The Journal of Systems and Software 84 (2011) 452–459

Contents lists available at ScienceDirect

The Journal of Systems and Software

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

Missing data imputation by utilizing information within incomplete instances

Shichao Zhang[a,b,*], Zhi Jin[c], Xiaofeng Zhu[d]

[a] Department of Computer Science, Zhejiang Normal University, Jinhua, China
[b] State Key Laboratory for Novel Software Technology, Nanjing University, China
[c] Key Lab of High Confidence Software Technologies, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China
[d] School of Information Technology & Electrical Engineering, University of Queensland, QLD 4072, Australia

ARTICLE INFO

Article history:
Received 23 December 2009
Received in revised form 13 October 2010
Accepted 3 November 2010
Available online 30 November 2010

Keywords:
Incomplete data analysis
Missing value
Nonparametric imputation
Iterative imputation

ABSTRACT

This paper proposes to utilize information within incomplete instances (instances with missing values) when estimating missing values. Accordingly, a simple and efficient nonparametric iterative imputation algorithm, called the *NIIA method*, is designed for iteratively imputing missing target values. The NIIA method imputes each missing value several times until the algorithm converges. In the first iteration, all the complete instances are used to estimate missing values. The information within incomplete instances is utilized since the second imputation iteration. We conduct some experiments for evaluating the efficiency, and demonstrate: (1) the utilization of information within incomplete instances is of benefit to easily capture the distribution of a dataset; and (2) the NIIA method outperforms the existing methods in accuracy, and this advantage is clearly highlighted when datasets have a high missing ratio.

1. Introduction

Missing data imputation is an actual and unavoidable issue in intelligent data analysis applications (Allison, 2001; Little and Rubin, 2002; Shum et al., 1995; Zhu et al., 2010), and various solutions for dealing with such issues have been developed in, for example, data mining (such as Gessert, 1991; Lakshminarayan et al., 1996, 1999; Pawlak, 1993; Pearson, 2005; Quinlan, 1989, 1993; Ramoni and Sebastiani, 2001) and statistics (such as Caruana, 2001; Dempster et al., 1977; Kahl et al., 2001; Wang and Rao, 2002). Typical strategies for missing data include omitting all instances with missing values from a dataset, re-weighting the complete instances, and imputing missing values. In real application, missing value imputation is a renowned strategy when compared to the others. Missing data imputation is *a procedure of estimating the missing values based on the complete instances* (instances without missing values) in a dataset. Imputation methods that are usually employed include parametric and nonparametric regression imputation methods (Allison, 2001; Little and Rubin, 2002). In the imputation strategy, missing data treatment is independent of data analysis algorithms (for example, data mining and machine learning algorithms). This allows users to select the most suitable and efficient imputation method for their data analysis applications.

From the definition, missing data imputation is based on only complete instances (instances without missing values) in a dataset when estimating plausible values for the missing values in the dataset. From some of existing imputation algorithms, the observed information within incomplete instances (instances with missing values) can also play an important role in estimating missing values. For example, the information has been applied to identifying neighbours of an instance with missing values in NN (nearest neighbour) imputation algorithms (Chen and Shao, 2001), and the class of the instance in clustering-based imputation algorithms (Zhang et al., 2007), where NN and clustering-based imputations are well-known efficient algorithms. Therefore, it is advocated here to utilize the known information within incomplete instances well when estimating missing values.

This is crucial due to the fact that there are a great many incomplete datasets in real world applications that do not have enough complete instances for estimating missing values, even if the datasets have only a low missing ratio. For example, the missing ratio in a UCI dataset, Bridge, is only 5.56% (Blake and Merz, 1998). This is a low missing ratio in real applications, because many datasets in industrial areas often reach 50% or above. However, there are 38 complete instances out of all 108 instances in the dataset with 6 class labels. The missing values were imputed with existing imputation algorithms, based on the 38 complete instances. The experimental results show that the imputation frequently generates bias due to the few complete instances, because the size of a large sample should be beyond 30 in statistics. Moreover, there are 6 classes in this dataset, and the maximal number of complete instances in a class is only 11. Therefore, it is difficult to

* Corresponding author.
E-mail addresses: zhangsc@zjnu.cn (S. Zhang), zhijin@sei.pku.edu.cn (Z. Jin), x.zhu3@uq.edu.au (X. Zhu).

0164-1212/$ – see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jss.2010.11.887

obtain a satisfactory classification accuracy based on the few complete instances, even if the most excellent classification algorithm is employed.

As an attempt to utilize the known information within incomplete instances, in this paper a simple and efficient imputation algorithm, called *NIIA* (Nonparametric Iterative Imputation) *method*, is designed for iteratively imputing missing target values. We outline the NIIA method in Section 1.1 as follows.

1.1. Our approach

The NIIA is a strategy of iteratively imputing the missing values in a dataset. It works as follows: selecting some missing values (can be all missing values in an attribute, or a certain missing value), all complete instances are used to estimate the selected missing values. The information within incomplete instances is used from the second iteration onwards. The instances imputed in this imputation iteration are treated as observed data (or complete instances) for imputing the remained missing attributes. This process repeats until all missing attributes are imputed.

This strategy can be applied on top of an imputation algorithm to meet a data analysis task. This leads to that an existing imputation algorithm can easily be extended to utilize the known information within incomplete instances. Without loss of generality, the mean/mode imputation is employed in this research. To illustrate the NIIA in a simple way, the NN imputation is used in the following example, called NIIA with NN imputation, where the NN imputation is a procedure of imputing missing values in an instance *A* with plausible values that are generated from a complete instance that is the nearest neighbour of *A*.

Example 1. In the medical analysis of a kind of disease, the breast cancer for example, some tumour data are obtained from the patients.

Patient ID	Radius	Smoothness	Perimeter	Diagnosis
1	13.5	0.09779	?	Benign
2	21.16	0.1109	94.74	Malignant
3	12.5	?	62.11	Benign
4	14.64	0.01078	97.83	Benign

There are two complete instances (ID-2 and ID-4) and two complete instances (ID-1 and ID-3) in this example. Suppose attribute "Radius" is much important than "Perimeter". Traditional NN imputation works as follows. For the missing value in ID-1, ID-4 is the nearest neighbour of ID-1 in the complete instances, and then Perimeter = 97.83 in ID-4 is used to impute the missing value in ID-1. For the missing value in ID-3, ID-4 is the nearest neighbour of ID-3 in the complete instances, and then Smoothness = 0.01078 in ID-4 is used to impute the missing value in ID-3.

Different from traditional NN imputation, the NIIA with NN imputation works as follows. For the missing value in ID-1, ID-4 is the nearest neighbour of ID-1 in the complete instances, and then Perimeter = 97.83 in ID-4 is used to impute the missing value in ID-1. For the missing value in ID-3, there are three complete instances: ID-1, ID-2 and ID-4, and ID-1 is the nearest neighbour of ID-3 in the complete instances because attribute "Radius" is much important than "Perimeter", and then Smoothness = 0.09779 in ID-1 is used to impute the missing value in ID-3. That is, the known information within ID-1 is utilized with our approach. This will be formulated in Section 3.

We conduct some experiments to illustrate the efficiency. And the experimental results (see Section 4) show: (1) the information within incomplete instances is of benefit to capture the distribution of a dataset much better and easier than parametric imputation. (2) The NIIA method outperforms the existing methods at the accuracy, and this advantage is clearly highlighted when datasets have a high missing ratio.

1.2. Organization

In the remaining parts of this paper, related work is recalled in Section 2. Section 3 presents the NIIA method that uses a kernel-based approach. The efficiency of the proposed method is illustrated with various kinds of experiments in Section 4. Finally, the work is concluded with recommendations for future work in Section 5.

2. Related work

Little and Rubin (1987) classified missing data mechanisms into three categories as follows. (1) Missing Completely at Random (MCAR): cases with complete data are indistinguishable from cases with incomplete data. (2) Missing at Random (MAR): cases with incomplete data differ from cases with complete data, but the pattern of data missingness is traceable or predictable from other variables in the database rather than being due to the specific variable on which the data are missing. (3) Nonignorable: the pattern of data missingness is non-random and it is not predictable from other variables in the database.

For description purposes, this section recalls the main related techniques for missing data imputation.

2.1. Imputation based on complete instances

Statistical analysis with missing data has been noted in the literature for more than 70 years. Walks (1932) initiated a study on the maximum likelihood estimation for multivariate normal models with fragmentary data. Thereafter, extensive discussions on this topic continue. A useful reference for general parametric statistical inferences with missing data can be found in Little and Rubin (2002).

In data mining and machine learning, Magnani (2004) has recently reviewed the main missing data techniques (MDTs), including conventional methods, global imputation, local imputation, parameter estimation and direct management of missing data. They tried to highlight the advantages and disadvantages for all kinds of missing data mechanisms. For example, they revealed that statistical methods have been mainly developed to manage survey data and proved to be very effective in many situations. However, the main problem of these techniques is the need or strong model assumptions.

Batista and Monard (2003) have analyzed the performance of 10-NNI as an imputation method, comparing its performance with three other missing data methods: mean or mode imputation, C4.5, and CN2. This work proposed the advantages of the method are that it can predict both qualitative attributes and quantitative attributes, and it does not create explicit modes (like a decision tree or a rules), because it is a lazy model. The experiment shows that the method provides very good results, better than that of the other three methods, even for a large amount of missing data. However, the main drawback is that the algorithm searches through all the data sets are limited in large databases only based on MACR. Different imputations for industrial databases have also been studied in Lakshminarayan et al. (1996).

Yuan (2001) has also reviewed three methods of multiple imputations for missing data, including the regress method, the propensity score method and the MCMC method. Then, a standard statistical method was used to evaluate the efficiency of the multiple imputations. (In this paper, the MSE of these statistical variables were adopted by mixing them to evaluate the efficiency of the imputed results all round.)

Allison (2001) has evaluated two algorithms for producing multiple imputations or missing data by using simulated data regarding the software tool, SOLAS. Software using a propensity score classifier with the approximate Bayesian boostrap produces badly biased estimates of regression coefficients when data on predictor variables are MAR or MACR. Allison has also showed that listwise deletion produces unbiased regression estimates whenever the missing data mechanism depends only on the predictor variable, not on the response variable.

Other missing data imputation methods include a new family of reconstruction problems for multiple images from minimal data (Kahl et al., 2001), a method for handling inapplicable and unknown missing data (Gessert, 1991), different substitution methods for replacement of missing data values (Pesonen et al., 1998), robust Bayesian estimator (Ramoni and Sebastiani, 2001), and nonparametric kernel classification rules derived from incomplete (missing) data (Pawlak, 1993).

While various imputation techniques have been developed with great successes on dealing with missing values in datasets, a new research direction, the *parimputation strategy*, has recently been proposed in Zhang (2008). It advocates that a missing datum is imputed if and only if there are some complete instances in a small neighbourhood of the missing datum, otherwise it should not be imputed.

2.2. Imputation by utilizing information within incomplete instances

To the best of one's knowledge, the most relevant work should be the Nearest Neighbour method (referred to NN, for example, in Chen and Shao (2001) and CMI (Clustering-based Missing value Imputation, for example, in Zhang et al. (2007).

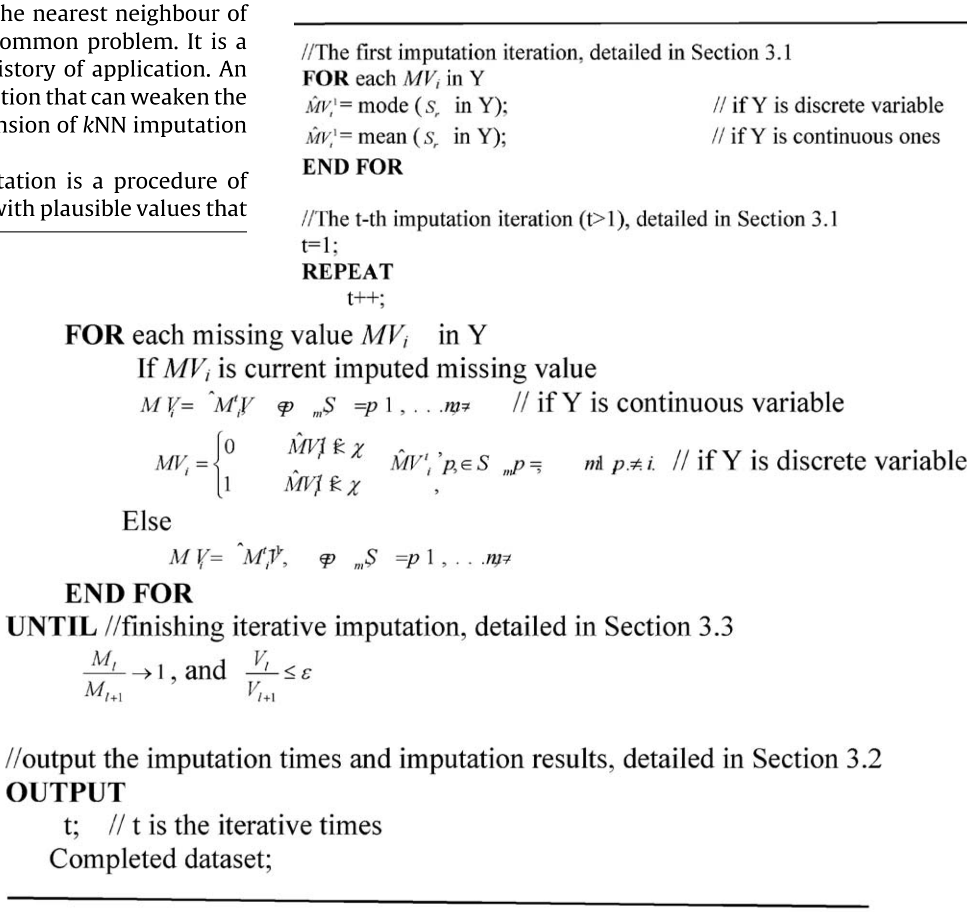
NN (nearest-neighbour) imputation is a procedure of imputing missing values of an instance *A* with plausible values that are generated from a complete instance that is the nearest neighbour of *A*, providing a practical solution to the common problem. It is a nonparametric method and has a long history of application. An extension of NN imputation is *k*NN imputation that can weaken the issue of over fitting. Recently, a new extension of *k*NN imputation is SN imputation (Zhang, 2010).

Clustering-based missing value imputation is a procedure of imputing missing values of an instance *A* with plausible values that are generated from the complete instances that are most similar to *A*. It first divides instances (including incomplete instances) in the dataset into clusters. Next, missing values of the instance *A* are patched up with the plausible values generated only from *A*'s cluster.

From the above, in *k*NN and CMI imputation methods, the information within incomplete instances is utilized only for identifying the nearest-neighbour, or the cluster of an instance with missing values. Different from these algorithms, this paper advocates utilizing the information well within incomplete instances when estimating missing values. As an attempt, an NIIA method is designed for iteratively imputing missing target values. The information within incomplete instances is used from the second iteration onwards.

3. NIIA algorithm

This section presents the NIIA method, in which the information within incomplete instances is utilized when estimating missing values. This information assists in capturing the distribution of a dataset. In the first imputation iteration of the NIIA method, a certain existing method is employed, which fits for statistical proof (such as mean/mode method), to estimate plausible values for all missing values in a dataset. From the second imputation iteration onwards, imputation is based on all instances in the dataset, where missing values in incomplete instances have been replaced with the plausible values estimated in the last iteration.



```
//The first imputation iteration, detailed in Section 3.1
FOR each $MV_i$ in Y
    $\hat{MV}_i^1$ = mode ( $s_r$  in Y);            // if Y is discrete variable
    $\hat{MV}_i^1$ = mean ( $s_r$  in Y);            // if Y is continuous ones
END FOR

//The t-th imputation iteration (t>1), detailed in Section 3.1
t=1;
REPEAT
        t++;
    FOR each missing value $MV_i$   in Y
        If $MV_i$ is current imputed missing value
            $M\,V_i = \hat{}M_i^tV \;\; \varphi \;\; {}_mS \;\; =p\,1\,,\,.\,.\,.m\neq$     // if Y is continuous variable
            $MV_i = \begin{cases} 0 & \hat{M}V_i^t \not\in \chi \\ 1 & \hat{M}V_i^t \not\in \chi \end{cases} \;\; \hat{M}V'_i\,'p,\in S \;\; {}_mp = \;\; m\,p\neq i.$   // if Y is discrete variable
        Else
            $M\,V_i = \hat{}M_i^tV, \;\; \varphi \;\; {}_mS \;\; =p\,1\,,\,.\,.\,.m\neq$
    END FOR
UNTIL //finishing iterative imputation, detailed in Section 3.3
    $\frac{M_l}{M_{l+1}} \to 1$, and $\frac{V_l}{V_{l+1}} \le \varepsilon$

//output the imputation times and imputation results, detailed in Section 3.2
OUTPUT
    t;   // t is the iterative times
    Completed dataset;
```

The Pseudo-code of NIIA Algorithm

Generally, the missing value is denoted as MVi, $i = 1, \ldots, n$ (n is the number of missing values) corresponding to imputed missing values denoted as $\hat{MV}^j$, $i = 1, \ldots, n$, $j = 1, \ldots, t$ (j is the imputation time), all missing values MVi are imputed as $\hat{MV}^1$ with the first imputation. From the second imputation onwards, the observed information will include $\hat{MV}^{j-1}$, $i = 1, \ldots, k-1, k+1, n$, $j = 2, \ldots, t-1$ while wanting to impute a missing value $\hat{MV}^j_k$, $k \neq i$, $j = 2, \ldots, t$, the imputation process will continue until algorithms reach the approximate convergence. Meanwhile, from the second imputation onwards, the kernel regression method is employed for imputing missing values under the nonparametric model. The pseudo codes of the algorithm NIIA is presented as follows:

3.1. *First imputation iteration*

Many existing methods can be used to fill in missing values in the first imputation, including any single imputation methods, such as the C4.5 algorithm and the *k*NN algorithm. Zhang et al. (2007) compute the mean (or the mode if the attribute is discrete) to impute missing values in the first instance. They think that the method is an accepted and feasible imputation method in data mining and statistics. Meanwhile, they also believe that to impute with the mean (or mode) is valid if and only if the dataset is chosen from a population with a normal distribution. However, in real world applications, one cannot know the real distribution of the dataset in advance. Therefore, running the extra iteration imputations to improve imputation performance is reasonable based on the first imputation for dealing with the missing values. Caruana (2001) thinks the first step, which imputes each missing value with the mean/mode values calculated from cases that are not missing that value, will cause cases missing many values to appear to be artificially close to each other. Hence, the author proposes a new method for avoiding this case. Moreover, the paper demonstrates this subtlety is not critical for the proper behaviour of the method, but does speed convergence on datasets that have many missing values. However, the method used by Caruana (2001) is designed to impute missing attribute values rather than missing target values. In this paper, the mean/mode method is employed to impute missing values in the first imputation iteration.

3.2. *Successive imputation iterations*

As mentioned before, nonparametric techniques will be utilized to impute missing values while there is no prior knowledge for the current dataset. Many methods exist in the kernel methods, such as the deterministic kernel method in Wang and Rao (2002), and the random kernel imputation method in Qin et al. (2009) and Zhang et al. (2007).

Assuming multi-dimension vectors without missing values are denoted as X_i and one dimension vector Y_i with missing values. Let S_r and S_m denote the sets of observed and missing values respectively, n is the number of instances in the dataset, $r = \sum_{i=1}^{n} \delta_i$, $m = n - r(\delta_i = 0$ if Y_i is missing, otherwise $\delta_i = 1)$. Let $\hat{Y}_i$, $i \in S_m$ be the imputed values and can be imputed with kernel imputation methods in Zhang et al. (2007, 2008) as:

$$\hat{Y}_i = \hat{m}_n(X_i) + \varepsilon^*, i \in S_m \tag{1}$$

where $\{\varepsilon^*\}$ is a simple random sample of size m with replacement from $\{Y_j - \hat{m}_n(X_i), j \in S_r\}$, and

$$\hat{m}_n(x) = \frac{\sum_{i=1}^{n} \delta_i Y_i K((x - X_i)/h)}{\sum_{i=1}^{n} \delta_i K((x - X_i)/h) + n^{-2}}$$

where $\hat{m}_n(x)$ is based on the completely observed pairs (X_i, Y_i). Where h is a bandwidth sequence that decreases toward 0 as the sample size n increases toward ∞; the term n^{-2} is introduced to avoid the denominator to zero. $K((x - X_i)/h)$ is a symmetric probability density function and claimed kernel function. There are some widely used kernel functions in nonparametric inference, i.e. Gaussian kernel (standard normal density function) and uniform kernel. In practice, there is no significant difference using these kernel functions. In the algorithm NIIA, the Gaussian kernel is used in the experiments.

In the NIIA algorithm, the deterministic kernel method is revised in Wang and Rao (2002) rather than the random kernel imputation methods in Qin et al. (2009) and Zhang et al. (2007) as the variation of the value of ε^* is difficult to be controlled in the iterative method. Moreover, the iterative method can reach the aim for setting ε^* which is designed to avoid large variations of the imputation values. Hence, the *t*th imputation values of the *i*th missing value is defined as $\hat{Y}_i^t$:

$$\hat{Y}_i^t = \hat{m}_t(X_i) \tag{2}$$

where t is the number of iterative imputation, $\hat{m}_t(x)$ denotes kernel estimator for $m_t(x)$ based on the completely observed pairs (X^t, Y^t):

$$\hat{m}_t(x) = \frac{n^{-1} \sum_{i=1}^{n} \delta_i Y_i^t K((x^t - X^t)/h)}{n^{-1} \sum_{i=1}^{n} \delta_i K((x^t - X^t)/h) + n^{-2}}$$

$$\text{where } Y_i^t = \begin{cases} Y_i, & \text{if } \delta_i = 0 \text{ or } i = 1, \ldots, r \\ \hat{Y}_i^{t-1}, & \text{if } \delta_i = 1 \text{ or } i = r+1, \ldots, n \end{cases}$$

In particular, $\hat{Y}_i^1 = (1/r) \sum_{i=1}^{r} Y_i$, comes from the result of the first imputation. The selection of kernel function $K((x^t - X^t)/h)$ is the same kernel function in the deterministic kernel method. Hence, in the algorithm NIIA, since the second imputation, Eq. (2) is used to impute missing target values for continuous missing target attributes until the algorithm converges.

In fact, the imputed values based on Eq. (2) are always continuous values, and the NIIA algorithm can also impute a discrete missing target attribute which is presented in pseudo of the NIIA algorithm. In this paper, the case is considered with two classes and the reader can extend our method to the case with multiple classes. In the NIIA algorithm, instances are defined as belonging to class 0 if $\hat{MV}_i^t < \chi$, and class 1 otherwise. The actual value of the class for each incomplete instance x_i is denoted by MC_{x_i}. The new class assignment based on the imputed class is denoted by $\hat{MC}^t$ in *t*th imputation to stress the dependence of the classification. More specifically, the imputed value $\hat{m}_t(x) \in R$ is transformed into a (binary) class $MC^t \in \{0, 1\} \forall x_i \in D$ based on the rule specified in the NIIA algorithm. χ is specified by the user of the technique and in many applications is set so that $|\{x_i | MC^t = 1\}| = |\{x_i | \hat{MC}^t = 1\}|$ (i.e., the number of class 1 instances before and after the application of this technique is the same). This rule for class assignment is the most natural choice and is the one primarily considered in the case studies, although the user of the technique may explore different choices near this preferred cutoff point. Based on this rule, the proposed algorithm NIIA can also be used to impute discrete missing target values.

Finally, an output of the final imputation result can be made after the algorithm has converged. Note that the imputation times is $(t+1)$ rather than $(t+2)$ times even if the iterative procedure is performed $(t+1)$ times and the first iteration will be added. This is because the last imputation does not generate imputation results and only judges the fact of whether the imputation reaches convergence.

3.3. Algorithm convergence and complexity

An important practical issue concerning the iterative imputation method is to determine at which point additional iterations have no meaningful effect on imputed values, i.e., how to judge the convergence of the algorithm. Literatures (Caruana, 2001; Qin et al., 2009) conclude that the average distance that missing attribute values move in successive iterations drops to zero, that no missing values have changed and that the method has converged in a nonparametric model. Here, a strategy is outlined for the stopping criterion for the algorithms. With t imputation times, assuming mean and variance of three successive imputations are M_l, M_{l+1}, M_{l+2}, and V_l, V_{l+1}, V_{l+2}, $(1 < l < t-2)$ respectively. If

$$\frac{M_l}{M_{l+1}} \to 1 \quad \text{and} \; \frac{V_l}{V_{l+1}} \le \varepsilon$$

That can be inferred that there is little change in imputations between the last and the former imputation, and the algorithm can be stopped for imputing without substantial impact on the resulting inferences. Different from the converged condition in existing algorithms, the stopping strategy is summarized using terminology such as 'satisfying a convergence diagnostic' rather than 'achieving convergence' to clarify that convergence is an elusive concept with iterative imputation.

While the complexity of the kernel method is $O(mn^2)$, where n is the number of instances of the dataset, m is the number of attributes, therefore the algorithm complexity of both NIIA and SIIA is $O(kmn^2)$ (k is the number of iteration imputation).

4. Experiments

In order to show the efficiency of the proposed methods, extensive experiments were done on real datasets with VC++ programming by using a DELL Workstation PWS650 with 2G main memory, 2.6G CPU, and WINDOWS 2000. For saving space, we report some of them in this section. The performance of the NIIA was compared with the existing iterative method *k*NN (Qin et al., 2009) as well as the single imputation methods for imputing continuous missing target attributes in terms of imputation accuracy with RMSE in Section 4.1, and the performance of the NIIA algorithm is presented with the existing methods for imputing discrete missing target attributes in terms of classification accuracy in a real dataset in Section 4.2. We analyze and summarize our experimental results in Section 4.3.

4.1. Experimental study on continuous missing target attribute

At first, different algorithms were designed to impute target missing values, such as the proposed algorithm NIIA, the *k*NN algorithm (Zhang, 2008, 2010), and the two single imputation methods (deterministic method for single imputation (Zhang et al., 2007)), DS for shorted, random method for single imputation (Qin et al., 2009; Zhang et al., 2007), RS for shorted. The RMSE was used to assess the predictive ability after the algorithm has converged for iterative imputation methods or the missing values are imputed for single imputation methods:

$$\text{RMSE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(e_i - \tilde{e}_i)^2}$$

where e_i is the original attribute value; $\tilde{e}_i$ is the estimated attribute value, and m is the total number of predictions. The larger the value of the RMSE, the less accurate is the prediction.

Two datasets from UCI (Blake and Merz, 1998), Housing and Auto-mpg were used in the experiment. Housing contains 506 instances and 10 continuous attributes. Auto-mpg contains 398 instances and 8 attributes. The two datasets have no missing values, and those datasets that originally come with missing values were not intentionally selected, because if they contain missing values, one could not know the real values for the missing values. Thus, the complete datasets and missing data were adopted at random to systematically study the performance of the proposed method; the percentage of missing values (missing ratio for short) was fixed at 10%, 20%, and 40%, respectively for each dataset.

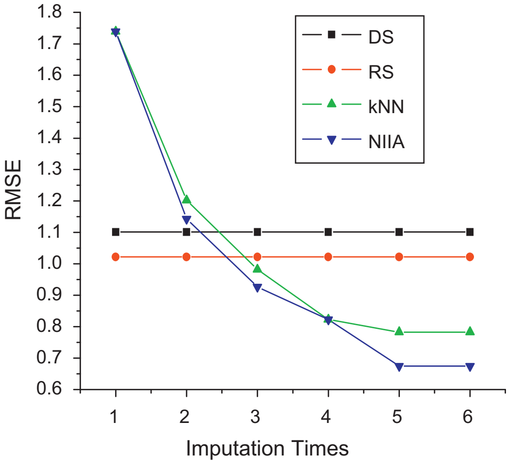


Fig. 1. The RMSE in dataset Housing with missing ratio at 10%.

Figs. 1, 3, 5 and 2, 4, 6 present the values of the RMSE in dataset Housing and Auto-mpg with missing ratios at 10%, 20%, and 40%, respectively. The experimental results show:

The NIIA algorithm can converge in all cases in the experiments. For example, in dataset Housing, the NIIA algorithm converges with the imputation times 5, 8 and 9 at different missing ratios at 10%, 20%, and 40%, respectively. Corresponding to the dataset Auto-mpg, the number is 6, 8 and 10. The higher the missing ratio, the more imputation time there is. It is obvious as more missing values must be imputed and the imputation performance will be improved slower than the case with a lower missing ratio. The results confirm that the proposed iterative imputation method is effective.

By comparing iterative imputation methods (such as the *k*NN and NIIA) with single imputation methods (such as the DS and RS), in the first imputation, iterative imputation methods show lower imputation performance than single imputation methods. It is a fact that iterative imputation methods employ the most prevalent method (i.e., mean/mode method) to impute missing values. However, since then, the situation varies, that is, the iterative algorithms outperform single methods. This implies the following facts. Firstly, it is reasonable to employ the kernel regression imputation method to impute iterative missing values rather than mean/mode which is the simplest imputation method from the second imputation onwards. As to selecting the mean/mode method as the first imputation method, this is done because the method is simple and presents low computation complexity, and the most highlighted benefit in the iterative imputation method is that the successive imputation can incrementally improve imputation performance. For instance, in the former imputation times after the first imputation, the values of the RMSE in the NIIA or *k*NN algorithm are worse than the DS or RS algorithm; conversely, the values of the RMSE in the NIIA or *k*NN algorithm are a little better than the DS and RS algorithm. However, the NIIA or *k*NN algorithm is better than the DS or RS algorithm while the algorithm nearly converges. This demonstrates that iterative imputation methods can improve imputation performance by successive imputation. Secondly, experimental results from Figs. 1–6 show that iterative

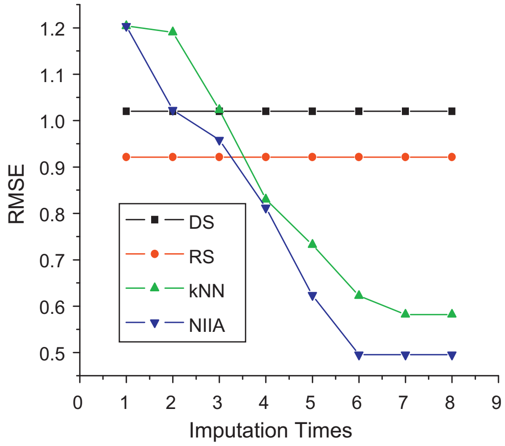


Fig. 2. The RMSE in dataset Auto-mpg with missing ratio at 10%.

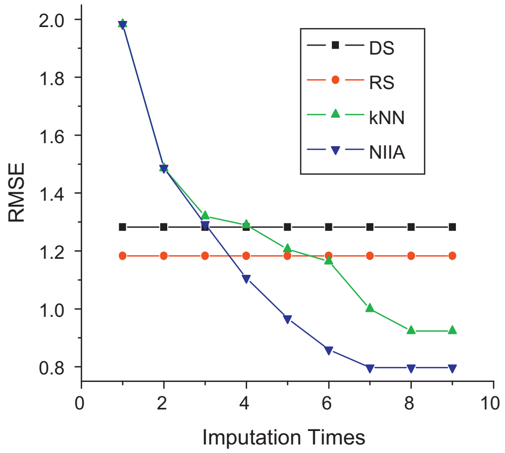


Fig. 3. The RMSE in dataset Housing with missing ratio at 20%.

imputation methods outperform than single imputation methods. Furthermore, the difference between iterative imputation methods and single ones is that the maximal values are present while the missing ratio reaches 40%. That is because the higher the missing ratio, the more obvious are the advantages of the iterative imputation methods.

When comparing the *k*NN algorithm with the NIIA method, the NIIA is better than the *k*NN method in Kahl et al. (2001) in terms of the values of the RMSE or the imputation times after the algorithm converges. It was found that they contain the same performance in the first imputation, because the two methods employ the mean/mode method for the first imputation. From the second imputation onwards, the *k*NN algorithm presents a little profit than the NIIA, for example, in the dataset Housing, the values of the RMSE of *k*NN algorithm is 1.486 at the missing ratio 20% in the second imputation, and the corresponding value in the NIIA is 1.487. In the left iteration, the results of the NIIA algorithm are better than the *k*NN, particularly in the high missing ratio, such as at the missing ratio 40%.

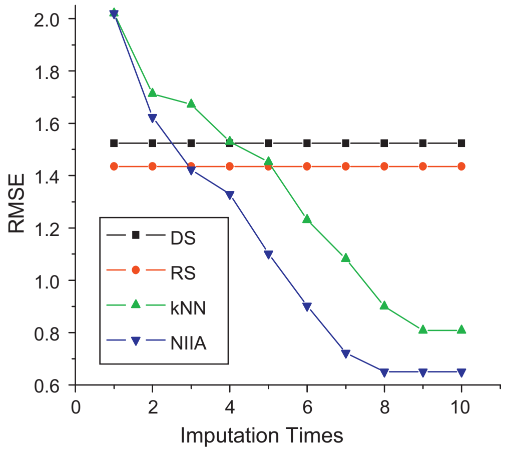


Fig. 4. The RMSE in dataset Auto-mpg with missing ratio at 20%.

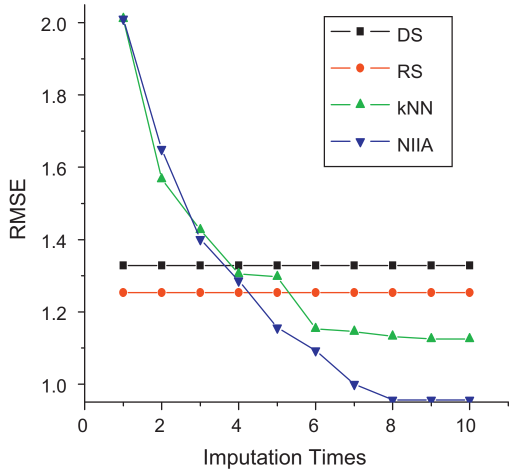


Fig. 5. The RMSE in dataset Housing with missing ratio at 40%.

4.2. Experimental study on discrete missing target attribute

The UCI datasets 'Abalone', 'Vowel', and 'CMC', in which the class attribute is discrete, are applied to compare the performances in terms of classification accuracy of the above four methods.

It is necessary to assess the performance of these prediction procedures. Their Classification Accuracy (CA) is evaluated, which is defined as:

$$\mathrm{CA} = \frac{1}{n}\sum_{i=1}^{n} l(IC_i, RC_i)$$

where t is the number of missing values, n is the number of instances in the dataset. The indicator function $l(x, y) = 1$ if $x = y$;

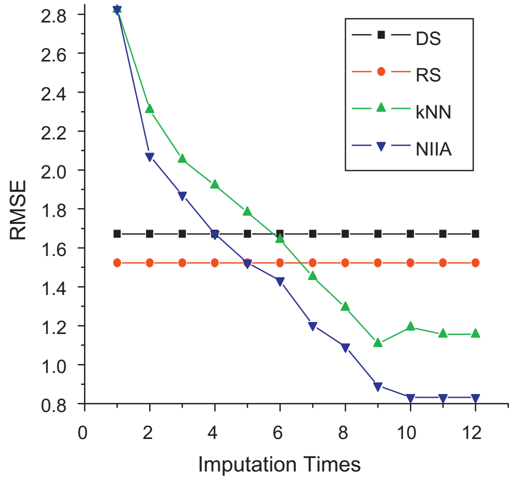


Fig. 6. The RMSE in dataset Auto-mpg with missing ratio at 40%.

Table 1
Classification accuracy: iterative algorithms reach convergence vs. single imputation algorithms.

	Abalone			Vowel			CMC		
	10%	20%	40%	10%	20%	40%	10%	20%	40%
NIIA	0.781	0.775	0.711	0.855	0.83	0.807	0.873	0.843	0.827
*k*NN	0.773	0.742	0.672	0.842	0.801	0.773	0.859	0.821	0.792
RS	0.755	0.732	0.643	0.818	0.788	0.73	0.838	0.798	0.739
DS	0.716	0.695	0.639	0.813	0.771	0.726	0.825	0.775	0.717

Table 2
Imputation times after the algorithms converge.

	Abalone			Vowel			CMC		
	10%	20%	40%	10%	20%	40%	10%	20%	40%
NIIA	6	8	10	8	10	13	8	9	12
*k*NN	7	10	12	8	11	17	8	10	15

otherwise it is 0. IC_i and RC_i are the imputation and real class label for the *i*th missing value respectively. Obviously, the larger value of the CA, the more efficient is the algorithm.

Table 1 shows the results of the classification accuracy after the iterative imputation algorithms (such as the *k*NN and NIIA) reach convergence or the result for single imputation methods (i.e., DS and RS) at the different missing ratios 10%, 20%, and 40%, respectively. Table 2 presents the results of iterative times after these two iterative algorithms have been terminated at the different missing ratios 10%, 20%, and 40%, respectively on datasets 'Abalone', 'Vowel', and 'CMC'. Due to lack of space, the details are not presented with figures, as in Section 4.1. However, similar to the results in Section 4.1, the classification accuracy of iterative algorithms for imputing discrete missing values are better than the results of single imputation in different missing ratios, particularly in the case with a high missing ratio. For example, the minimal difference between iterative algorithms and single algorithms is 0.029 (*k*NN vs. RS in Abalone), 0.043 (*k*NN vs. RS in Vowel), 0.033 (*k*NN vs. RS in CMC), and the maximum difference is 0.042 (NIIA vs. DS in abalone), 0.081 (NIIA vs. DS in Vowel), and 0.11 (NIIA vs. DS in CMC) respectively, while the missing ratio is 40%. However, the corresponding minimal values only are 0.018, 0.024, and 0.021 at the missing ratio 10%, 0.01, 0.011, and 0.023 at the missing ratio 20% respectively. Moreover, the corresponding maximum values are 0.065, 0.042, and 0.048 at the missing ratio 10%, 0.08, 0.059, and 0.068 at the missing ratio 20%. Table 2 shows that the two iterative algorithms (such as the *k*NN and NIIA) can reach convergence with the similar imputation times while the missing ratio is low, such as 10% or 20%. However, while the missing ratio is high, for example 40% in the experiments, the difference of imputation times begin to vary, that is, the *k*NN algorithm converges slower than the NIIA algorithm. By combining Tables 1 and 2, a conclusion is drawn that the NIIA algorithm outperforms the *k*NN algorithm on both the classification accuracy and imputation times for imputing discrete missing target values.

4.3. Analysis and summary

As stated previously, extensive experiments were done on real datasets downloaded from UCI. The experimental results show: (1) for complete datasets, we randomly remove some values and take them as missing values. The experiments demonstrate the NIIA algorithm is always more accurate than the extended one. (2) For incomplete datasets in different missing ratios, we test the NIIA algorithm and the extended method for a classification task. The classification accuracy based on the NIIA algorithm is always not bad than that based on the extended one.

Although the NIIA method worked well for the datasets in UCI, we can artificially construct such some datasets that the NIIA approach cannot perform that well.

5. Conclusion and future work

By developing missing data imputation techniques, this study advocates good utilization of the information within incomplete instances in existing imputation algorithms. This is because there are a great many incomplete datasets in real world applications that do not have enough complete instances for estimating missing values. In an attempt to utilize the information within missing data, a nonparametric iterative imputation algorithm, namely the NIIA method, has been designed for imputing iteratively missing target values when there is no priori knowledge to the distribution of a dataset. The NIIA method imputes each missing value several times until the algorithm converges. The information within incomplete instances is used from the second iteration onwards. Intensive experiments have been conducted to evaluate the proposed approach. The experimental results have demonstrated that the NIIA method outperforms the existing methods in terms of the RMSE (for continuous missing target attribute), or the classification accuracy (for a discrete missing target attribute), and the convergence times at different missing ratios in different real datasets. In particular, they have illustrated that the utilization of information within incomplete instances is of benefit to capture the distribution of a dataset much better and easier than parametric imputations. The experiments have also shown that the iterative imputation methods are much better than the existing imputation methods.

In future research, focus will be made on how to more effectively estimate, impute missing values with the semi-parametric model, and new measures for evaluating the imputation efficiency in general purpose.

Acknowledgements

This work was supported in part by the Australian Research Council (ARC) under large grant DP0985456, the Nature Science Foundation (NSF) of China under grants 90718020 and 10661003, the Research and Innovation Team for Guangxi Universities & Colleges, and the Guangxi NSF (Key) grants.

References

Allison, P., 2001. Missing Data. In: Sage University Papers Series on Quantitative Applications in the Social Sciences, 07-136. Sage, Thousand Oaks, CA.

Batista, G., Monard, M., 2003. An analysis of four missing data treatment methods for supervised learning. Appl. Artif. Intell. 17 (5–6), 519–533.

Blake, C., Merz, C., 1998. UCI Repository of Machine Learning Databases.

Caruana, R., 2001. An non-parametric EM-style algorithm for imputing missing values. In: Proceedings of the Eighth International Workshop on Artificial Intelligence and Statistics , Key West, Florida, Morgan Kaufmann.

Chen, J., Shao, J., 2001. Jackknife variance estimation for nearest-neighbor imputation. J. Am. Stat. Assoc. 96, 260–269.

Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. J. R. Stat. Soc. 39, 1–38.

Gessert, G., 1991. Handling missing data by using stored truth values. SIGMOD Rec. 20 (3), 30–42.

Kahl, F., et al., 2001. Minimal projective reconstruction including missing data. IEEE Trans. Pattern Anal. Mach. Intell. 23 (4), 418–424.

Lakshminarayan, K., et al., 1996. Imputation of missing data using machine learning techniques. KDD, 140–145.

Lakshminarayan, K., et al., 1999. Imputation of missing data in industrial databases. Appl. Intell. 11, 259–275.

Little, R.J.A., Rubin, D.A., 1987. Statistical Analysis with Missing Data. John Wiley and Sons, New York.

Little, R., Rubin, D., 2002. Statistical Analysis with Missing Data, second ed. John Wiley and Sons, New York.

Magnani, M., 2004. Techniques for Dealing with Missing Data in Knowledge Discovery Tasks, Available at: http://magnanim.web.cs.unibo.it/index.html.

Pawlak, M., 1993. Kernel classification rules from missing data. IEEE Trans. Inf. Theory 39 (3), 979–988.

Pesonen, E., Eskelinen, M., Juhola, M., 1998. Treatment of missing data values in a neural network based decision support system for acute abdominal pain. Artif. Intell. Med. 13 (3), 139–146.

Pearson, P.K., 2005. Mining Imperfect Data: Dealing with Contamination and Incomplete Records. SIAM.

Quinlan, J., 1989. Unknown attribute values in induction. In: Proc 6th Int workshop on machine learning , Ithaca, pp. 164–168.

Quinlan, J.R., 1993. C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo, USA.

Qin, Y.S., et al., 2009. POP algorithm: kernel-based imputation to treat missing values in knowledge discovery from databases. Exp. Syst. Appl. 36, 2794–2804.

Ramoni, M., Sebastiani, P., 2001. Robust learning with missing data. Mach. Learn. 45 (2), 147–170.

Shum, H., Ikeuchi, K., Reddy, R., 1995. Principal component analysis with missing data and its application to polyhedral object modeling. IEEE Trans. Pattern Anal. Mach. Intell. 17 (9), 854–867.

Wang, Q., Rao, J.N.K., 2002. Empirical likelihood-based inference under imputation for missing response data. Ann. Stat. 30, 896–924.

Walks, S., 1932. Moments and distributions of estimates of population parameters from fragments samples. Ann. Math. Stat. 3, 163–203.

Yuan, Y.C., 2001. Multiple Imputation for Missing Data: Concepts and New Development SAS/STAT 8.2. SAS Institute Inc., NC, Cary, Available at: http://www.sas.com/statistics.

Zhang, S.C., 2008. Parimputation: from imputation and null-imputation to partially imputation. IEEE Intell. Inform. Bull. 9 (1), 32–38.

Zhang, S.C., 2010. Shell-neighbor method and its application in missing data imputation. Appl. Intell., doi:10.1007/s10489-009-0207-6.

Zhang, S.C., Jin, Z., Zhu, X.F., 2008. NIIA: nonparametric iterative imputation algorithm. In: Tenth Pacific Rim International Conference on Artificial Intelligence (PRICAI-08) , pp. 544–555.

Zhang, S.C., et al., 2007. Missing value imputation based on data clustering. Trans. Comp. Sci. 2, 128–138.

Zhu, X.F., et al., 2010. Missing value estimation for mixed-attribute datasets. IEEE Trans. Knowl. Data Eng., http://doi.ieeecomputersociety.org/10.1109/TKDE.2010.99.

Shichao Zhang is a “Bairen” Program Distinguished Professor and the Director of Artificial Intelligence Institute at Zhejiang Normal University, China. He holds a PhD degree from the CIAE, China. His research interests include machine learning and information quality. He has published about 60 international journal papers and over 60 international conference papers. He is a CI for 11 competitive nation-level grants, including China NSF, China 863 Program, China 973 Program, and Australia large ARC. He is a senior member of the IEEE, a member of the ACM; and served/ing as an associate editor for IEEE Transactions on Knowledge and Data Engineering, Knowledge and Information Systems, and IEEE Intelligent Informatics Bulletin.

Zhi Jin is a professor of Computer Science at Peking University, Beijing, China. Before joined Peking University, she was a professor of Academy of Mathematics and System Science at China Academy of Sciences since 2001. She received the MS degree in computer science in 1987 and the PhD degree in 1992, both from Changsha Institute of Technology, China. Her research interests include software requirements engineering and knowledge engineering. She has published a co-authored monograph by Kluwer Academic Publishers and more than 50 referred journal/conference papers in these areas. She has won various nation-class awards/honors in China, mainly including the Natural Science Foundation for Distinguished Young Scholars of China (2006), the Award for Distinguished Women IT Researcher of China (2004), and the Zhongchuang Software Talent Award (1997). She is the leader of over 10 national competitive grants, including 3 China NSF grants, 2 China 973 program grant and 2 China 863 program grants. She is a senior member of the IEEE, a standing senior member of the China Computer Federation (CCF), a grant review panelist for China NSF (Information Science Division); serving as an executive editor-in-chief for Journal of Software, an editorial board member for Expert Systems and Chinese Journal of Computers; and served as a PC co-chair, area chair, or PC member for various conferences.

Xiaofeng Zhu is currently a PhD student at the University of Queensland, Australia. His research interests include data mining, machine learning and multimedia. He has published about 20 papers in international journals/conferences.