Missing Value Estimation for Mixed-Attribute Data Sets

Xiaofeng Zhu, Shichao Zhang, *Senior Member*, *IEEE*,
Zhi Jin, *Senior Member*, *IEEE*, Zili Zhang, and Zhuoming Xu

Abstract—Missing data imputation is a key issue in learning from incomplete data. Various techniques have been developed with great successes on dealing with missing values in data sets with homogeneous attributes (their independent attributes are all either continuous or discrete). This paper studies a new setting of missing data imputation, i.e., imputing missing data in data sets with heterogeneous attributes (their independent attributes are of different types), referred to as imputing mixed-attribute data sets. Although many real applications are in this setting, there is no estimator designed for imputing mixed-attribute data sets. This paper first proposes two consistent estimators for discrete and continuous missing target values, respectively. And then, a mixture-kernel-based iterative estimator is advocated to impute mixed-attribute data sets. The proposed method is evaluated with extensive experiments compared with some typical algorithms, and the result demonstrates that the proposed approach is better than these existing imputation methods in terms of classification accuracy and root mean square error (RMSE) at different missing ratios.

Index Terms—Classification, data mining, methodologies, machine learning.

1 Introduction

Missing data imputation aims at providing estimations for missing values by reasoning from observed data [5]. Because missing values can result in bias that impacts on the quality of learned patterns or/and the performance of classifications, missing data imputation has been a key issue in learning from incomplete data. Various techniques have been developed with great successes on dealing with missing values in data sets with homogeneous attributes (their independent attributes are all either continuous or discrete). However, these imputation algorithms cannot be applied to many real data sets, such as equipment maintenance databases, industrial data sets, and gene databases, because these data sets are often with both continuous and discrete independent attributes [21]. These heterogeneous data sets are referred to as mixed-attribute data sets and their independent attributes are called as mixed independent attributes in this research. To meet the above practical requirement, this paper studies a new setting of missing data imputation, i.e., imputing missing data in mixed-attribute data sets.

- *X. Zhu is with the School of Information Technology and Electrical Engineering, University of Queensland, QLD 4072, Australia. E-mail: x.zhu3@ uq.edu.au.*
- *S. Zhang is with the Computer Department, Zhejiang Normal University, Jinhua 321004, China. E-mail: zhangsc@zjnu.cn.*
- *Z. Jin is with the Key Lab of High Confidence Software Technologies, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China. E-mail: zhijin@sei.pku.edu.cn.*
- *Z. Zhang is with the Faculty of Computer and Information Science, Southwest University, Chongqing 400715, China, and the School of Information Technology, Deakin University, Geelong VIC 3217, Australia. E-mail: zhangzl@swu.edu.cn, zzhang@deakin.edu.au.*
- *Z. Xu is with the College of Computer and Information, Hohai University, Nanjing 210098, China. E-mail: zmxu@hhu.edu.cn.*

Manuscript received 15 July 2009; revised 2 Jan. 2010; accepted 6 Jan. 2010; published online 9 June 2010.
Recommended for acceptance by D. Tao.
For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDE-2009-07-0552.
Digital Object Identifier no. 10.1109/TKDE.2010.99.

Imputing mixed-attribute data sets can be taken as a new problem in missing data imputation because there is no estimator designed for imputing missing data in mixed-attribute data sets. The challenging issues include, such as how to measure the relationship between instances (transactions) in a mixed-attribute data set, and how to construct hybrid estimators using the observed data in the data set. To address the issue, this research proposes a nonparametric iterative imputation method based on a mixture kernel for estimating missing values in mixed-attribute data sets. It first constructs a kernel estimator to infer the probability density for independent attributes in a mixed-attribute data set. And then, a mixture of kernel functions (a linear combination of two single kernel functions, called mixture kernel) is designed for the estimator in which the mixture kernel is used to replace the single kernel function in traditional kernel estimators. These estimators are referred to as mixture kernel estimators. Based on this, two consistent kernel estimators are constructed for discrete and continuous missing target values, respectively, for mixed-attribute data sets. Further, a mixture-kernel-based iterative estimator is proposed to utilize all the available observed information, including observed information in incomplete instances (with missing values). Finally, a grid research method is presented to obtain the optimal bandwidth for the proposed mixture kernel estimators, instead of the data-driven method in [29]. The proposed algorithm is experimentally evaluated in terms of root mean squared error (RMSE), classification accuracy and the convergence speed of the algorithm, compared with extant methods, such as the nonparametric imputation method with a single kernel, the nonparametric method for continuous attributes, and frequency estimator (FE). These experiments were conducted on UCI data sets and a real data set at different missing ratios.

The rest of the paper is organized as follows: It begins with briefly recalling related work in Section 2. The new algorithm is designed and analyzed in Section 3. The

experimental results are reported and analyzed in Section 4. Finally, Section 5 concludes the paper.

2 Related Work

Methods for dealing with missing values can be classified into three categories by following the idea from [10], [36], [25]: 1) case deletion, 2) learning without handling of missing values, and 3) missing value imputation.

The case deletion is to simply omit those cases with missing values and only to use the remaining instances to finish the learning assignments [22], [27]. The second approach is to learn without handling of missing data, such as Bayesian Networks method [23], Artificial Neural Networks method 15, the methods in [14], [17].

Different from the former two, missing data imputation method advocates filling in missing values before a learning application. Missing data imputation is a procedure that replaces the missing values with some plausible values, such as [35], [26]. While the imputation method is regarded as a more popular strategy [16], a new research direction, the parimputation strategy, has recently been proposed in [38]. It advocates that a missing datum is imputed if and only if there are some complete instances in a small neighborhood of the missing datum, otherwise, it should not be imputed.

2.1 Research into Missing Value Imputation

Commonly used methods to impute missing values include parametric and nonparametric regression imputation methods. The parametric method, such as linear regression [13], [22], and [31], is superior while the data set are adequately modeled. However, in real applications, it is often impossible to know the distribution of the data set. Therefore, the parametric estimators can lead to highly bias, and the optimal control factor settings may be miscalculated. For this case, nonparametric imputation method [28], [35], [38] can provide superior fits by capturing the structure of the data set.

However, these imputation methods are designed for either continuous or discrete independent attributes. For example, the well-established imputation methods in [4], [35] are developed for only continuous attributes. And these estimators cannot handle discrete attributes well. Some methods, such as C4.5 algorithm [28], association-rule-based method [40], and rough-set-based method [24], are designed to deal with only discrete attributes. In these algorithms, continuous attributes are always discretized before imputing. This possibly leads to a loss of useful characteristics of the continuous attributes. There are some conventional imputation approaches, such as [6], [1], and [11], designed for discrete attributes using a "frequency estimator" in which a data set is separated into several subsets or "cells." However, when the number of cells is large, observations in each cell may not be enough to nonparametrically estimate the relationship among the continuous attributes in the cell.

When facing with mixed independent attributes, some imputation methods take the discrete attributes as continuous ones, or other methods are used. Some reports, for instance, [6], [1], and [11], selected to smooth the mixed regressors, but without taking the selection of bandwidth into account. Therefore, Racine and Li [29] proposed a natural extension of the method in [3] to model the settings of discrete and continuous independent attributes in a fully nonparametric regression framework.

However, all the above methods were designed to impute missing values with only the observed values in complete instances, and did not take into account observed information in incomplete instances. On the other hand, all the above methods are designed to impute missing values one time. John et al. [18] thought that iterative approaches impute missing values several times and can be usefully developed for missing data imputation. Zhang et al. [42] thought it is necessary to iteratively impute missing values while suffering from large missing ratio. Hence, many iterative imputation methods have been developed, such as the Expectation-Maximization (EM) algorithm which is a classical parametric method. Zhang et al. [42] and Caruana [9] proposed nonparametric iterative methods but based on a k-nearest neighborhood framework. In this paper, the proposed iterative imputation method is a nonparametric model specially designed for those data sets with both continuous and discrete attributes, which is based on a kernel regression imputation framework.

2.2 Research into Bandwidth Selection and Kernel Function Selection

Kernel function is popularly used in building imputation models, such as [35], [26] and [29], denoted by kernel imputation. When kernel imputation method is employed to impute missing values, it usually consists of two parts: kernel function selection and bandwidth adjustment.

During the process for selecting kernel functions, what we need to consider is not only the ability to learn from the data (i.e., "interpolation"), but also the ability to predict unseen data (i.e., "extrapolation"). Smits and Jordan [34] argued that these two characteristics are largely determined by the choice of kernel functions. For example, a global kernel (such as the polynomial kernel) has better extrapolation abilities at lower order degrees, but requires higher order degrees for good interpolation. A local kernel (such as the RBF kernel or Gaussian kernel) has good interpolation abilities, but fails to provide longer range extrapolation. Jordan [20] demonstrated that a mixed kernel, a linear combination between poly kernel and Gaussian kernel, gives the extrapolation and interpolation much better than either a local kernel or a global kernel. In this paper, a mixture of kernels is employed to replace the single kernel in continuous kernel estimator.

Silverman [33] pointed out that the selection of optimal bandwidth is much more important than kernel function selection. This is because smaller values of bandwidth make the estimate look "wiggly" and show spurious characteristics, whereas too large values of bandwidth will result in an estimation that is too smooth, in the sense that it is too biased to reveal structural features. However, there is not a generally accepted method for choosing the optimal bandwidth. The popular methods [19], [30] include rules of thumb, oversmoothing, least squares cross validation, biased cross validation, direct plug-in methods, solve-the-equation plug-in methods, and bootstrap methods.

In contrast to the existing bandwidth selections, this paper employs a mixture kernel for building kernel estimators, and presents a grid search strategy to select the optimal bandwidth. From the experiments, the proposed method really not only demonstrates better extrapolation and interpolation, but also decreases the exponential time nearly to a polynomial one.

3 Nonparametric Iterative Imputation Method

As demonstrated in this paper, the numbering for sections is upper case Arabic numerals, then upper case Arabic numerals separated by periods. Initial paragraphs after the section title are not indented. Only the initial, introductory paragraph has a drop cap. Before presenting the new imputation algorithm in Section 3.1, the work in [29] is first recalled, which reported on kernel functions for discrete attributes (including ordering and nonordering discrete attributes/variables). Then, a mixture kernel function is proposed by combining a discrete kernel function with a continuous one presented in [26]. Furthermore, a new estimator is constructed based on the mixture kernel. Section 3.2 develops novelty kernel estimators for discrete and continuous target values, respectively. In Section 3.3, the nonparametric iterative imputation algorithm is extended from a single kernel to a mixture of kernels. In Section 3.4, the nonparametric iterative imputation algorithm is designed and simply analyzed.

3.1 Single Kernel Imputation Method by Mixture Kernel Estimator

Let $X_i^d \in S^r$ denote a $k \times 1$ vector of the estimator designed for discrete *variables* (or *attributes*), and $X_i^c \in S^p$ the estimator for continuous variables remained, where d and p are the number of dimensions of discrete and continuous variables, respectively. Assume that $X_{u,i}^d$ denotes the uth component of X_i^d, and $X_{u,i}^c$ contains $c_u \geq 2$ different values, i.e., for $u = 1, \ldots, k$, $X_{u,i}^d \in \{0, 1, \ldots, c_u - 1\}$, and $X_i = (X_i^d, X_i^c) \in S^r \times R^p$.

The kernel function for discrete variables, proposed in [29], is simply called as *discrete kernel function* in the rest of the paper. There are two kinds of discrete kernel functions, nonordering and ordering discrete kernel functions (see Definitions 1-3).

A nonordering discrete kernel function is constructed by four steps:

1. Define a univariate kernel function:

$$l\left(X_{u,i}^d, X_u^d\right) = \begin{cases} 1 & \text{if } X_{u,i}^d = X_u^d, \\ \lambda, & \text{otherwise.} \end{cases} \tag{1}$$

2. Define an indicator function $I(X_{u,i}^d \neq X_u^d)$, whose value is 1 if $I(X_{u,i}^d \neq X_u^d)$, and 0 otherwise.
3. Define $d_{x_i,x} = \sum_{u=1}^{k} I(X_{u,i}^d \neq X_u^d)$, whose value is the number of disagreement components between X_i^d and x^d.
4. Construct a nonordering discrete kernel function according to Definition 1.

Definition 1 (Nonordering Discrete Kernel Function). *For a $k \times 1$ vector with nonordering discrete values, such as $X_i^d \in S^r$, $x_i^d \in S^r$, then its corresponding kernel function is defined as follows:*

$$\left(X_i^d, x_i^d, \lambda\right) = \prod_{u=1}^{k} l\left(X_{u,i}^d, X_u^d\right) = 1^{k-d_{x_i,x}} \lambda^{d_{x_i,x}} = \lambda^{d_{x_i,x}}, \tag{2}$$

where λ is the smooth parameter and will be decided in experiments.

Similarly, a kernel function for ordering discrete variables is defined as follows:

Definition 2 (Ordering Discrete Kernel Function). *Let $l(X_{i,u}, x_u) = \lambda^s$, where $\left|X_{i,u} - x_u\right| = s$, then the ordering discrete kernel function is defined as*

$$L(X_i, x_i, \lambda) = \prod_{u=1}^{k} \lambda^{\left|X_{i,u} - x_u\right|} = \lambda^{\delta_{x_i,x}}, \tag{3}$$

where $\delta_{x_i,x} = \sum_{u=1}^{k} \left|X_{i,u} - x_u\right|$ is the L_1-distance between X_i^d and x^d, and λ is the smooth parameter.

Combining (2) with (3), it is easy to obtain the kernel function for discrete variables.

Definition 3 (Discrete Kernel Function). *Discrete variables include nonordering or ordering variables, based on Definitions 1 and 2, the discrete kernel function is defined as follows:*

$$L\left(X_i^d, x_i^d, \lambda\right) = \lambda^{d_{x_i,x} + \delta_{x_i,x}}. \tag{4}$$

Qin et al. [26] presented a kernel function for continuous variables as follows:

Definition 4 (Continuous Kernel Function). *For a $k \times 1$ vector with continuous values, such as, $x \in R^p$, then its kernel function is $K(x - X_i/h)$, and the K(.) is a mercer kernel, i.e., positive definite kernel.*

Based on the above work (Definitions 3 and 4), in this paper, a mixture kernel function is proposed for mixed independent attributes (see Definition 5).

Definition 5 (Mixture Kernel Function). *With integrating the discrete and continuous kernel functions, a mixture kernel function is constructed as follows:*

$$K_{h,\lambda,ix} = K(^{x - X_i}/_h) L\left(X_i^d, x_i^d, \lambda\right), \tag{5}$$

where $h \to 0$ and $\lambda \to 0$ (λ, h is the smoothing parameter for the discrete and continuous kernel functions, respectively), and $K_{h,\lambda,ix}$ is a symmetric probability density function.

Consequently, some estimators are constructed with (5) as follows:

Definition 6 (Estimator for Continuous Missing Attributes). *The kernel estimator, $\hat{m}(x)$, for continuous missing target values m(x) for data sets with mixed independent attributes is defined as follows:*

$$\hat{m}(x) = \frac{n^{-1} \sum_{i=1}^{n} Y_i K_{h,\lambda,ix}}{n^{-1} \sum_{i=1}^{n} K_{h,\lambda,ix} + n^{-2}}, \tag{6}$$

where the item n^{-2} in $\hat{m}(x)$ is only used for avoiding the denominator to be 0.

When the missing value m(x) is in a discrete attribute, the estimator is constructed with Definition 7.

Definition 7 (Estimator for Missing Discrete Attributes). *Let $D_{m(x)} = (0, 1, \ldots, c_u - 1\}$ denote the range of m(x), one could estimate m(x) by*

$$\hat{m}(x) = \frac{n^{-1}\sum_{i=1}^{n} Y_i K_{h,\lambda,ix}}{n^{-1}\sum_{i=1}^{n} K_{h,\lambda,ix} + n^{-2}} + \lambda \frac{n^{-1}\sum_{i=1}^{n}\sum_{y\in D_y, y\neq Y_i} K_{n,\lambda}}{n^{-1}\sum_{i=1}^{n} K_{n,\lambda}}, \quad (7)$$

where $l(Y_i, y, \lambda) = 1$ if y $= Y_i$, *and* λ if y $\neq Y_i$.

Equations (6) and (7) are designed for imputing missing target values only once, and they are similar to the function presented in [29]. However, there are essential differences between these two methods. For example, the kernel functions proposed in this paper employ the method given in [26] for dealing with continuous missing attributes. This is different from that in [29] because the former has been demonstrated to outperform other methods for imputing continuous attributes (including the method in [29)]. In particular, in this paper, a mixture kernel is constructed instead of the single kernel ([26]) in the continuous kernel estimator, where the mixture kernel will further be analyzed and explained in Section 3.3.

Theorem 1 (Asymptotic Normality for Single Imputation). *The estimator in Definition 6 or 7 is of asymptotic normality.*

The proof of Theorem 1 is informally outlined as follows:

Racine and Li [29] have presented the asymptotic normality of the estimator (see (8)) for discrete independent attributes.

$$\hat{m}(x) = \frac{n^{-1}\sum_{i=1}^{n} Y_i K_{h,\lambda,ix}}{n^{-1}\sum_{i=1}^{n} K_{h,\lambda,ix}} + \lambda \frac{n^{-1}\sum_{i=1}^{n}\sum_{y\in D_y, y\neq Y_i} K_{n,\lambda}}{n^{-1}\sum_{i=1}^{n} K_{h,\lambda}}. \quad (8)$$

Comparing (7) with (8), the only difference is that an item n^{-2} is added to (7) for avoiding the denominator to be 0. The item does not affect that the asymptotic normality of (7) and it is similar to that of (8), because the term n^{-2} is a positive value small enough. For the former part in (7) (i.e., (6)), the estimator is obtained by using a mixture kernel to replace the single kernel in existing continuous kernel estimator. Because the mixture kernel always leads to a mercer kernel, comparing (6) with the continuous kernel estimator in [29], the asymptotic normality of (6) is preserved from that of the continuous kernel estimator in [29]. Consequently, Theorem 1 holds.

Given the above improvement, one can find that the above imputation method for missing values is a single-imputation method. It is not feasible to use the above methods (based on (6) or (7)) to impute missing values when there are only a few completed instances, because the estimators can lead to bias in these cases, and this will become more serious when the missing rate is higher. In practice, most databases have a high missing rate, especially in industrial databases. For example, the industrial data set introduced in [21], there are 4,383 records in the data set, but none of the records are complete and only 33 variables out of 82 have more than 50 percent of the records completed. It certainly results in a low imputation performance to impute the missing values with such a limited percentage of complete information [42]. Therefore, it is important to consider how to utilize all the available information in the data set because the observed information in incomplete instances (with missing values) assists in improving the imputation performance [42]. The imputation algorithm designed in this paper gives a consideration to all the observed information in a data set.

Based on (6) and (7), a mixture-kernel-based nonparametric iterative imputation method is proposed to utilize all the observed information, including the observed information in incomplete instances. In the kernel estimator for continuous variables, a mixture kernel is used to replace the single kernel so as to obtain better interpolation and extrapolation. This algorithm will be designed in the following sections.

3.2 Nonparametric Iterative Imputation

The sets of respondents (observed values in the target variable Y) and nonrespondents (missing values in Y) are denoted by S_r $(r = 1, \ldots, \gamma)$ and S_m $(m = n - \gamma)$, respectively. Based on the results in Section 3.1, a random sample $(\text{sample size} = n)$ of incomplete data associated with a population (X^d, X^c, Y, δ) can be represented as follows:

$$(X^d, X^c, Y_i, \delta), i = 1, 2, \ldots, n,$$

where the X^d's, X^c's are observed when $\delta_i = 0$; Y_i is missing when $\delta_i = 1$.

The tth imputation value Y_i^t of the ith missing value is denoted by $\hat{Y}_i^t$ that is evaluated with (9) as follows:

$$\hat{Y}_i^t = \hat{m}_t(X_i) + \varepsilon_i^t, \quad (9)$$

where t is the number of iterative imputation times, $\hat{m}_t(x)$ is the kernel estimator for $m_t(x)$ $(x \in R^{d+p})$ based on the completely observed pairs (X^t, Y^t), and $\{\varepsilon_i^t\}$ is a simple random sample of size m with replacement from $\{\hat{Y}_i^t - \hat{m}_t(X_i)\}$ $i \in S_r$, in the tth imputation.

Definition 8 (Iterative Kernel Estimator for Continuous Target Variable). *If the independent attribute is mixed and the missing target variable Y is continuous variable, the kernel estimator, $\hat{m}_t(x)$, of Y is defined as follows:*

$$\hat{m}_t(x) = \frac{n^{-1}\sum_{i=1}^{n} Y_i^t K_{h,\lambda,ix}}{n^{-1}\sum_{i=1}^{n} K_{h,\lambda,ix} + n^{-2}}, \quad (10)$$

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

In particular,

$$\hat{Y}_i^1 = \frac{1}{r}\sum_{i=1}^{r} Y_i,$$

and the product kernel function $K(x - X_i/h)$ is replaced with a mixture kernel in this paper (will be explained in Section 3.3), whereas a single kernel is usually used for continuous target variables in traditional imputation techniques.

Definition 9 (Iterative Kernel Estimator for Discrete Target Variable). *Let $D_y = \{0, 1, \ldots, c_y - 1\}$ denote the range of Y_i in the discrete target variable Y, and the joint density of (Y_i, X_i) is estimated by $n^{-1}\sum_{i=1}^{n} l(Y_i^t, y_t, \lambda)K_{h,\lambda,ix}$, where the indicator function $\sum_{i=1}^{n} l(Y_i^t, y_t, \lambda) = 1$ if $Y_i^t = y_t$, and = λ otherwise. The kernel estimator, $\hat{m}_t(x)$, of Y is defined as follows:*

$$\hat{m}_t(x) = \frac{\sum_{i=1}^{n} \sum_{y \in D_y, y \neq Y_i} l\left(Y_i^y, y_t, \lambda\right) y_t K_{n,\lambda}}{\sum_{i=1}^{n} K_{h,\lambda}}, \qquad (11)$$

where

$$y_i^t = \begin{cases} Y_i & if\ \delta_{\mathrm{i}} = 0\ or\ i = 1, \ldots, \mathrm{r}, \\ \hat{Y}_i^{t-1} & if\ \delta_{\mathrm{i}} = 1\ or\ i = \mathrm{r}+1, \ldots, \mathrm{n}. \end{cases}$$

In particular, $\hat{Y}_i^1$is the best common class in the discrete target variable, and $\varepsilon_i^t = 0,\ i = r+1, \ldots, n$. The product kernel function $K(x - X_i/h)$ in (11) is replaced with a mixture kernel, whereas a single kernel is used in (7) and (8).

Thereby, in the following sections, $\hat{m}_t(x)$ is used to denote the kernel estimator regardless of whether or not $\mathrm{Y_i}$ is continuous or discrete, and $\hat{m}_t(x)$ will also be regarded as the imputed value of $m_t(x)$. Note that the kernel weights can be added up to $1 + (c_t - 1)\lambda \neq 1$ for $\lambda \neq 0$ in the case of (2) and (4). Racine and Li [29] demonstrated that this did not affect the nonparametric estimator defined in (2) and (4) because the kernel function appears in both the numerator and denominator of them. And the kernel function can be multiplied by any positive constant without changing the definition of $\hat{m}(x)$. In addition, in (6), the estimator is changed to conventional one when $\lambda = 0$, whereby one uses a frequency estimator to deal with the discrete variables. In Section 4, some experiments will be conducted to compare the proposed approach with the frequency estimator method.

Theorem 2 (Asymptotic Normality for Iterative Imputation). *The nonparametric kernel density estimator in Definition 8 or 9 is of asymptotic normality.*

The proof of this theorem is simply outlined as follows: In Definitions 8 and 9, if the number of the final imputation times is t, it means that the single imputation is performed t times. In each imputation, the difference between the single imputation (presented in Definition 6 or 7) and the iterative imputation (in Definition 8 or 9) is the utilizing of the imputed values. In the proposed iterative imputation, the imputed values are treated the same as the observed ones and used to impute subsequent missing values, whereas the single imputation does not utilize the imputed values when imputing subsequent missing values. However, this difference does not affect on the asymptotic normality of estimators due to the variance. Therefore, based on Theorem 1, each imputation in the iterative imputation (in Definition 8 or 9) is still of asymptotic normality. This means that the proposed iterative imputation is of asymptotic normality.

3.3 Mixture Kernel FunctionResearch into Missing Value Imputation

Zheng et al. [41] pointed out that a global kernel (such as the polynomial kernel) can present better extrapolation at lower order degrees, but need more higher order degrees for receiving a good interpolation. And a local kernel has better interpolation, but fails to provide stronger extrapolation. They also demonstrated that a mixture of kernels can lead to much better extrapolation and interpolation than using either the local or global kernels. In this research, the proposed imputation approach is based on a mixture kernel function constructed in Definition 10 as follows:

```
//the first imputation
FOR each MVi in Y
    M̂V_i^1 = mode ( S^r in Y );  // if Y is a discrete variable
    M̂V_i^1 = mean ( S^r in Y );  // if Y is a continuous variable
END FOR

//t-th iteration of imputation (t>1)
t=1;
REPEAT
t++;
FOR each MV_i in Y
    MV_i = M̂V_i^{t-1}, p ∈ S_m, p = 1,...,m, p ≠ i
    M̂V_i^t is got based on Eq. (11) // if discrete variable
    M̂V_i^t is got based on Eq. (10) // if continuous variable
END FOR
UNTIL
  |CA_t − CA_{t−1}| ≥ ε   // if discrete variable
  Convergence or Cycling   // if continuous variable

3.0 //finishing the iterative imputation
OUTPUT
t; // t is the iterative times
Completed dataset;
```

Fig. 1. The pseudocode of the proposed algorithm.

Definition 10 (Linear Mixture Kernel Function). *Let* $K_{poly} = (< x, x_i > +1)^q$, $K_{rbf} = \exp(-(x - x_i)^2/\sigma^2)$, *a linear mixture kernel function is defined as follows:*

$$K_{mix} = \rho K_{poly} + (1 - \rho) K_{rbf}, \qquad (12)$$

where q is the degree of the polynomial, σ is the width of the radial basis function (RBF), and ρ is the optimal mixed coefficient ($0 \leq \rho \leq 1$*). The values of ρ, q, and σ are constant scalars, but have to be determined with experiments.*

For the above mixture kernel model, four coefficients, namely λ (the parametric for discrete kernel function), ρ, q, and σ, are used so as to get the optimal result. It is very difficult to get the optimal result while dealing simultaneously with so many coefficients because the time complexity is often exponential. To circumvent this problem, in this research, a grid search strategy is designed for selecting the optimal bandwidth based on a principle that minimizes the Approximate Mean Integrated Square Error (AMISE) of the tth imputed missing values $\hat{m}_t(X_i)$. Let the AMISE of the tth imputed missing values $\hat{m}_t(X_i)$ be

$$CV_t(\lambda, (\rho, p, \sigma)) = \min\left\{ \sum_{i=1}^{n} [Y_i^t - m_{t-i}(X_i)]^2 \right\}, \qquad (13)$$

where $\hat{m}_{t-i}(X_i)$denotes the "leave-one-out" kernel estimator of $\hat{m}_{t-i}(X_i)$. This leads to the fact that the search space is only a part of the whole spaces for the grid parameter values. In Section 4.5, some experiments will be conducted to illustrate the use of the grid search method in choosing bandwidth.

3.4 Algorithm Design

In the proposed imputation approach, the ith missing value is denoted by MV_i and the imputed value of MV_i in tth iteration imputation is regarded as $\hat{M}V_i^t$. The algorithm is designed in Fig. 1.

From the above algorithm, all the imputed values are used to impute subsequent missing values, i.e., the $(t+1)$th $(t \geq 1)$ iteration imputation is carried out based on the imputed results of the tth imputation, until the filled-in values converge or begin to cycle or satisfy the demands of the users.

In the first iteration of imputation in the above algorithm, all the missing values are imputed using the mean for continuous attributes (the mode for discrete ones). Using the mean (or mode) of an attribute to replace missing values is a popular imputation method in machine learning and statistics. However, Brown [8] thought that imputing with the mean (or mode) will be valid if and only if the data set is chosen from a population with a normal distribution. This is usually impossible for real applications because the real distribution of a data set is not known in advance. On the other hand, Rubin [31] demonstrated that a single imputation cannot provide valid standard errors and confidence intervals, since it ignores the uncertainty implicit in the fact that the imputed values are not the actual values. Therefore, running extra iteration-imputations based on the first imputation is reasonable and necessary for better dealing with the missing values.

Since the second iteration of imputation, each of iteration-imputation is carried out based on former imputed results with the nonparametric kernel estimator. During the imputation process, when the missing value $\hat{MV}_i^t$ is imputed based on (10) or (11), all other missing values are regarded as observed values, i.e., $MV_i = \hat{MV}_i^{t-1}$, $p \in S_m$, $p = 1, \ldots, m, p \neq i$. In particular, $\hat{MV}_i^1 = \text{mean}\,(S^r \text{in}\, Y)$ if the target variable Y is a continuous variable, $\hat{MV}_i^1 = \text{mode}\,(S^r \text{in}\, Y)$ if Y is a discrete one in this algorithm. The iteration-imputation for missing continuous attributes will be terminated when the filled-in values converge or begin to cycle (details about cycle will be presented in Section 4.1). For discrete missing values, the imputation algorithm will be terminated if $|CA_t - CA_{t-1}| \geq \varepsilon$ based on the principle of the parameter iterative algorithm EM [13], where ε is a nonnegative constant specified by users; the classification accuracy for the tth imputation is denoted by CA_t. Then the time of iteration of the algorithm is t for imputing a discrete missing attribute because the first imputation has been finished.

4 Experimental Study

We considered several data sets from real applications and data sets taken from the UCI data set in [7] (see Table 1) in this section.

The first four data sets are used in Sections 4.1 and 4.4, and the remaining data sets for Sections 4.2 and 4.3. None of these data sets have missing values. The selected data sets let us compare the imputed values with their real values. For these complete data sets, missed values are generated at random so as to systematically study the performance of the proposed method. The percentage of missing values (the "missing rate") was fixed at 10, 20, 30, 50, and 80 percent for each data set. For comparison with the proposed method (denoted by **Mixing**), four selected imputation methods are the nonparametric iterative single-kernel imputation method with a polynomial kernel (denoted by **Poly**), a nonparametric iterative single-kernel imputation method with the RBF kernel (**RBF**), the traditional kernel nonparametric missing value imputation method from [38] (**Normal**), and the conventional frequency estimator (**FE**), setting $\lambda = 0$ in the experiments.

TABLE 1
Databases Used in Our Experiments

Name	Y	Attri.Type	#(attr.)	#(ins.)
Auto-mobile	C	15/10/1	26	205
Auto-mpg	C	4/1/3	8	398
HHG1984	C	2/1/3	6	896
Housing	C	12/1/1	14	506
Abalone	D(29)	7/1/0	8	4177
AC	D(10)	2/1/4	7	6000
Annealing	D(6)	6/29/3	38	798
CMC	D(3)	2/3/4	9	1473
Pima	D(2)	6/0/2	8	768
Vowel	D(11)	10/0/0	10	528

Each column represents the name of the database, dependent attribute (C: continuous, D(29): discrete attribute has 29 classes), the type of independent attribute (continuous/un-ordering/ordering), the number of independent attributes, and the number of instances, respectively.

For the missing target value in the data set, they are first imputed with mean (or mode) method for continuous (or discrete) variable. From the second iteration of imputation, previously imputed missing values are regarded as known values and used in next iteration of imputation. This leads to the utilization of observed information in incomplete instances. The imputation process is stopped when the imputation results satisfy the conditions presented in Section 3.4.

4.1 Y_i Is a Continuous Variable

The first data set is the balanced panel data from [43] and is indicated as **HHG (1984)** in the paper. The other three data sets were obtained from UCI, i.e., Auto-mpg, Housing, and Automobile.

4.1.1 Convergence of the Imputed Values

An important practical consideration with the iterative imputation methods is to determine at which point additional iterations have no meaningful effect on the imputed values, i.e., how to judge the convergence of the algorithm. Each iteration of EM algorithm is guaranteed [13] to be nondecreasing in maximum likelihood, thus, EM algorithm converges to a local maximum in likelihood. However, it is difficult to make similar guarantees for nonparametric methods. Caruana [9] concluded that the average distance (i.e., the attribute values move from successive iterations when the algorithm is applied to the pneumonia data set) drops to zero, means that no missing values have changed and that the method has converged in the parametric model. But in this case, motions of nonparametric models do not drop all the way to zero, indicating that the algorithm converged to a cycle. However, Caruana [9] and [42] argued that cycles are rare with parametric methods if density calculations are exact, and that they are more likely in the nonparametric models, but the nonparametric method has never diverged in their experiments.

TABLE 2
The Results of the Iterative Imputation Times (Denoted by T) and the Mean for the Imputed Values After Convergence (Denoted by V) for the Data Set HHG (1984) at Different Missing Rates

	10%		20%		30%		50%		80%	
	T	V	T	V	T	V	T	V	T	V
Mixing	8	0.085	10	0.098	14	0.128	17	1.27	20	1.53
Poly	10	0.103	13	0.138	19	0.167	21	1.79	25	2.11
RBF	11	0.107	14	0.14	20	0.174	21	1.95	29	2.86
Normal	14	0.121	20	0.1851	28	0.219	27	2.77	30	3.01
FE	13	0.117	17	0.163	22	0.197	23	2.26	29	2.59

Here, a stopping criterion is designed for nonparametric iterations. With t imputation times, there will be $(t-1)$ chains of iterations. Note that the first imputation won't be considered when talking about the convergence because the final results will be decided mainly by imputation from the second imputation. Of course, the result in the first imputation always generates, to some extent, effects for the final results. This will be discussed in future work. Since the number of individual components of missing values is high, it is not feasible to monitor convergence for every imputed missing value. Schafer [32] considered that since convergence rates are closely related to missing information, it makes sense to focus on parameters (in our paper, it will be variance and mean of the imputed values. Obviously, we can also use other parameters, such as distribution function, or quantile) for which the fractions of missing information are high.

Assuming that the 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,

$$\text{If } \frac{M_l}{M_{l+2}} \to 1, \text{ and } \frac{V_l}{V_{l+2}} \leq \varepsilon. \tag{14}$$

It can be inferred that there is little change in imputations between the first and third time and one can stop iterating without substantial impact on the resulting inferences. Unlike the converged condition in the EM algorithm, in this paper, it first summarizes a stopping strategy by using terminology such as "satisfying a convergence constraint" rather than "achieving convergence" to clarify that convergence is an elusive concept with iterative imputation. And the variability across the first third of the chains is compared with the variability across the last third of the chains. The middle one in each of the three iterations is ignored to avoid dependence between two segments in each iteration imputation, since consecutive iterations tend to be correlated.

TABLE 3
The Results of T and V After Convergence for the Data Set Housing at Different Missing Rates

	10%		20%		30%		50%		80%	
	T	V	T	V	T	V	T	V	T	V
Mixing	5	0.048	6	0.070	10	0.096	15	0.351	17	0.958
Poly	8	0.061	10	0.088	16	0.120	18	0.563	21	1.235
RBF	9	0.065	10	0.091	15	0.119	18	0.694	20	1.225
Normal	14	0.095	18	0.165	23	0.201	23	0.854	27	1.653
FE	12	0.071	14	0.129	18	0.165	19	0.679	22	1.265

TABLE 4
The Results of T and V After Convergence for the Data Set Auto-mpg at Different Missing Rates

	10%		20%		30%		50%		80%	
	T	V	T	V	T	V	T	V	T	V
Mixing	4	0.022	5	0.030	7	0.032	12	0.045	19	0.179
Poly	6	0.041	8	0.068	10	0.082	14	0.091	24	0.256
RBF	6	0.040	9	0.070	10	0.083	13	0.085	23	0.233
Normal	10	0.095	16	0.139	19	0.155	15	0.176	26	0.589
FE	8	0.056	12	0.081	15	0.115	13	0.126	22	0.429

Compared with the method used for taking the convergence of the algorithm into account in [9], this paper considers the mean of the imputed values as well as taking the variance of the imputed values into account. Caruana [9] only considered the former parametric. That shows the convergence condition in the proposed approach is stronger than the existing one. On the other hand, the parameter (such as mean or variance) of the imputed values in an iteration of imputation is partitioned into three chains and given up considering the second part. Caruana [9] took all the imputed values in one imputation into account. Obviously, the proposed approach can efficiently avoid dependence on imputed values.

Tables 2, 3, 4, and 5 show the experimental results, including the iterative times and the average value after these five algorithms have converged for the data sets HHG (1984), Housing, Auto-mpg, and Automobile, with missing rates of 10, 20, 30, 50, and 80 percent, respectively. The iterative times after the algorithm has converged for four nonparametric methods (i.e., Mixing, Poly, RBE, and FE) for mixed independent attributes are lesser than the Normal algorithm that deals with only continuous independent attributes. Moreover, they have better efficiency in convergence than the Normal algorithm because the average distance is closer to zero than that of the Normal algorithm. This demonstrates that the proposed approach is significantly better than the other methods in these experiments.

4.1.2 RMSE and Correlation Coefficient

Below, the RMSE is used to assess the predictive ability after the algorithm has converged:

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(e_i - \tilde{e}_i)^2}, \tag{15}$$

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 the

TABLE 5
The Results of T and V After Convergence for the Data Set Automobile at Different Missing Rates

	10%		20%		30%		50%		80%	
	T	V	T	V	T	V	T	V	T	V
Mixing	3	0.018	5	0.021	8	0.030	10	0.049	15	0.097
Poly	4	0.019	6	0.025	10	0.035	14	0.085	17	0.143
RBF	4	0.018	5	0.022	12	0.034	16	0.075	18	0.126
Normal	5	0.025	8	0.027	13	0.060	19	0.153	24	0.236
FE	4	0.020	5	0.024	11	0.051	16	0.125	20	0.224

TABLE 6
The RMSE (Denoted by R) and Correlation Coefficient (Denoted by CC) After Convergence for the Data Set HHG (1984)

	10%		20%		30%		50%		80%	
	CC	R	CC	R	CC	R	CC	R	CC	R
Mixing	0.98	0.351	0.96	0.562	0.93	0.701	0.92	1.952	0.85	3.257
Poly	0.97	0.532	0.94	0.722	0.91	0.982	0.90	2.427	0.83	4.526
RBF	0.97	0.482	0.93	0.730	0.90	1.003	0.90	2.533	0.82	5.327
Normal	0.96	0.924	0.90	1.575	0.86	1.928	0.88	3.895	0.80	8.956
FE	0.95	0.678	0.91	0.911	0.89	1.625	0.89	3.452	0.81	7.622

prediction is. At the same time, the correlation coefficient between the actual and predicted values of missing attributes is calculated after convergence.

Tables 6, 7, 8, and 9 show the results: the predictive accuracy (through the measure of RMSE) and the correlation coefficient between the actual and predicted values of missing attributes after these five algorithms have converged for the data sets HHG (1984), Housing, Auto-mpg, and Automobile, with missing rates of 10, 20, 30, 50, and 80 percent, respectively.

These results demonstrate that the new approach completely dominates the other four algorithms, especially when the missing rate is moderate, such as 20 percent, since the proposed approach is nonparametric, with mixture kernels, and for mixed attributes during the imputation process.

The four algorithms (i.e., Mixing, Poly, RBF, and FE) that handle the mixed independent attributes outperform the traditional algorithm, Normal, in terms of the RMSE and correlation coefficient, under different missing rate cases on these three real data sets. The reason is that the Normal algorithm cannot handle the discrete independent attributes well because it treats the discrete attributes as continuous, and it will further aggravate the "curse of dimensionality" when the number of continuous attributes increases.

Compared with the four algorithms (dealing with the mixed independent attributes) in Tables 6, 7, 8, and 9, for all kinds of situations in the experiments, the results of Mixing, Poly, and RBF are better than that of the FE algorithm in terms of the RMSE and correlation coefficient. When $\lambda = 0$, the estimator in (7) and (8) will become the conventional frequency estimator method to deal with the discrete variables. However, the FE algorithm has a major weakness because the number of discrete cells may exceed the sample size. Therefore, it does not have enough observations in each cell for building a nonparametric estimator. The estimators for other three algorithms smooth the discrete variables to avoid this problem, as Racine and Li [29] argued, and they can reduce the variance significantly by a trade-off between bias and variance, resulting in performance much better than the frequency estimator for finite samples.

TABLE 8
The RMSE and Correlation Coefficient (Denoted by CC) After Convergence for the Data Set Auto-mpg

	10%		20%		30%		50%		80%	
	CC	R	CC	R	CC	R	CC	R	CC	R
Mixing	0.97	0.495	0.95	0.651	0.94	0.832	0.92	2.682	0.88	7.530
Poly	0.95	0.622	0.94	0.900	0.91	1.433	0.90	3.267	0.88	9.102
RBF	0.95	0.624	0.93	0.902	0.90	1.451	0.90	2.962	0.88	8.125
Normal	0.92	1.204	0.88	2.020	0.87	2.826	0.85	4.176	0.82	9.983
FE	0.94	0.958	0.92	1.623	0.89	2.053	0.87	3.269	0.86	9.624

Considering the results of Mixing, RBF, and Poly, all the results of the Mixing algorithm are better than the other two. This means that using mixture kernels in nonparametric kernel estimation can provide much better learning capacity and generalization ability than those estimators only using either the local or global kernels. On the other hand, the performance of most situations in the Polynomial kernel is evidently better than the ones in RBF. It can be concluded that a global kernel, such as the Polynomial kernel, is very good at capturing general trends and extrapolation behavior, and only a little of a local kernel (such as the RBF kernel) needs to be added to the global kernel to obtain a good combination of interpolation and extrapolation abilities in the mixture kernel.

4.2 Y_i Is Discrete Variable

The UCI data sets "Abalone," "Pima," "Vowel," "CMC," "Anneal," and "AC" in which the class attribute is discrete are applied to the above five methods to compare the performances in terms of classification error rate and paired t-test.

Tables 10 and 11 show the results: the iterative times and predictive accuracy with respect to classification error rate after these five algorithms have terminated on six data sets. Similar to the results for imputing continuous missing values, the results of the Mixing algorithm for imputing discrete missing values are better than the other four algorithms in all respects, such as iterative time or predictive error rate. For example, all the algorithms that considered discrete independent attributes achieved from 2.6 to 41.5 percent improvement, with respect to classification errors, over the conventional Normal algorithm. These three algorithms (Mixing, Poly, and RBF) averagely outperform the FE algorithm by about 8-16 percent. Most of the results of Poly are better than those of the RBF. The best method, the Mixing algorithm, outperforms the Poly, RBF, FE, and Normal

TABLE 7
The RMSE and Correlation Coefficient (Denoted by CC) After Convergence for the Data Set Housing

	10%		20%		30%		50%		80%	
	CC	R	CC	R	CC	R	CC	R	CC	R
Mixing	0.97	0. 502	0.96	0.699	0.94	0.973	0.90	1.452	0.86	3.965
Poly	0.95	0.673	0.94	0.920	0.92	1.257	0.88	2.694	0.83	4.256
RBF	0.95	0.672	0.93	0.930	0.91	1.258	0.88	2.753	0.83	4.858
Normal	0.93	0.145	0.89	2.519	0.87	2.907	0.86	3.562	0.81	5.638
FE	0.94	0.903	0.92	1.391	0.88	1.868	0.87	3.129	0.82	4.956

TABLE 9
The RMSE and Correlation Coefficient (Denoted by CC) After Convergence for the Data Set Automobile

	10%		20%		30%		50%		80%	
	CC	R	CC	R	CC	R	CC	R	CC	R
Mixing	0.98	0.475	0.98	0.754	0.96	0.959	0.92	1.759	0.89	2.636
Poly	0.97	0.495	0.96	0.895	0.94	1.257	0.90	2.524	0.88	3.255
RBF	0.97	0.485	0.96	0.786	0.95	1.125	0.91	2.548	0.86	3.628
Normal	0.95	0.502	0.94	1.204	0.92	1.966	0.88	4.257	0.82	5.265
FE	0.96	0.500	0.95	1.053	0.95	1.026	0.91	3.528	0.88	4.256

TABLE 10
Iterative Times After the Algorithms Have Finished the Iterative Imputation After Convergence for the Six Data Sets

	Abalone					Pima					Vowel				
	10%	20%	30%	50%	80%	10%	20%	30%	50%	80%	10%	20%	30%	50%	80%
Mixing	6	8	11	18	19	10	13	15	18	25	8	10	13	18	25
Poly	7	10	12	19	21	12	14	17	22	27	10	13	17	21	29
RBF	7	10	13	21	21	12	15	17	20	27	10	12	17	20	31
Normal	10	15	18	24	26	16	19	22	25	29	13	16	20	26	32
FE	8	11	16	20	22	14	19	20	24	28	12	16	20	23	30
	CMC					Anneal					AC				
	10%	20%	30%	50%	80%	10%	20%	30%	50%	80%	10%	20%	30%	50%	80%
Mixing	8	9	12	16	22	10	11	12	18	22	4	5	7	10	12
Poly	9	12	15	17	25	12	14	16	21	26	5	5	8	14	17
RBF	10	12	16	19	29	11	14	17	20	25	5	6	8	13	16
Normal	12	17	19	26	29	14	15	18	24	29	6	8	10	17	20
FE	10	15	17	20	26	12	14	15	19	24	5	7	9	15	18

TABLE 11
Classification Error Rate After the Algorithms Have Finished the Iterative Imputation After Convergence for the Six Data Sets

	Abalone					Pima					Vowel				
	10%	20%	30%	50%	80%	10%	20%	30%	50%	80%	10%	20%	30%	50%	80%
Mixing	0.208	0.235	0.298	0.356	0.521	0.116	0.131	0.167	0.365	0.489	0.145	0.169	0.193	0.254	0.425
Poly	0.232	0.259	0.316	0.385	0.595	0.139	0.158	0.194	0.374	0.563	0.152	0.180	0.227	0.298	0.485
RBF	0.232	0.260	0.316	0.395	0.624	0.139	0.159	0.199	0.401	0.625	0.152	0.195	0.230	0.264	0.457
Normal	0.284	0.305	0.351	0.415	0.675	0.172	0.223	0.251	0.425	0.685	0.187	0.229	0.264	0.312	0.510
FE	0.246	0.268	0.332	0.399	0.612	0.154	0.172	0.215	0.405	0.647	0.182	0.211	0.253	0.304	0.468
	CMC					Anneal					AC				
	10%	20%	30%	50%	80%	10%	20%	30%	50%	80%	10%	20%	30%	50%	80%
Mixing	0.127	0.153	0.176	0.247	0.341	0.145	0.194	0.254	0.351	0.415	0.085	0.097	0.106	0.356	0.459
Poly	0.141	0.159	0.196	0.289	0.452	0.192	0.205	0.296	0.384	0.512	0.096	0.125	0.156	0.445	0.524
RBF	0.146	0.161	0.198	0.262	0.595	0.187	0.226	0.312	0.421	0.576	0.089	0.121	0.125	0.478	0.564
Normal	0.176	0.227	0.243	0.287	0.875	0.212	0.294	0.335	0.445	0.609	0.112	0.148	0.149	0.511	0.785
FE	0.163	0.202	0.231	0.268	0.612	0.157	0.276	0.268	0.402	0.593	0.094	0.135	0.136	0.495	0.658

methods by about 4-14 percent, 7-18 percent, 16-39 percent, and 18-41.5 percent, respectively, with respect to classification accuracy.

We analyze the statistical significance of differences in classification errors between our method (i.e., Mixing) and the compared algorithms (the left four algorithms) based on paired t-tests at the 95 percent significance level. The significance is computed for each of the five amounts of missing values and each pair compared algorithm based on average classification errors across the six data sets. The results are presented in Table 12. The results show that our algorithm can more improve the classification errors even if with high missing rate.

4.3 Experimental Results between Single and Iterative Imputations

From Sections 4.1 and 4.2, the mixture-kernel-based nonparametric iterative imputation method outperforms the other methods under the assumption of iterative imputation. In particular, the proposed approach is the best one when the missing rate is moderate, such as 20 percent. This section will experimentally demonstrate the advantages of the proposed algorithm over single imputation about the constructed confidence interval for continuous variables. The results about classification accuracy will be presented by comparing the proposed approach with multiple imputation in Section 4.4. Due to the space limitation, only the results with a missing rate of 20 percent are presented in these two sections.

TABLE 12
Statistical Significance of Difference Algorithm **Mixing** and the Other Compared Algorithms, i.e., **Poly**, **RBF**, **Normal**, and **FE**, Respectively

	10%	20%	30%	50%	80%
Mixing ~Poly	++(2.1)	++(2.7)	++(2.1)	++(0.8)	++(0.21)
Mixing ~RBF	++(1.9)	++(2.9)	++(2.0)	++(0.7)	++(0.39)
Mixing ~Normal	++ (2.5)	++(3.4)	++(3.1)	++(1.5)	++(0.92)
Mixing ~FE	++ (2.0)	++(3.2)	++(2.5)	++(1.2)	++(0.15)

Note that "++" indicated the proposed algorithm that gives statistically significantly better classification errors for a given amount of missing values; positive t-value indicates that the classification errors for the proposed algorithm were better.

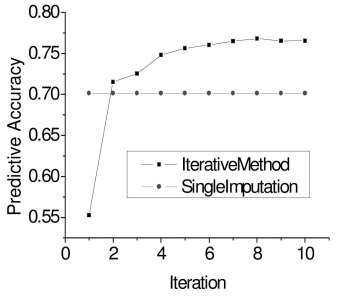

Fig. 2. Abalone.

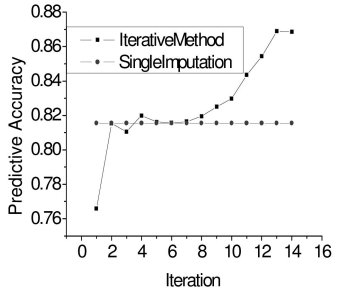

Fig. 3. Pima.

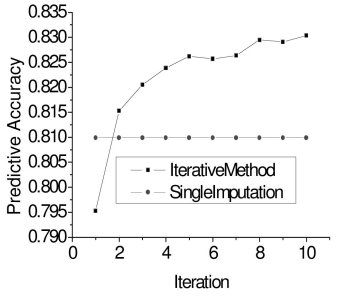

Fig. 4. Vowel.

At first, all missing values will be imputed based on the single imputation method, nonparametric mixture-kernel-based. So, (10) and (11) are revised as follows:

$$\hat{m}(x) = \frac{n^{-1}\sum_{i=1}^{n} Y_i K_{h,\lambda,ix}}{n^{-1}\sum_{i=1}^{n} K_{h,\lambda,ix}}, \tag{16}$$

$$\hat{m}(x) = \frac{\sum_{i=1}^{r}\sum_{y\in D_y} l(Y_i, y, \lambda) y K_{n,\lambda}}{\sum_{i=1}^{n} K_{h,\lambda}}. \tag{17}$$

Figs. 2, 3, 4, 5, 6, and 7 show the results for classification accuracy after the proposed algorithm has converged for the data sets "Abalone," "Pima," "Vowel," and "CMC," with missing rates of 20 percent. The results show that the performance of the proposed approach in the first imputation is worse than the result for single imputation. This is because the missing values are imputed by mean or mode in the first imputation in the proposed approach. Since the second imputation, most of the performances of the iterative method show that the mixture-kernel-based nonparametric iterative imputation method performs better than the single-imputation methods. The exception is found in data sets CMC and Pima. However, the left results in the other iterations are better than the results of single imputation in these two data sets.

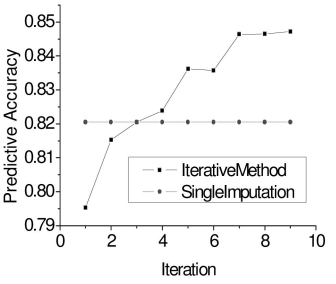

Fig. 5. CMC.

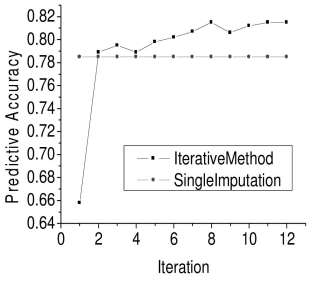

Fig. 6. Anneal.

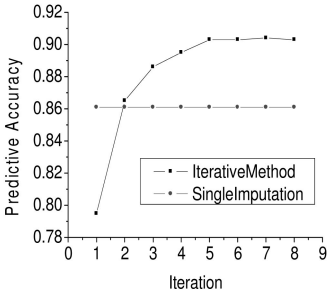

Fig. 7. AC.

4.4 Experimental Results between Multiple and Iterative Imputations

At first, the missing values were multiple imputed, based on [31], [4] with (16) or (17), and our method with (10) or (11), and the imputation times were 10, 6, and 5 for data set HHG (1984), Housing, and Auto-mpg because these iterative times allowed the proposed algorithm to converge. Next, a $(1-\alpha)$ percent interval estimate is constructed for mean based on [31], [4], where α is the significance level, and $\alpha = 0.05$ throughout the paper (other values for α can be chosen in practice). The performances of the multiple imputation are compared with the proposed algorithm according to their coverage probabilities (CPs) and average lengths of confidence intervals (denoted by AL) based on our constructed confidence intervals when the missing rate is 20 percent in these three data sets, and the results are shown in Table 13. These results demonstrate that our iterative imputation method performs better than the Multiple Method for the AL of confidence interval or convergence probabilities.

4.5 Experimental Selection for Bandwidth

The four coefficients, λ, ρ, q, and σ, have simultaneously been considered in order to get optimal results. It is very difficult for our algorithm to get an optimal result for so many coefficients at the same time because of the

TABLE 13
AL or CP of Confidence Interval between Multiple Imputation Methods and Iterative Imputation Method After Convergence for the Four Data Sets

	HHG(1984)		Housing		Auto-mpg		Auto-Mobile	
	AL	CP	AL	CP	AL	CP	AL	CP
Iterative	7.62	93.96	3.21	94.56	2.95	95.02	5.88	94.59
Multiple	8.25	93.50	3.95	94.47	3.20	94.52	9.26	93.65

exponential time complexity. Fortunately, Jordan [20] demonstrated experimentally that only a "pinch" of a local kernel, (i.e., $1-\rho=0.01$), needs to be added to the global kernel in order to obtain a combination of good interpolation and extrapolation abilities. Moreover, our experimental results show that using higher degrees of polynomials or larger widths of RBF kernels did not produce better results.

In the experiments, the coefficients ρ, q, and σ are combined with the coefficient λ so as to optimize the AMISE in sections for selecting optimal bandwidth, including nonordering attributes and ordering attributes. A grid search is used to optimize one parameter at a time. The important thing is to limit the search space in order to decrease the complexity of the algorithm.

First, the value of σ is limited. If the data are in a (0, 1) scaled input space, a pure RBF-kernel with $\sigma > 0.4$ behaves like a lower degree polynomial in the known learning space. That is precisely what it does not want when using the mixture kernel, because the polynomial part will consider the global behavior. The RBF-kernel part is specifically needed when modeling the local behavior. On the other hand, using one σ that is too small will result in overly complex models that also model the noise. Therefore, it is appropriate that σ be set between 0.15 and 0.3 in the proposed approach. Second, as a global kernel, the polynomial kernel is very good at capturing general trends and extrapolation behavior. The extrapolation behavior of the model becomes erratic and shows sudden increases or decreases in the response surface when the value of q is too high. So, a lower degree for the polynomial kernel may be chosen. In the experiments, $d > 2$ is seldom used, and q is usually set to 1 or 2. Third, the choice of ρ is related to how much of the local behavior needs to be modeled by the RBF kernel. Since the RBF-kernel is a very powerful kernel for modeling local behavior, it will not need much of its effects in order to see a huge improvement in the model. In the experiments, it is better if ρ is a value between 0.95 and 0.99.

Another question is how to find the best combination. The best approach, given the already limited search space, is to do a grid search. In the proposed approach, the value of σ is changed from 0.1 to 0.3, the degree of p is changed with 1 and 2; and the value of ρ is changed from 0.95 to 0.99. Once a combination is obtained, e.g., $\sigma=0.2$, $q=2$, $\rho=0.95$, they are fixed, and the value of λ is changed until the best AMISE is searched. The best AMISE can be obtained by scanning all the combinations of ρ, q, σ, and λ, where the complexity is reduced compared with the original one due to the limited search space.

5 Conclusions and Future Work

In this paper, a consistent kernel regression has been proposed for imputing missing values in a mixed-attribute data set by extending the method in [29]. The mixture-kernel-based iterative nonparametric estimators are proposed against the case that data sets have both continuous and discrete independent attributes. It utilizes all available observed information, including observed information in incomplete instances (with missing values), to impute missing values, whereas existing imputation methods use only the observed information in complete instances (without missing values). The optimal bandwidth is experimentally selected by a grid search method. The experimental results have demonstrated that the proposed algorithms outperform the existing ones for imputing both discrete and continuous missing values.

In future, we plan to further explore global or local kernel functions, instead of the existing ones, in order to achieve better extrapolation and interpolation abilities in learning algorithms.

Acknowledgments

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 nos. 90718020 and 10661003, the China 973 Program under grant no. 2008CB317108, the MOE Project of Key Research Institute of Humanities and Social Sciences at Universities (07JJD720044), the Guangxi NSF (Key) grants, and the Guangxi Colleges' Innovation Group. Shichao Zhang was the corresponding author for this paper.

References

[1] I.A. Ahamad and P.B. Cerrito, "Nonparametric Estimation of Joint Discrete-Continuous Probability Densities with Applications," *J. Statistical Planning and Inference,* vol. 41, pp. 349-364, 1994.
[2] P. Allison, *Missing Data.* Sage Publication, Inc., 2001.
[3] J. Aitchison and C.G.G. Aitken, "Multivariate Binary Discrimination by the Kernel Method," *Biometrika,* vol. 63, pp. 413-420, 1976.
[4] J. Barnard and D. Rubin, "Small-Sample Degrees of Freedom with Multiple Imputation," *Biometrika,* vol. 86, pp. 948-955, 1999.
[5] G. Batista and M. Monard, "An Analysis of Four Missing Data Treatment Methods for Supervised Learning," *Applied Artificial Intelligence,* vol. 17, pp. 519-533, 2003.
[6] H. Bierens, "Uniform Consistency of Kernel Estimators of a Regression Function under Generalized Conditions," *J. Am. Statistical Assoc.,* vol. 78, pp. 699-707, 1983.
[7] C. Blake and C. Merz UCI Repository of Machine Learning Database, http://www.ics.uci.edu/~mlearn/MLResoesitory.html, 1998.
[8] M.L. Brown, "Data Mining and the Impact of Missing Data," *Industrial Management and Data Systems,* vol. 103, no. 8, pp. 611-621, 2003.
[9] R. Caruana, "A Non-Parametric EM-Style Algorithm for Imputing Missing Value," *Artificial Intelligence and Statistics,* Jan. 2001.
[10] K. Cios and L. Kurgan, "Knowledge Discovery in Advanced Information Systems," *Trends in Data Mining and Knowledge Discovery,* N. Pal, L. Jain, and N. Teoderesku, eds., Springer, 2002.
[11] M.A. Delgado and J. Mora, "Nonparametric and Semi-Parametric Estimation with Discrete Regressors," *Econometrica,* vol. 63, pp. 1477-1484, 1995.
[12] A. Dempster and D. Rubin, *Incomplete Data in Sample Surveys: Theory and Bibliography,* W.G. Madow, I. Olkin, and D. Rubin, eds., vol. 2, pp. 3-10, Academic Press, 1983.

[13] A. Dempster, N.M. Laird, and D. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *J. Royal Statistical Soc.*, vol. 39, pp. 1-38, 1977.
[14] U. Dick et al., "Learning from Incomplete Data with Infinite Imputation," *Proc. Int'l Conf. Machine Learning (ICML '08)*, pp. 232-239, 2008.
[15] Z. Ghahramani and M. Jordan, "Mixture Models for Learning from Incomplete Data," *Computational Learning Theory and Natural Learning Systems*, R. Greiner, T. Petsche, and S.J. Hanson, eds., vol. IV: Making Learning Systems Practical, pp. 67-85, The MIT Press, 1997.
[16] J. Han and M. Kamber, *Data Mining Concepts and Techniques*, second ed. Morgan Kaufmann Publishers, 2006.
[17] M. Huisman, "Missing Data in Social Network," *Proc. Int'l Sunbelt Social Network Conf. (Sunbelt XXVII)*, 2007.
[18] G. John et al., "Ir-Relevant Features and the Subset Selection Problem," *Proc. 11th Int'l Conf. Machine Learning*, W. Cohen and H. Hirsch, eds., pp. 121-129, 1994.
[19] M.C. Jones, J.S. Marron, and S.J. Sheather, "A Brief Survey of Bandwidth Selection for Density Estimation," *J. Am. Statistical Assoc.*, vol. 91, no. 433, pp. 401-407, 1996.
[20] E.M. Jordaan, "Development of Robust Inferential Sensors: Industrial Application of Support Vector Machines for Regression," PhD thesis, Technical University Eindhoven, 2002.
[21] K. Lakshminarayan et al., "Imputation of Missing Data in Industrial Databases," *Applied Intelligence*, vol. 11, pp. 259-275, 1999.
[22] R. Little and D. Rubin, *Statistical Analysis with Missing Data*, second ed. John Wiley and Sons, 2002.
[23] R. Marco, "Learning Bayesian Networks from Incomplete Databases," Technical Report kmi-97-6, Knowledge Media Inst., The Open Univ., 1997.
[24] C. Peng and J. Zhu, "Comparison of Two Approaches for Handling Missing Covariates in Logistic Regression," *Educational and Psychological Measurement*, vol. 68, no. 1, pp. 58-77, 2008.
[25] Y.S. Qin et al., "Semi-Parametric Optimization for Missing Data Imputation," *Applied Intelligence*, vol. 21, no. 1, pp. 79-88, 2007.
[26] Y.S. Qin et al., "POP Algorithm: Kernel-Based Imputation to Treat Missing Values in Knowledge Discovery from Databases," *Expert Systems with Applications*, vol. 36, pp. 2794-2804, 2009.
[27] J.R. Quinlan, "Unknown Attribute values in Induction," *Proc. Sixth Int'l Workshop Machine Learning*, pp. 164-168, 1989.
[28] J.R. Quinlan, *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993.
[29] J. Racine and Q. Li, "Nonparametric Estimation of Regression Functions with Both Categorical and Continuous Data," *J. Econometrics*, vol. 119, no. 1, pp. 99-130, 2004.
[30] V.C. Raykar and R. DuraiswamiFast, "Fast Optimal Bandwidth Selection for Kernel Density Estimation," *Proc. SIAM Int'l Conf. Data Mining (SDM '06)*, pp. 524-528, 2006.
[31] D. Rubin, *Multiple Imputation for Nonresponse in Surveys*. Wiley, 1987.
[32] J.L. Schafer, *Analysis of Incomplete Multivariate Data*. Chapman and Hall/CRC, 1997.
[33] B. Silverman, *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, 1986.
[34] G.F. Smits and E.M. Jordaan, "Improved SVM Regression Using Mixtures of Kernels," *Proc. 2002 Int'l Joint Conf. Neural Networks*, pp. 2785-2790, 2002.
[35] Q.H. Wang and R. Rao, "Empirical Likelihood-Based Inference under Imputation for Missing Response Data," *Annals of Statistics*, vol. 30, pp. 896-924, 2002.
[36] S.C. Zhang et al., "Missing Is Useful: Missing Values in Cost-Sensitive Decision Trees," *IEEE Trans. Knowledge and Data Eng.*, vol. 17, no. 12, pp. 1689-1693, Dec. 2005.
[37] S.C. Zhang et al., "Information Enhancement for Data Mining," *IEEE Intelligent Systems*, vol. 19, no. 2, pp. 12-13, Mar./Apr. 2004.
[38] S.C. Zhang, "Parimputation: From Imputation and Null-Imputation to Partially Imputation," *IEEE Intelligent Informatics Bull.*, vol. 9, no. 1, pp. 32-38, Nov. 2008.
[39] S. Zhang, "Shell-Neighbor Method and Its Application in Missing Data Imputation," *Applied Intelligence*, doi: 10.1007/s10489-009-0207-6.
[40] W. Zhang, "Association Based Multiple Imputation in Multivariate Data Sets: A Summary," *Proc. Int'l Conf. Data Eng. (ICDE)*, p. 310, 2000.
[41] S. Zheng, J. Liu, and J. Tian, "An Efficient Star Acquisition Method Based on SVM with Mixtures of Kernels," *Pattern Recognition Letters*, vol. 26, pp. 147-165, 2005.
[42] C. Zhang, X. Zhu, J. Zhang, Y. Qin, and S. Zhang, "GBKII: An Imputation Method for Missing Values," *Proc. 11th Pacific-Asia Knowledge Discovery and Data Mining Conf. (PAKDD '07)*, pp. 1080-1087, 2007.
[43] J. Hausman, B.H. Hall, and Z. Griliches, "Econometric Models for Count Data with an Application to the Patents-R&D Relationship," *Econometrica*, vol. 52, no. 4, pp. 909-938, 1984.

Xiaofeng Zhu's bio and photo are not available.

Shichao Zhang received the PhD degree in computer science from Deakin University, Australia. He is a distinguished professor and the director of the Institute of Computer Software and Theory at the Zhejiang Normal University, Jinhua, China. His research interests include information quality and pattern discovery. He has published about 60 international journal papers and more than 60 international conference papers. He has won more than 10 nation class grants. He served/is serving as an editor-in-chief for the *International Journal of Information Quality and Computing*, an associate editor for the *IEEE Transactions on Knowledge and Data Engineering*, *Knowledge and Information Systems*, and the *IEEE Intelligent Informatics Bulletin*. He is a senior member of the IEEE and the IEEE Computer Society and a member of the ACM.

Zhi Jin received the MS and PhD degrees in computer science from Changsha Institute of Technology, China, in 1987 and 1992, respectively. She is currently a professor of computer science at Peking University, Beijing, China. Before joined Peking University, she has been a professor at the Academy of Mathematics and System Science at the Chinese Academy of Sciences since 2001. Her research interests include software requirements engineering and knowledge engineering. She has published a coauthored 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 more than 10 national competitive grants, including three China NSF grants, two China 973 program grants, and two China 863 program grants. She is a standing senior member of the China Computer Federation (CCF) and a grant review panelist for China NSF (Information Science Division). She is serving as an executive editor-in-chief for the *Journal of Software*, an editorial board member for the *Expert Systems*, and *Chinese Journal of Computers*; and served as a PC cochair, area chair, or PC member for various conferences. She is a senior member of the IEEE and the IEEE Computer Society.

Zili Zhang's bio and photo are not available.

Zhuoming Xu received the bachelor's and master's degrees in computer science from Hohai University, China, in 1986 and 1994, respectively, and the PhD degree in computer science from Southeast University, China, in 2005. He is currently a professor of computer science in the College of Computer and Information at Hohai University, where he has been the deputy director of Science and Technology Office since 2007. He was the chairman of the Department of Computer Science and Engineering at Hohai University from 2004 to 2006. He is also a member of the council of China Computer Federation. His research interests include databases, ontological engineering, and the semantic web.