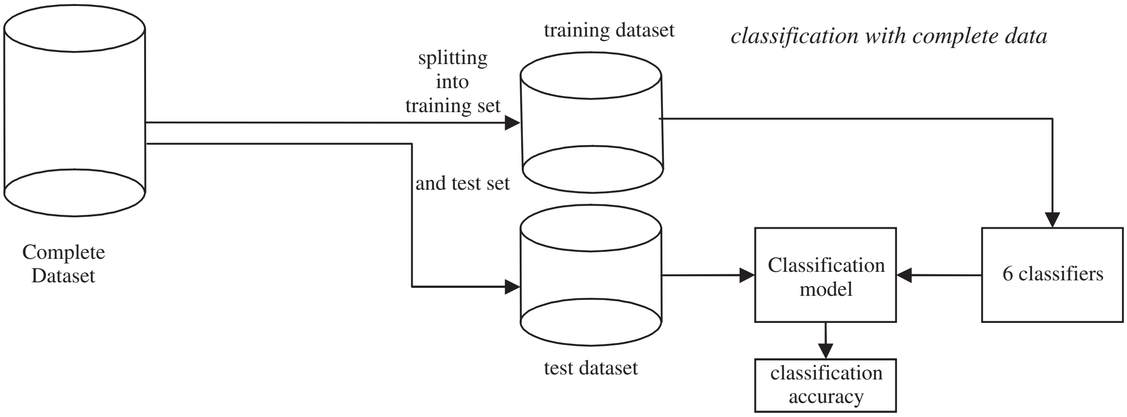

Fig. 4. Experimental procedure that includes classification with complete data.

RIPPER, C4.5, SVM with polynomial kernel (SVMP), SVM with RBF kernel (SVMR), Naïve-Bayes (NB) and KNN, and six imputation methods, Hot deck (HD), imputation framework with Hot deck (FHD), Naïve-Bayes (NB), imputation framework with Naïve-Bayes (FNB), multiple imputation Polytomous regression (Poly), and mean imputation (mean). These results are compared against classification on the missing data that was not processed through imputation (missing) and on complete data before the missing values were introduced (complete). We used WEKA's [42] implementation for all classifiers in this study except for KNN, for which we used an in-house implementation. All the parameters are set as the WEKA's default, and for KNN, five neighbors are considered.

In these experiments the classification error is measured based on a zero–one loss, which is commonly used to evaluate the performance of classifiers [43]. Although some datasets may assume different costs for their classification decisions, we assume a uniform cost for all the classes to be able to compare the results across different datasets. A future extension to this work would consider different classes to have different associated cost functions.

Table 4 presents the average classification improvement (negative sign shows deterioration) resulting from the imputation of missing data. The improvement is defined as a difference between the classification accuracy on missing data and accuracy on the imputed data. The best results for imputed data and complete data are highlighted in bold. We note that on average, over all amounts of missing data, the highest improvement is 6.52% for 20% missing data imputed by the Poly method and classified by RIPPER. In this case, classification on complete data would have given us 7.8% improvement compared to missing data that is fairly close to 6.52% that was obtained on imputed data. This shows that imputed data can render similar classification performance as that of complete data.

A side-by-side comparison between different imputation methods, irrespective of the classification algorithms, is given in Fig. 5. It shows the average, over the six classifiers and the 15 datasets, improvement in classification error resulting from imputation against different amounts of missing values. The improvements are provided for the six imputation methods and data without missing values (complete).

As expected, the classification error rates with the imputed data range between the results obtained on the complete data and the results on data without imputation. Fig. 5 shows that, on average, classification on imputed data with different classifiers is more accurate than classification on missing data. Therefore, we conclude that on average, across most considered amounts of missing values (above 5%) and classifiers, imputation improves the classification. In few isolated cases, which include the use of mean imputation method on data with 5%, 10%, 30%, and 40% of missing data, the imputation deteriorates the classification accuracy. Polytomous regression achieves on average the lowest classification error for several amounts of missing values (10%, 20%, and 50%), while Naïve-Bayes achieves the best performance for 40% amounts of missing data.

The maximal (depicted by light gray bars) and average (depicted by dark gray bars) improvements as well as maximum loss (depicted by black bars) of the classification error rates obtained by

Table 4
Average classification improvement (negative sign shows deterioration) resulting from imputation of missing data for the 15 datasets, six classifiers, six imputation methods, and six amounts of missing data

Imputation methods	Missing data amounts (%)	Classifiers											
		RIPPER		C4.5		SVMR		SVMP		KNN		NB	
		Mean	Stdev	Mean	Stdev	Mean	Stdev	Mean	Stdev	Mean	Stdev	Mean	Stdev
FHD	5	−0.89	2.15	−2.24	4.55	0.82	2.42	0.28	3.39	−0.04	3.77	0.06	1.14
	10	0.79	1.70	−2.03	5.13	2.10	5.43	−1.48	3.84	1.93	6.02	−0.87	2.45
	20	5.59	8.38	−0.89	3.57	0.63	2.63	1.71	5.66	2.69	5.24	−0.54	2.69
	30	2.20	2.30	−0.49	6.84	1.99	6.47	0.44	5.34	1.69	7.96	−0.77	3.37
	40	1.83	10.35	−1.80	6.14	0.52	3.72	0.00	5.30	1.71	6.42	−1.89	4.16
	50	3.55	13.30	−0.64	8.68	0.38	4.02	−1.34	8.08	4.98	9.35	−2.31	3.79
HD	5	0.37	2.48	−1.70	6.36	−0.59	2.44	−0.34	5.44	−0.41	3.63	−0.02	0.83
	10	0.84	2.23	−2.03	5.01	1.24	5.66	−0.93	5.26	1.93	5.83	0.25	0.84
	20	5.48	10.26	−2.63	3.90	1.37	5.63	2.07	5.58	1.57	5.01	0.09	1.75
	30	1.31	4.21	−2.41	7.56	1.99	6.02	1.51	6.69	0.80	7.20	−1.45	2.03
	40	1.81	11.56	−2.69	6.14	0.63	2.24	−0.47	4.95	1.49	6.30	−2.57	3.65
	50	2.75	12.00	−0.18	5.19	1.05	1.55	−0.88	7.83	2.24	7.20	−1.73	4.07
FNB	5	−0.04	1.63	−1.79	4.45	0.42	2.50	−0.04	2.54	0.13	3.56	−0.36	0.96
	10	0.70	1.85	−2.09	4.61	1.99	5.45	−0.76	3.93	1.79	6.48	−1.57	2.59
	20	4.36	8.47	−1.57	3.38	1.93	6.07	1.42	4.89	2.41	4.82	−0.63	2.48
	30	**2.53**	3.51	−0.91	5.87	2.38	6.46	1.56	5.96	1.91	7.19	−1.00	2.27
	40	2.03	11.14	−0.60	6.06	2.49	6.39	−0.12	5.35	1.31	6.97	−1.36	4.57
	50	3.24	12.58	−0.50	8.09	2.04	6.44	−0.31	6.65	4.27	9.97	−2.09	4.58
NB	5	−0.48	1.56	−1.91	5.32	−0.12	2.33	0.34	3.15	−0.07	3.32	−0.25	1.75
	10	1.56	4.21	−2.62	3.62	1.21	5.40	−1.23	3.25	1.61	6.57	−0.21	1.07
	20	4.49	8.36	−1.46	4.01	0.55	3.88	2.63	4.26	3.23	5.02	−0.77	1.72
	30	2.35	3.55	−0.96	5.83	1.31	2.22	1.37	5.33	2.06	7.63	−0.18	3.37
	40	**3.00**	11.81	0.77	3.88	1.60	3.02	1.64	3.65	1.10	7.49	−1.50	2.74
	50	5.64	11.75	−1.73	9.44	0.96	2.98	0.06	3.83	2.68	4.55	−1.19	2.21
Poly	5	−0.94	2.14	−1.67	5.35	−0.04	2.21	**1.13**	2.67	0.13	3.89	−0.11	1.81
	10	1.80	4.46	−0.94	2.54	−0.20	3.97	2.00	5.16	**2.44**	6.22	−0.20	1.06
	20	**6.52**	10.14	−0.15	4.58	0.89	2.08	2.98	7.32	2.31	5.59	−0.64	1.41
	30	1.60	5.18	−0.37	5.98	0.69	2.40	1.91	7.56	1.90	9.18	−0.51	1.85
	40	0.50	17.31	−3.58	10.18	1.22	2.05	0.89	5.71	0.30	8.22	−1.54	3.26
	50	**6.06**	11.78	1.13	5.99	0.90	3.78	1.50	6.89	2.78	7.84	−0.54	2.59
Mean	5	−0.05	2.64	0.01	2.46	−0.69	1.55	0.02	0.76	0.25	4.09	−0.09	1.27
	10	0.65	1.61	−1.94	4.81	−0.56	2.75	−0.04	0.59	1.31	5.98	−0.77	2.61
	20	5.81	9.97	−2.96	3.46	−0.51	2.47	0.30	1.12	1.87	8.11	−1.01	2.24
	30	0.56	5.34	−3.12	5.30	−0.05	2.83	0.02	0.08	0.67	5.44	−1.43	3.60
	40	2.87	9.23	−3.25	7.56	0.01	0.20	0.00	1.39	0.13	5.04	−1.43	2.99
	50	5.55	12.75	1.20	3.31	0.37	1.45	−0.05	2.32	2.06	9.17	−1.25	2.99
Complete data (improvement over classification with a given amount of missing data)	5	0.25	2.03	0.25	2.84	0.19	2.27	**2.10**	5.39	0.52	4.12	0.02	2.02
	10	**2.73**	4.39	0.52	3.71	1.63	5.62	2.50	7.45	2.24	7.46	−0.38	1.56
	20	**7.80**	9.93	2.24	2.58	2.75	6.45	6.13	13.01	4.29	5.67	0.84	1.95
	30	6.29	7.89	3.75	6.91	3.58	6.28	**6.83**	15.04	4.23	10.74	0.10	2.84
	40	**8.59**	12.97	5.01	8.32	3.96	5.50	6.54	15.12	4.28	10.20	−0.22	2.04
	50	**11.51**	17.24	6.80	8.97	4.25	5.55	7.38	17.02	7.27	11.76	0.31	3.42

Best results are shown in bold.

imputing missing values (when compared with classification on missing data) are shown in Fig. 6. On average, imputation reduces the error rates by about 1%, while individual improvements (computed over all datasets) can be as high as 6.52% for RIPPER classifier and Poly imputation method for 20% of missing values, and individual accuracy losses due to imputation (computed over all datasets) can be as high as 3.58% for C4.5 classifier and Poly imputation method for 40% of missing values. Later in this section we show that imputation in general helps RIPPER classifier to improve its performance and therefore is recommended. On the other hand C4.5 can handle the missing values internally and imputation in some cases may result in the loss of classification accuracy. Two of the other maximal losses (about 3% loss) for 20% and 30% of missing values are due to the use of mean imputation method, which is shown to be outperformed by other considered imputation methods; see Sections 3.4 and 3.5.

3.4. Analysis of the results with respect to different classifiers

We show a detailed analysis of the results from Section 3.3, which considers individual classifiers and different amounts of missing data. We analyze the statistical significance of differences in accuracy between using the imputation methods and directly applying the data with missing values based on paired *t*-tests at the 95% significance level. In this experiment we compare whether mean accuracies of the prediction for a given classifier and given imputation method with and without the imputation are different. The mean is computed over a set of 15 datasets. We use the "paired" *t*-test in which each member of one numerical set is assumed to have a unique relationship with a particular member of the other set. Although we have 15 different datasets, their classification accuracy is computed using the same imputation procedure for each dataset (we compare results for the same dataset with and without imputed missing values). The statistical significance of the classification error

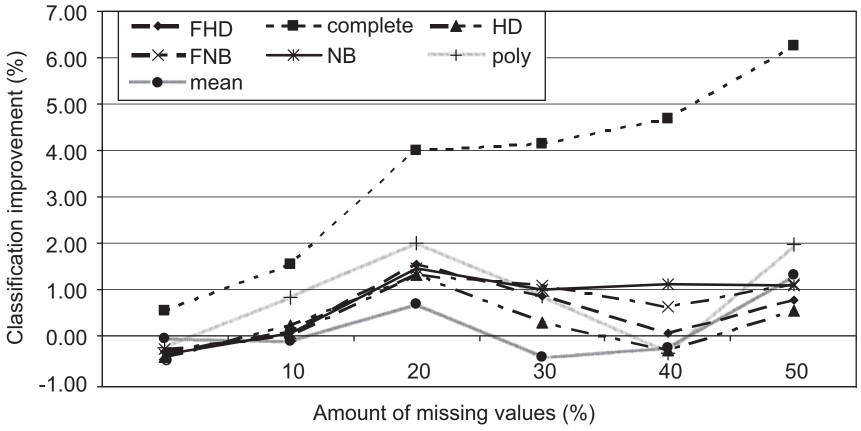

Fig. 5. Average, over the six classifiers, classification error rates.

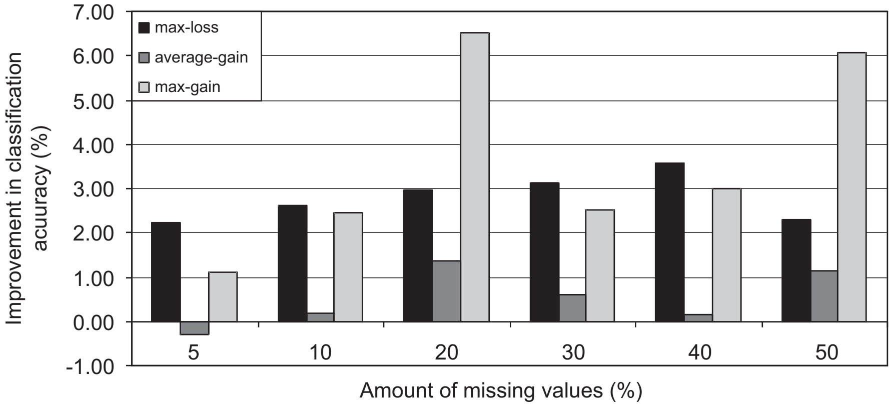

Fig. 6. Average improvement in classification error rates due to using imputation when compared with classification on missing data.

Table 5
Statistical significance of the difference between classification on imputed data and on data with missing values, which were classified with each of the considered classifiers (columns)

Imputation method	Classifiers						
	RIPPER significance (*t*-value)	C4.5 significance (*t*-value)	SVMR significance (*t*-value)	SVMP significance (*t*-value)	NB significance (*t*-value)	KNN significance (*t*-value)	Average significance (*t*-value)
FHD	+ + (2.4)	− − (−3.2)	+ + (3.4)	~ (−0.1)	− − (−3.3)	+ + (3.2)	~ (1.8)
HD	+ + (2.8)	− − (−4.7)	+ + (2.7)	~ (−0.1)	− − (−2.3)	+ + (3.3)	~ (−1.0)
FNB	+ + (3.3)	− − (−3.7)	+ + (6.2)	~ (0.7)	− − (−4.2)	+ + (3.5)	+ + (2.4)
NB	+ + (3.1)	− − (−2.8)	+ + (3.6)	~ (1.3)	− − (−3.4)	+ + (3.7)	+ + (2.4)
Poly	+ + (2.1)	~ (−1.1)	+ + (2.7)	+ + (6.1)	− − (−2.8)	+ + (3.5)	~ (2.0)
Mean	+ + (2.4)	−(−2.2)	~ (0.8)	~ (0.7)	− − (−4.8)	+ + (3.1)	~ (0.7)

"++" indicates that using a given imputation method (rows) gives statistically significantly better classification errors, "−−" indicates that it gives statistically significantly worse classification errors, "~" indicates that the difference in the errors is insignificant; positive *t*-value means that classification errors for the imputed data were better, while negative values means that classification errors for the data with missing values were better.

differences for different classifiers and between using imputed and missing data is summarized in Table 5.

The results show that the impact of the imputation varies for different classifiers. The largest improvements are achieved for SVMR, KNN, RIPPER, and SVMP. The results for C4.5 and Naïve-Bayes show that imputation does not improve the subsequent classification and indicate that C4.5 and Naïve-Bayes are on average missing data resistant, i.e., they can produce accurate classification in the presence of missing data. The imputation performed for the SVMR, RIPPER, and KNN classifiers almost always results in significantly better classification error, except for mean imputation for SVMR. In the case of SVMP, the Poly imputation method provides a significant improvement, while the remaining imputation methods for this classifier result in statistically insignificant differences. The last column in Table 5 shows the average improvement for all the six classifiers.

Fig. 7 shows the classification improvement with imputed data against increasing amounts of missing values for the RIPPER, C4.5, SVMR, SVMP, Naïve-Bayes, and KNN classifiers, respectively. Similar to Fig. 5, it shows the classification improvements for the six imputation methods, as well as the complete data.

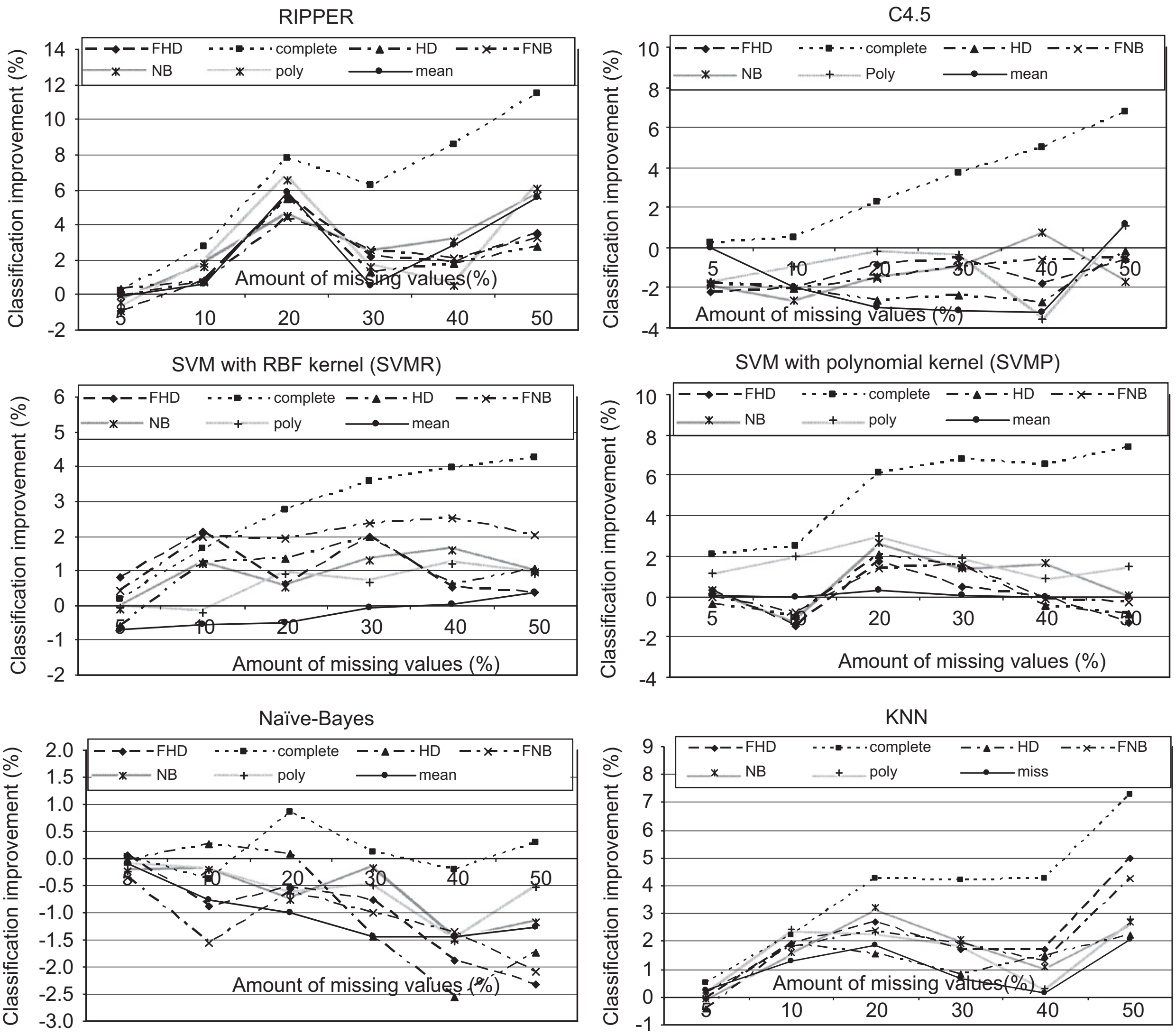

Fig. 7. Classification improvement for the RIPPER (top left), C4.5 (top right), SVM with RBF kernel (middle left), SVM with polynomial kernel (middle right), Naïve-Bayes (bottom left), and KNN (bottom right) classifiers.

For the RIPPER classifier, all the imputation methods improve its classification accuracy for data with more than 10% missing values. Here the application of NB imputation results on average in the improvement of classification performance of up to 6% for 20% and 50% of missing data. In this case, the mean imputation is characterized by similar performance as other imputation methods. Also, for the low amounts of missing values, i.e., 5%, all imputation methods perform similarly.

The C4.5 classifier can handle missing data on its own. The only improvements due to the imputation are observed for larger amounts of missing data, i.e., 50%, in which case Poly and mean imputations provide superior results when compared with the other imputation methods included in this study.

The SVM classifier with RBF kernel (SVMR) is shown to benefit from the imputation since its performance with missing data is, on average, the worst. This is true for all considered imputation methods, except for mean imputation, in which case the improvements are achieved for datasets with at least 40% of missing data. The best improvements are achieved with data imputed by FNB, and the differences are as large as 2–3% across the entire spectrum of the amounts of missing values. Therefore, imputation of missing values while working with SVMR is recommended.

The accuracy of SVM classifier with polynomial kernel (SVMP) is also improved due to using imputation methods. In particular, the polytomous regression based imputation provides superior results across virtually all amounts of missing values. This method, on average, improved the classification accuracy by 2% when compared to using missing data.

We also note that the SVMP's average error rate on complete data is the lowest among the classifiers included in this study at 35.5%, which suggests that this classifier provides high quality classifications. Most importantly, when working with missing data SVMP's classifications can be further improved (when compared to results on missing data) by using the imputation methods. This shows that using imputation and classification in tandem gives the best results.

Naïve-Bayes classifier is shown to be missing data resistant across different amounts of missing values, i.e. classifications on missing data are better than those which use imputed data.

Finally, KNN classifier is shown to be the most susceptible to missing data. Its performance with data that contain missing values is always worse than the performance when an imputation method is used. The highest classification improvements, on average, are achieved with data imputed by both framework based methods, and the average improvement in classification accuracy between the best

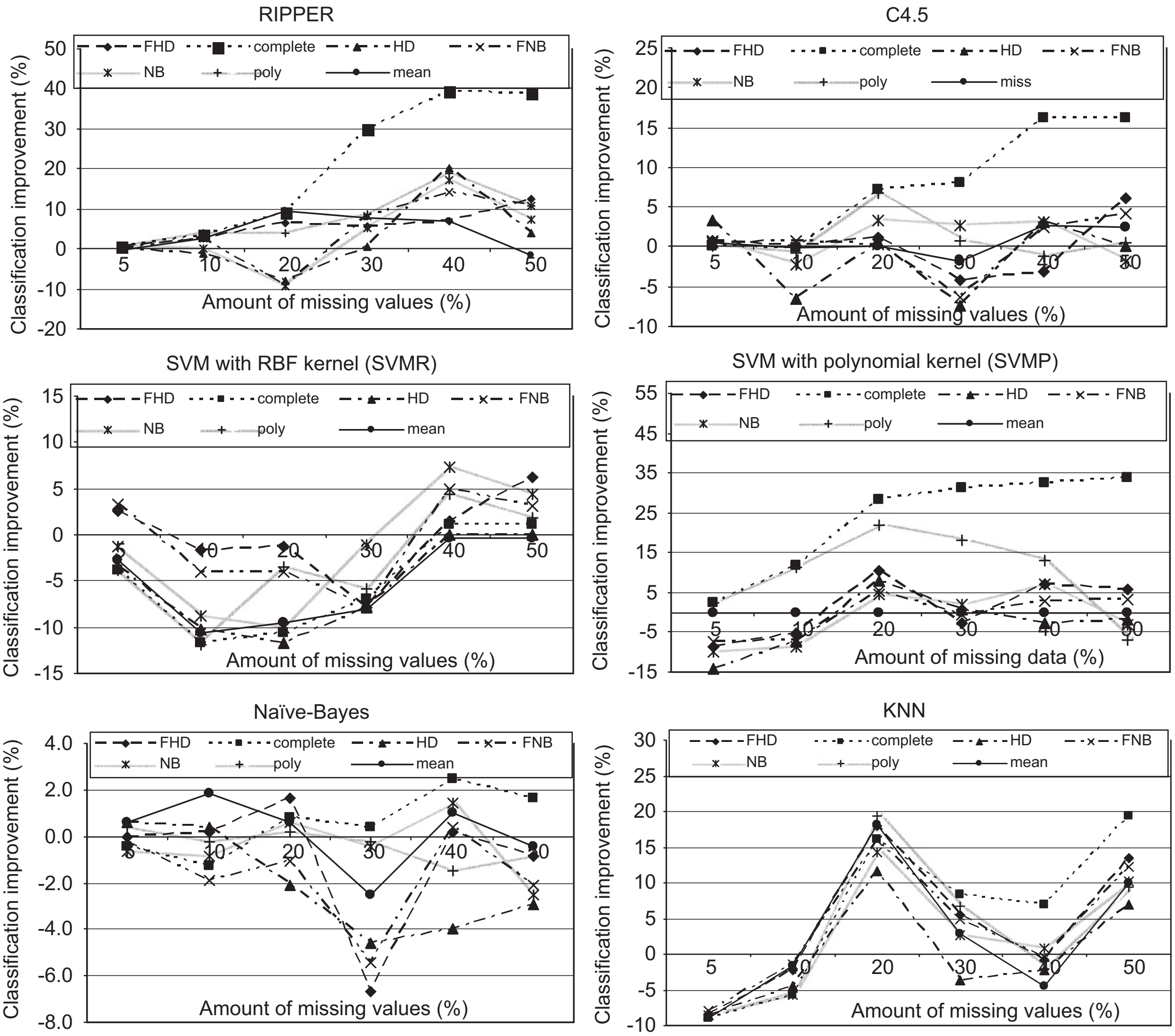

Fig. 8. Classification improvement on the Tic dataset for the RIPPER (top left), C4.5 (top right), SVM with RBF kernel (middle left), SVM with polynomial kernel (middle right), Naïve-Bayes (bottom left), and KNN (bottom right) classifiers.

FHD method and using the un-imputed (missing) data is about 5% for datasets with 50% of the values missing.

3.5. *Analysis of results for Spl and Tic datasets*

In addition to average results reported in Section 3.3, we provide results for the considered imputation methods and the six classifiers obtained on two representative datasets, Spl and Tic. The motivation behind this selection is that they cover different sizes of the datasets (3200 samples in Spl and 960 samples in Tic) and different number of attributes and classes (61 attributes and 3 classes in Spl and 10 attributes and 2 classes in Tic). The classification error improvement (lift) for Tic and Spl datasets based on the six imputation methods as well as the complete data for all six classifiers are shown in Figs. 8 and 9, respectively. Imputation of missing values has a positive impact on the classification performance of SVMP, SVMR, and KNN for both datasets, and RIPPER for the Tic dataset. Similar to average results shown in the previous section, C4.5 and Naïve-Bayes classifiers can appropriately handle the missing values for both datasets and they show little or no improvement as a result of imputation. The results on these two datasets show that the improvements can be quite significant, up to 25% better for SVMP on the Tic dataset, although the difference is usually smaller, and in case of the Spl dataset it stays below 5%. These results demonstrate that the amount of improvement due to using imputation of missing values varies for different datasets and classifiers. In some cases, such as Tic dataset, the benefits could be significant confirming the necessity of performing the imputation. However, in case of the other datasets, such as Spl, the difference is less significant and depends on the classifier that is used. Here, imputation performed for data classified with KNN and SVMR provides improvements for high amounts of missing values, while for some other classifiers, like Naïve-Bayes, the imputation may even have a detrimental effect.

3.6. *Analysis of the results with respect to different amounts of missing data*

The statistical significance of the difference between the classification errors when an imputation method is used and when classification is performed without imputation on data with different amounts of missing values is summarized in Table 6. The significance is computed for each of the six amounts of missing data and six

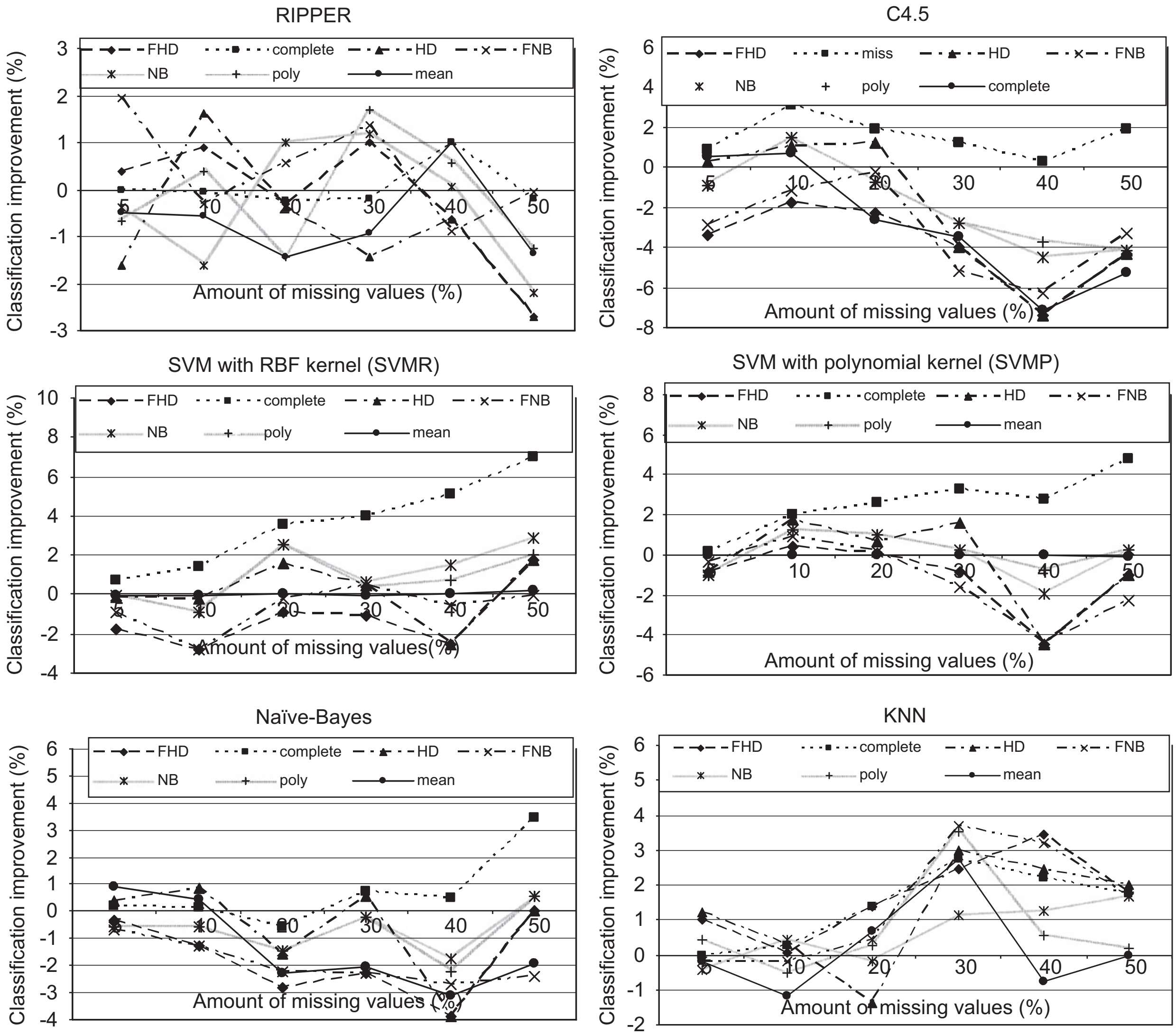

Fig. 9. Classification improvement on the Spl dataset for the RIPPER (top left), C4.5 (top right), SVM with RBF kernel (middle left), SVM with polynomial kernel (middle right), Naïve-Bayes (bottom left), and KNN (bottom right) classifiers.

Table 6
Statistical significance of the difference between using imputed and missing data for each of the considered imputation methods (columns) for different amounts of missing values (rows)

Amount of missing data	Imputation methods					
	FHD significance (*t*-value)	HD significance (*t*-value)	FNB significance (*t*-value)	NB significance (*t*-value)	Poly significance (*t*-value)	Mean significance (*t*-value)
5%	∼ (−0.4)	∼ (−1.0)	∼ (−0.7)	∼ (−1.0)	∼ (−0.3)	∼ (−0.7)
10%	∼ (−0.1)	∼ (0.2)	∼ (−0.1)	∼ (−0.3)	∼ (1.1)	∼ (−0.5)
20%	∼ (1.6)	∼ (1.0)	∼ (1.3)	∼ (1.1)	+ + (**2.3**)	∼ (0.5)
30%	∼ (1.5)	∼ (0.7)	∼ (1.7)	∼ (2.0)	+ + (**2.2**)	∼ (−0.9)
40%	∼ (−0.3)	∼ (−0.9)	∼ (−0.4)	∼ (1.9)	∼ (−0.8)	∼ (−0.3)
50%	∼ (0.4)	∼ (0.5)	∼ (0.9)	∼ (0.8)	+ + (**3.5**)	∼ (1.4)

"++" indicates that using a given imputation method gives statistically significantly better classification errors for a given amount of missing data, "−" indicates that it gives statistically significantly worse classification errors, "∼" indicates that the difference in the errors is insignificant; positive *t*-value means that classification errors for the imputed data were better, while negative values means that classification errors for the data with missing values were better.

The statistically significant results are shown in bold.

imputation methods based on average classification errors across the six classifiers and the fifteen datasets.

The table shows that for 20%, 30% and 50% amounts of missing data and all imputation methods, except the mean imputation, imputation improves classification errors (the corresponding *t*-values are positive). The results also show that imputation performed when 5% and 40% of the data is missing does not, on average, improve the classification. Virtually all differences are statistically insignificant, and there are no clear trends that would indicate the relationship between the quality of the imputation and the amount of missing data. Only three statistically significant improvements in classification errors occurred for 20%, 30%, and 50% of missing data for the Poly method. At the same time, the mean imputation is shown on average to consistently provide negative *t*-values, which means that the corresponding classification accuracy was lower than using the missing data. The low statistical significance of the differences stems from using the C4.5 and Naïve-Bayes classifiers, which on their own are shown to be missing data resistant. We conclude that on average for most non-trivial (i.e., above 10%) amounts of missing values and across various classifiers, performing imputation is shown to result in improved (smaller) classification errors.

4. Summary and conclusions

The need for missing data imputation methods is stimulated by the existence of numerous industrial and research databases that contain missing values. In this paper we examine the impact of performing missing data imputation on the subsequently performed classification.

To this end, we performed a comprehensive experimental study, which includes six single and multiple imputation methods that were used to impute missing values in 15 discrete datasets. The imputed data were used to perform classification with six modern classifiers to investigate the effect of imputation on the classification errors. We also considered imputations for six different amounts of missing data (i.e., 5%, 10%, 20%, 30%, 40%, and 50%) for each of the datasets, and compared the results obtained from classification on imputed data with the ones on missing data.

Based on the above experiments, we conclude that, on average, imputation improves the subsequent classification, except for the mean imputation method which provides improvements only when used for a substantial (50%) amount of missing data. Although these results in general agree with the results presented by Grzymala-Busse and Hu [11], a substantially more comprehensive and detailed analysis is presented in this paper. Our study shows that the impact of the imputation varies between different classifiers. The major (and statistically significant) improvements were achieved for the KNN, SVMs with RBF kernels, SVM with polynomial kernels and RIPPER classifiers, while the C4.5 and Naïve-Bayes classifiers are shown to be missing data resistant. We also conclude that imputation is beneficial for most amounts of missing data above 5% and that the amount of improvement does not depend on the amount of missing data. The performed experimental study also shows that there is no universally best imputation method. Naïve-Bayes based imputation is shown to be the best for RIPPER and datasets with high, i.e., 40% and 50%, amount of missing data. The multiple imputation polytomous regression method is shown to be the best for the SVM with polynomial kernel for all amounts of missing values and for C4.5 for datasets with high (50%) amount of missing data. Also the application of the recently proposed imputation framework is shown to be best for SVM with RBF kernel and KNN. Finally, the mean imputation is shown to be the least beneficial.

References

[1] J.L. Shafer, Analysis of Incomplete Multivariate Data, Chapman and Hall, London, 1997.

[2] K. Lakshminarayan, S.A. Harp, T. Samad, Imputation of missing data in industrial databases, Appl. Intell. 11 (1999) 259–275.

[3] R.J. Little, D.B. Rubin, Statistical Analysis with Missing Data, Wiley, New York, 1987.

[4] H.L. Oh, F.L. Scheuren, Weighting adjustments for unit nonresponse, in: W.G. Madow, I. Olkin, D.B. Rubin (Eds.), Incomplete Data in Sample Survey, Theory and Bibliographies, vol. 2, Academic Press, New York, 1983, pp. 143–183.

[5] D.B. Rubin, Multiple imputation after 18+ years, J. Am. Stat. Assoc. 91 (1996) 473–489.

[6] K.J. Cios, W. Pedrycz, R. Swiniarski, Data Mining Methods for Knowledge Discovery, Kluwer Academic Publishers, Dordrecht, 1998.

[7] K. Chan, T.W. Lee, T.J. Sejnowski, Variational Bayesian learning of ICA with missing data, Neural Comput. 15 (8) (2003) 1991–2011.

[8] A. Farhangfar, L. Kurgan, W. Pedrycz, A novel framework for imputation of missing values in databases, IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans 37 (5) (2007) 692–709.

[9] A. Farhangfar, L. Kurgan, W. Pedrycz, Experimental analysis of methods for imputation of missing values in databases, in: Intelligent Computing: Theory and Applications II Conference, in conjunction with the SPIE Defense and Security Symposium (formerly AeroSense), Orlando, FL, 2004, pp. 172–182.

[10] A.J. Feelders, Handling missing data in trees: surrogate splits or statistical imputation, in: Proceedings of the 3rd European Conference on Principles and Practice of Knowledge Discovery in Data Bases (PKDD99), Springer, Berlin, 1999, pp. 329–334.

[11] J.W. Grzymala-Busse, M. Hu, A comparison of several approaches to missing attribute values in data mining, in: Proceedings of the 2nd International Conference on Rough Sets and Current Trends in Computing 2000 (RSCTC'2000), Banff, Canada, 2000, pp. 340–347.

[12] W. Zhang, Association based multiple imputation in multivariate datasets: a summary, in: Proceedings of the 16th International Conference on Data Engineering (ICDE-2000), 2000, pp. 310–311.

[13] L.A. Kurgan, K.J. Cios, M. Sontag, F.J. Accurso, Mining the cystic fibrosis data, in: J. Zurada, M. Kantardzic (Eds.), Next Generation of Data-Mining Applications, IEEE Press, New York, 2005, pp. 415–444.

[14] E. Acuna, C. Rodriguez, The treatment of missing values and its effect in the classifier accuracy, in: D. Banks, L. House, F.R. McMorris, P. Arabie, W. Gaul (Eds.), Classification, Clustering and Data Mining Applications, Springer, Berlin, Heidelberg, 2004, pp. 639–648.

[15] J.K. Dixon, Pattern recognition with partly missing data, IEEE Trans. Systems Man Cybern. SMC-9 10 (1979) 617–621.

[16] G.E.A.P.A. Batista, M.C. Monard, An analysis of four missing data treatment methods for supervised learning, Appl. Artif. Intell. 17 (5/6) (2003) 519–533.

[17] J.R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufman, Los Altos, CA, 1993.

[18] P. Clark, T. Niblett, The CN2 induction algorithm, Mach. Learn. 3 (4) (1989) 261–283.

[19] J. Grzymala-Busse, LERS—a system for learning from examples based on rough sets, in: R. Slowinski (Ed.), Intelligent Decision Support—Handbook of Applications and Advances of the Rough Sets Theory, Kluwer, Dordrecht, 1992, pp. 3–18.

[20] D.J. Mundfrom, A. Whitcomb, Imputing missing values: the effect on the accuracy of classification, Multiple Linear Regression Viewpoints 25 (1) (1998) 13–19.

[21] W. Cohen, Fast effective rule induction, in: Proceedings of the 12th International Conference on Machine Learning (ICML-05), 1995, pp. 115–123.

[22] W. Cohen, Learning trees and rules with set-valued features (postscript), in: Proceeding of the 13th National Conference on Artificial Intelligence (AAAI-96), 1996, pp. 709–716.

[23] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines, Cambridge University Press, Cambridge, 2000.

[24] V. Vapnik, The Nature of Statistical Learning Theory, Springer, Berlin, 1995.

[25] A.L. Bello, Imputation techniques in regression analysis: looking closely at their implementation, Comput. Stat. Data Anal. 20 (1995) 45–57.

[26] N. Tang, V.N. Vemuri, Web-based knowledge acquisition to impute missing values for classification, in: Proceedings of the IEEE/WIC/ACM International Joint Conference on Web Intelligence, Beijing, China, 2004, pp. 124–130.

[27] G.W. Snedecor, W.G. Cochran, Statistical Methods, eighth ed., Iowa State University Press, 1989.

[28] H.Y. Chen, R. Little, A test of missing completely at random for generalized estimating equations with missing data, Biometrika 86 (1) (1999) 1–13.

[29] G. Sande, Hot Deck Imputation Procedures, Incomplete Data in Sample Surveys, vol. 3, Academic Press, New York, 1983.

[30] R.O. Duda, P.E. Hart, Pattern Classification and Scene Analysis, Wiley, New York, 1973.

[31] D.B. Rubin, Formalizing subjective notions about the effect of nonrespondents in sample surveys, J. Am. Stat. Assoc. 72 (1977) 538–543.

[32] D.B. Rubin, Multiple imputations in sample surveys, in: Proceedings of the Survey Research Methods Section of the American Statistical Association, 1978, pp. 20–34.

[33] S. Buuren, C.G.M. Oudshoorn, Flexible Multivariate Imputation by MICE, Leiden: TNO Preventie en Gezondheid, TNO/VGZ/PG 99.045, 1999.

[34] K.-H. Li, Imputation using Markov chains, J. Stat. Comput. Simul. 30 (1988) 57–79.

[35] D.B. Rubin, J.L. Schafer, Efficiently creating multiple imputations for incomplete multivariate normal data, in: Proceedings of the Statistical Computing Section of the American Statistical Association, 1990.

[36] C. Alzola, F. Harrell, An Introduction of S-Plus and the Hmisc and Design Libraries, Available from ⟨http://lib.stat.cmu.edu/S/Harrell/doc/splusp.pdf⟩, 1999.

[37] S.V. Buuren, E.M. Mulligen, J.P.L. Brand, Routine multiple imputation in statistical databases, in: J.C. French, H. Hinterberger (Eds.), Proceedings of the Seventh International Working Conference on Scientific and Statistical Database Management, IEEE Computer Society Press, Los Alamitos, CA, 1994, pp. 74–78.

[38] G. Casella, E.L. George, Explaining the Gibbs sampler, Am. Statist. 46 (1992) 167–174.

[39] Z. Ghahramani, M.I. Jordan, Mixture models for learning from incomplete data, in: R. Greiner, T. Petsche, S.J. Hanson (Eds.), Computational Learning Theory and Natural Learning Systems, Volume IV: Making Learning Systems Practical, MIT Press, New York, 1997, pp. 67–85.

[40] D.J. Newman, S. Hettich, C.L. Blake, C.J. Merz, UCI repository of machine learning databases, Department of Information and Computer Science, University of California, Irvine, CA, 1998 ⟨http://www.ics.uci.edu/∼mlearn/MLRepository.html⟩.

[41] S. Hettich, S.D. Bay, The UCI KDD Archive, Department of Information and Computer Science, University of California, Irvine, CA, 1999 ⟨http://kdd.ics.uci.edu⟩.

[42] I.H. Witten, E. Frank, Data Mining: Practical Machine Learning Tools and Techniques, second ed., Morgan Kaufmann, San Francisco, 2005.

[43] R. Kohavi, D.H. Wolpert, Bias plus variance decomposition for zero–one loss functions, in: Proceedings of the Thirteenth International Conference on Machine Learning (ICML), Morgan Kaufmann, London, 1996, pp. 275–283.

About the Author—ALIREZA FARHANGHFAR received his B.Sc. and M.Sc. in electrical engineering from the University of Tehran in 2002 and his M.Sc. in computer engineering from the University of Alberta in 2004. He is currently a Ph.D. student at the University of Alberta, Canada. He has received several major scholarships during his studies, including NSERC PGSD and iCore graduate scholarship in Canada. His research interests are machine learning, active learning, and bioinformatics.

About the Author—LUKASZ KURGAN received his M.Sc. degree with honors (recognized by an Outstanding Graduate Student Award) in automation and robotics from the AGH University of Science and Technology, Krakow, in 1999, and his Ph.D. degree in computer science from the University of Colorado at Boulder, in 2003. He is an associate professor at the department of Electrical and Computer Engineering at the University of Alberta in Edmonton. His research interests include data mining and knowledge discovery, machine learning, and bioinformatics. He authored and co-authored several inductive machine learning and data mining algorithms and published 60+ refereed journal and conference articles. Dr. Kurgan is a member of a steering committee of the International Conference on Machine Learning and Applications, and has been a member of numerous conference program/organizing committees in the area of data mining, machine learning, computational intelligence, and bioinformatics. He currently serves as an associate editor of the Neurocomputing, Open Proteomics Journal, and Journal of Biomedical Science and Engineering, and is a member of the IEEE, ACM, and ISCB.

About the Author—JENNIFER DY is an assistant professor at the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA, since 2002. She obtained her M.S. and Ph.D. in 1997 and 2001, respectively, from the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, and her B.S. degree in 1993 from the Department of Electrical Engineering, University of the Philippines. She received an NSF Career award in 2004. She is an editorial board member for the Machine Learning journal since 2004, and publications chair for the International Conference on Machine Learning in 2004. Her research interests include machine learning, data mining, statistical pattern recognition, and computer vision.