Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 538613, 14 pages
http://dx.doi.org/10.1155/2015/538613

Research Article

Missing Values and Optimal Selection of an Imputation Method and Classification Algorithm to Improve the Accuracy of Ubiquitous Computing Applications

Jaemun Sim,[1] Jonathan Sangyun Lee,[2] and Ohbyung Kwon[2]

[1] *SKKU Business School, Sungkyunkwan University, Seoul 110734, Republic of Korea*
[2] *School of Management, Kyung Hee University, Seoul 130701, Republic of Korea*

Correspondence should be addressed to Ohbyung Kwon; obkwon@khu.ac.kr

Received 18 June 2014; Revised 29 September 2014; Accepted 11 October 2014

Academic Editor: Jong-Hyuk Park

In a ubiquitous environment, high-accuracy data analysis is essential because it affects real-world decision-making. However, in the real world, user-related data from information systems are often missing due to users' concerns about privacy or lack of obligation to provide complete data. This data incompleteness can impair the accuracy of data analysis using classification algorithms, which can degrade the value of the data. Many studies have attempted to overcome these data incompleteness issues and to improve the quality of data analysis using classification algorithms. The performance of classification algorithms may be affected by the characteristics and patterns of the missing data, such as the ratio of missing data to complete data. We perform a concrete causal analysis of differences in performance of classification algorithms based on various factors. The characteristics of missing values, datasets, and imputation methods are examined. We also propose imputation and classification algorithms appropriate to different datasets and circumstances.

1. Introduction

Ubiquitous computing has been the central focus of research and development in many studies; it is considered to be the third wave in the evolution of computer technology [1]. In ubiquitous computing, data must be collected and analyzed accurately in real time. For this process to be successful, data must be well organized and uncorrupted. Data preprocessing is an essential but time- and effort-consuming step in the process of data mining. Several preprocessing methods have been developed to overcome data inconsistencies [2].

Data incompleteness due to missing values is very common in datasets collected in real settings [3]; it presents a challenge in the data preprocessing phase. Data is often missing when user input is required. For example, in human-centric computing, systems often require user profile data for the purpose of personalization [4]. In the case of Twitter, text data is used for sentiment analysis in order to analyze user behaviors and attitudes [5]. As a final example, in ubiquitous commerce, customer data has been used to personalize services for users [6]. Values may be missing when users are reluctant to provide their personal data due to privacy concerns or lack of motivation. This is especially true for optional data requested by the system.

Missing values can also be present in sensor data. Sensor data is usually in quantitative form. Sensors provide physical information regarding temperature, sound, or trajectory. Sensor technology has advanced over the years; it is an essential source of data for ubiquitous computing and is used for situation awareness and circumstantial decision-making. For example, human interaction sensors read and react to current situations [7]. Analysis of image files for face recognition and object detection using sensors is widely used in ubiquitous computing [8]. However, incorrect data and missing values are possible even using advanced sensor technology due to mechanical and network errors. Missing values can interfere with decision-making and personalization, which can ultimately lead to user dissatisfaction. In many cases, the impact of missing data is costly to users of data analysis methods such as classification algorithms.

Data incompleteness may have negative effects on data preprocessing and decision-making accuracy. Extra time and effort are required to compensate for missing data. Using uncertain or null data results in fatal errors in the classification algorithm, and deleting all records that contain missing data (i.e., using the listwise deletion method) reduces the sample size, which might decrease statistical power and introduce potential bias to the estimation [9]. Finally, unless the researcher can be sure that the data values are missing completely at random (MCAR), then the conclusions resulting from a complete-case analysis are most likely to be biased.

In order to overcome issues related to data incompleteness, many researchers have suggested methods of supplementing or compensating for missing data. The missing data imputation method is the most frequently used statistical method developed to deal with missing data problems. It is defined as "a procedure that replaces the missing values in a dataset by some plausible values" [3]. Missing values occur when no data is stored for a given variable in the current observation.

Many studies have attempted to validate the missing data imputation method of supplementing or compensating for missing data by testing it with different types of data; other studies have attempted to develop the method further. Studies have also compared the performance of various imputation methods based on benchmark data. For example, Kang investigated the ratio of missing to complete data in various datasets and compared the average accuracy of several imputation methods, such as MNR, k-NN, CART, ANN, and LLR [10]. The results demonstrated that k-NN performed best on datasets with less than 10% of data missing and LLR performed best on those with more than 10% of data missing.

However, after multiple tests using complete datasets, not much difference in performance was observed, and some datasets were linearly inferior. In Kang's study [10], many datasets with equivalent conditions yielded different results. Thus, the fit between the dataset characteristics and the imputation method must also be considered. Previous studies have compared imputation methods by varying the ratio of missing to complete data or evaluating performance differences between complete and incomplete datasets. However, the reasons for these different results between datasets under equivalent conditions remain unexplained. Various factors may affect the performance of classification algorithms. For example, the interrelationship or fitness between the dataset, imputation method, and characteristics of the missing values may be important to the success or failure of the analytical process.

The purpose of this study is to examine the influence of dataset characteristics and patterns of missing data on the performance of classification algorithms using various datasets. The moderating effects of different imputation methods, classification algorithms, and data characteristics on performance are also analyzed. The results are important because they can suggest which imputation method or classification algorithm to use depending on the data conditions. The goal is to improve the performance, accuracy, and time required for ubiquitous computing.

2. Treating Datasets Containing Missing Data

Missing information is an unavoidable aspect of data analysis. For example, responses may be missing to items on survey instruments intended to measure cognitive and affective factors. Various imputation methods have been developed and used for treatment of datasets containing missing data. Some popular methods are listed below.

(1) Listwise Deletion. Listwise deletion (LD) involves the removal of all individuals with incomplete responses for any items. However, LD reduces the effective sample size (sometimes greatly, resulting in large amounts of missing data), which can, in turn, reduce statistical power for hypothesis testing to unacceptably low levels. LD assumes that the data are MCAR (i.e., their omission is unrelated to all measured variables). When the MCAR assumption is violated, as is often the case in real research settings, the resulting estimates will be biased.

(2) Zero Imputation. When data are omitted as incorrect, the zero imputation method is used, in which missing responses are assigned an incorrect value (or zero in the case of dichotomously scored items).

(3) Mean Imputation. In this method, the mean of all values within the same attribute is calculated and then imputed in the missing data cells. The method works only if the attribute examined is not nominal.

(4) Multiple Imputations. Multiple imputations can incorporate information from all variables in a dataset to derive imputed values for those that are missing. This method has been shown to be an effective tool in a variety of scenarios involving missing data [11], including incomplete item responses [12].

(5) Regression Imputation. The linear regression function is calculated from the values within the same attribute and then used as the dependent variable. The other attributes (except the decision attribute) are then used as independent variables. Then the estimated dependent variable is imputed in the missing data cells. This method works only if all considered attributes are not nominal.

(6) Stochastic Regression Imputation. Stochastic regression imputation involves a two-step process in which the distribution of relative frequencies for each response category for each member of the sample is first obtained from the observed data.

In this paper, the details of the seven imputation methods used herein are as follows.

(i) Listwise Deletion. All instances are deleted that contain more than one missing cell in their attributes.

(ii) Mean Imputation. The missing values from each attribute (column or feature) are replaced with the mean of all known values of that attribute. That is, let X_i^j be the jth missing attribute of the ith instance, which is imputed by

$$X_i^j = \sum_{k \in I(\text{complete})} \frac{X_k^j}{n_{|I(\text{complete})|}}, \tag{1}$$

where $I(\text{complete})$ is a set of indices that are not missing in X_i and $n_{|I(\text{complete})|}$ is the total number of instances where the jth attribute is not missing.

(iii) Group Mean Imputation. The process for this method is the same as that for mean imputation. However, the missing values are replaced with the group (or class) mean of all known values of that attribute. Each group represents a target class from among the instances (recorded) that have missing values. Let $X_{m,i}^j$ be the jth missing attribute of the ith instance of the mth class, which is imputed by

$$X_{m,i}^j = \sum_{k \in I(m\text{th class incomplete})} \frac{X_{m,k}^j}{n_{|I(m\text{th class incomplete})|}}, \tag{2}$$

where $I(m\text{th class incomplete})$ is a set of indices that are not missing in $X_{m,i}^j$ and $n_{|I(m\text{th class incomplete})|}$ is the total number of instances where the jth attribute of the mth class is not missing.

(iv) Predictive Mean Imputation. In this method, the functional relationship between multiple input variables and single or multiple target variables of the given data is represented in the form of a linear equation. This method sets attributes that have missing values as dependent variables and other attributes as independent variables in order to allow prediction of missing values by creating a regression model using those variables. For a regression target y_i, the MLR equation with d predictors and n training instances can be written as

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_d x_{id} \quad \text{for } i = 1, \ldots, n. \tag{3}$$

This can be rewritten in matrix form such that $y = X\beta$, and the coefficient β can be obtained explicitly by taking a derivative of the squared error function as follows:

$$\begin{gathered} \min E(\beta) = \frac{1}{2}(y - X\beta)^T (y - X\beta), \\ \frac{\partial E(\beta)}{\partial \beta} = X^T X\beta - X^T y = 0, \\ \beta = \left(X^T X\right)^{-1} \cdot X^T y. \end{gathered} \tag{4}$$

(v) Hot-Deck. This method is the same in principle as case-based reasoning. In order for attributes that contain missing values to be utilized, values must be found from among the most similar instances of nonmissing values and used to replace the missing values. Therefore, each missing value is replaced with the value of an attribute with the most similar instance as follows:

$$X_i^j = X_k^j, \quad k = \arg\min_P \sqrt{\sum_{j \in I(\text{complete})} \text{Std}_j \left(X_i^j - X_P^j\right)^2}, \tag{5}$$

where Std_j is the standard deviation of the jth attribute which is not missing.

(vi) k-NN. Attributes are found via a search among nonmissing attributes using the 3-NN method. Missing values are imputed based on the values of the attributes of the k most similar instances as follows:

$$X_i^j = \sum_{P \in k\text{-NN}(X_i)} k\left(X_i^{I(\text{complete})}, X_P^{I(\text{complete})}\right) \cdot X_P^j, \tag{6}$$

where k-NN(X_i) is the index set of the kth nearest neighbors of X_i based on the nonmissing attributes and $k(X_i, X_j)$ is a kernel function that is proportional to the similarity between the two instances X_i and X_j ($k = 4$).

(vii) k-Means Clustering. Attributes are found through formation of k-clusters from nonmissing data, after which missing values are imputed. The entire dataset is partitioned into k clusters by maximizing the homogeneity within each cluster and the heterogeneity between clusters as follows:

$$\arg \min_{C^{h(\text{complete})}} \sum_{i=1}^{k} \sum_{X_j^{I(\text{complete})} \in C_i^{h(\text{complete})}} \left\| X_j^{I(\text{complete})} - C_i^{I(\text{complete})} \right\|^2, \tag{7}$$

where $C_i^{I(\text{complete})}$ is the centroid of $C_i^{I(\text{complete})}$ and $C^{I(\text{complete})}$ is the union of all clusters ($C^{I(\text{complete})} = C_1^{I(\text{complete})} \cup \cdots \cup C_k^{I(\text{complete})}$). For a missing value X_i^j, the mean value of the attribute for the instances in the same cluster with $X_i^{I(\text{complete})}$ is imputed thus as follows:

$$\begin{gathered} X_i^j = \frac{1}{\left|C_k^{I(\text{complete})}\right|} \cdot \sum_{X_P^{I(\text{complete})} \in C_k^{I(\text{complete})}} X_P^j \\ \text{s.t. } k = \arg\min_i \left| X_j^{I(\text{complete})} - C_i^{I(\text{complete})} \right|. \end{gathered} \tag{8}$$

3. Model

In this paper, we hypothesize an association between the performance of classification algorithms and the characteristics of missing data and datasets. Moreover, we assume that the chosen imputation method moderates the causality between these factors. Figure 1 illustrates the posited relationships.

3.1. Missing Data Characteristics. Table 1 describes the characteristics of missing data and how to calculate them. The pattern of missing data characteristics may be univariate, monotone, or arbitrary [11]. A univariate pattern of missing data occurs when missing values are observed for a single variable only; all other data are complete for all variables.

Table 1: The characteristics of missing data.

Variables	Meaning	Calculation
Missing data ratio	The number of missing values in the entire dataset as compared to the number of nonmissing values	The number of empty data cells/total cells
Patterns of missing data	Univariate Monotone Arbitrary	Ratio of missing to complete values for an existing feature compared to the values for all features
Horizontal scatteredness	Distribution of missing values within each data record	Determine the number of missing cells in each record and calculate the standard deviation
Vertical scatteredness	Distribution of missing values for each attribute	Determine the number of missing cells in each feature and calculate the standard deviation
Missing data spread	Larger standard deviations indicate stronger effects of missing data	Determine the weighted average of the standard deviations of features with missing data (weight: the ratio of missing to complete data for each feature)

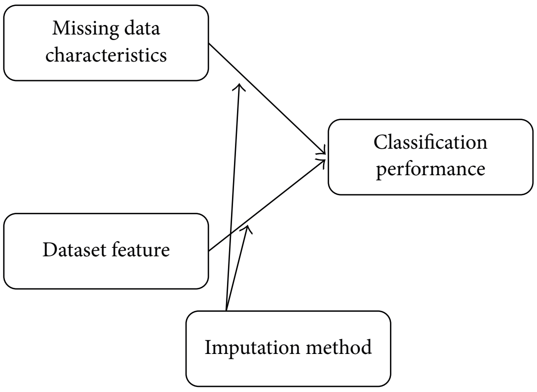


Figure 1: Research model.

A monotone pattern occurs if variables can be arranged such that all $Y_{j+1}, \ldots, Y_k$ are missing for cases where Y_j is missing. Another characteristic, missing data spread, is important because larger standard deviations for missing values within an existing feature indicate that the missing data has greater influence on the results of the analysis (Figure 2).

3.2. Dataset Features. Table 2 lists the features of datasets. Based on the research of Kwon and Sim [15], in which characteristics of datasets that influence classification algorithms were identified, we considered the following statistically significant features in this study: missing values, the number of cases, the number of attributes, and the degree of class imbalance. However, the discussion of missing values is omitted here because it has already been analyzed in detail by Kwon and Sim [15].

3.3. Imputation Methods. Table 3 lists the imputation methods used in this study. Since datasets with categorical decision attributes are included, imputation methods that do not accommodate categorical attributes (e.g., regression imputation) are excluded from this paper.

Table 2: Dataset features.

Variables	Description
Number of cases	Number of records in the dataset
Number of attributes	Number of features characteristic of the dataset
Degree of class imbalance	Ratio

3.4. Classification Algorithms. Many studies have compared classification algorithms in various areas. For example, the decision tree is known as the best algorithm for arrhythmia classification [16]. In Table 4, six types of representative classification algorithms for supervised learning are described: C4.5, SVM (support vector machine), Bayesian network, logistic classifier, k-nearest neighbor classifier, and regression.

4. Method

We conducted a performance evaluation of the imputation methods and classification algorithms described in the previous section using actual datasets taken from the UCI dataset archive. To ensure the accuracy of each method in cases with no missing values, datasets with missing values were not included. Among the selected datasets, six (Iris, Wine, Glass, Liver Disorder, Ionosphere, and Statlog Shuttle) were included for comparison with the results of Kang [10]. These datasets are popular and frequently utilized benchmarks in the literature, which makes them useful for demonstrating the superiority of the proposed idea.

Table 5 provides the names of the datasets, the numbers of cases, and the descriptions of features and classes. The numbers in parentheses in the last two columns represent the number of features and classes for the decision attributes. For example, in dataset Iris, "Numeric (4)" indicates that there are four numeric attributes, and "Categorical (3)" means that there are three classes in the decision attribute.

Since UCI datasets have no missing data, target values in each dataset were randomly omitted [10]. Based on

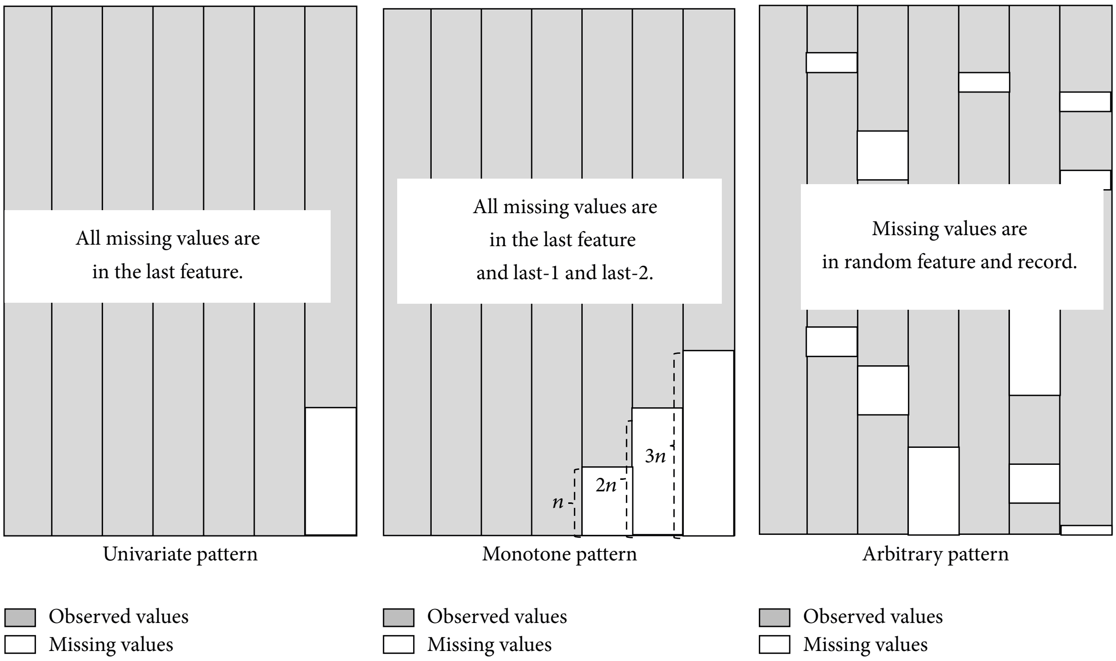


FIGURE 2: Missing data patterns.

TABLE 3: Imputation methods.

Imputation methods	Description
Listwise deletion	Perhaps the most basic traditional technique for dealing with missing data. Cases with missing values are discarded, restricting the analyses to cases for which complete data are available.
Mean imputation	Involves replacing missing data with the overall mean for the observed data.
Group mean imputation	A missing value is replaced by the mean of a subset of the data, based on other observed variable(s) in the data.
Predictive mean imputation	Also called regression imputation. Predictive mean imputation involves imputing a missing value using an ordinary least-squares regression method to estimate missing data.
Hot-deck	Most similar records are imputed to missing values.
k-NN	The attribute value of k is imputed to the most similar instance from nonmissing data.
k-means clustering	k numbers of sets are created that are homogeneous on the inside and heterogeneous on the outside.

TABLE 4: Classification algorithms.

Algorithms	Description
C4.5	Estimates the known data using learning rules. C4.5 gradually expands the conditions of the algorithm, splitting the upper node into subnodes using a divide-and-conquer method until it comes to the end node.
SVM	Classifies the unknown class by finding the optimal hyperplane with the maximum margin that reduces the estimation error.
Bayesian network	A probability network with a high posterior probability given the instances. Such a network can provide insight into probabilistic dependencies among the variables in the training dataset.
Logistic classifier	Takes the functional form of logistic CDF (cumulative distribution function). This function relates the probability of some event to attribute variables through regression coefficients and alpha and beta parameters, which are estimated from training data [13].
k-nearest neighbor classifier	Simple instance-based learner that uses the class of the nearest k training instances for the class of the test instances.
Regression	The class is binarized, and one regression model is built for each class value [14].

Table 5: Datasets used in the experiments.

Dataset	Number of cases	Features	Decision attributes
Iris	150	Numeric (4)	Categorical (3)
Wine	178	Numeric (13)	Categorical (3)
Glass	214	Numeric (9)	Categorical (7)
Liver disorder	345	Numeric (6)	Categorical (2)
Ionosphere	351	Numeric (34)	Categorical (2)
Statlog Shuttle	57,999	Numeric (7)	Categorical (7)

the list of missing data characteristics, three datasets with three different missing data ratios (5%, 10%, and 15%) and three sets representing each of the missing data patterns (univariate, monotone, and arbitrary) were created for a total of nine variations for each dataset. In total, 54 datasets were imputed for each imputation method, as 6 datasets were available. We repeated the experiment for each dataset 1000 times in order to minimize errors and bias. Thus, 5,400 datasets were imputed in total for our experiment. All imputation methods were implemented using packages written in Java. In order to measure the performance of each imputation method, we applied imputed datasets to the six classification algorithms listed in Table 4.

There are various indicators to measure performance, such as accuracy, relative accuracy, MAE (mean absolute error), and RMSE (root mean square error). However, RMSE is one of the most representative and widely used performance indicators in the imputation research. Therefore, we also adopted RMSE as the performance indicator in this study. The performance of the selected classification algorithms was evaluated using SPSS 17.0.

RMSE measures the difference between predicted and observed values. The term "relative prediction accuracy" refers to the relative ratio of accuracy, which is equivalent to 1 when there are no missing data [10]. The no-missing-data condition was used as a baseline of performance. As the next step, we generated a missing dataset from the original no-missing-dataset and then applied an imputation method to replace the null data. Then a classification algorithm was conducted to estimate the results of the imputed dataset. With all combinations of imputation methods and classification algorithms, a multiple regression analysis was conducted using the following equation to understand the input factors, the characteristics of missing data, and those of the datasets, in order to determine how the selected classification algorithms affected performance:

$$y_p = \sum_{\forall j \in \mathrm{M}} \beta_{pj} x_j + \sum_{\forall k \in \mathrm{D}} \chi_{pk} z_k + \varepsilon_p. \tag{9}$$

In this equation, x is the value of the characteristics of the missing data (M), z is the value of each dataset's characteristics in the set of dataset (D), and y is a performance parameter. Note that M = {missing data ratio, patterns of missing data, horizontal scatteredness, vertical scatteredness, missing data spread} and D = {number of cases, number of attributes, degree of class imbalance}. In addition, $p = 1$ indicates relative prediction accuracy, $p = 2$ represents RMSE, and $p = 3$ means elapsed time. We performed the experiment using the Weka library source software (release 3.6) to determine the reliability of the implementation of the algorithms [17]. We did not use the Weka GUI tool but developed a Weka library-based performance evaluation program in order to conduct the automatized experiment repeatedly.

5. Results

In total, 32,400 datasets (3 missing ratios × 3 imputation patterns × 6 imputation methods × 100 trials) were imputed for each of the 6 classifiers. Thus, in total, we tested 226,800 datasets (32,400 imputed dataset × 7 classifier methods). The results were divided by those for each dataset, classification algorithm, and imputation method for comparison in terms of performance.

5.1. Datasets. Figure 3 shows the performance of each imputation method for the six different datasets. On the x-axis, three missing ratios represent the characteristics of missing data, and on the y-axis, performance is indicated using the RMSE. All results of three different variations of the missing data patterns and tested classification algorithms were merged for each imputation method.

For Iris data (Figure 3(a)), the mean imputation method yielded the worst results and the group mean imputation method the best results.

For Glass Identification data (Figure 3(b)), hot-deck imputation was the least effective method and predictive mean imputation was the best.

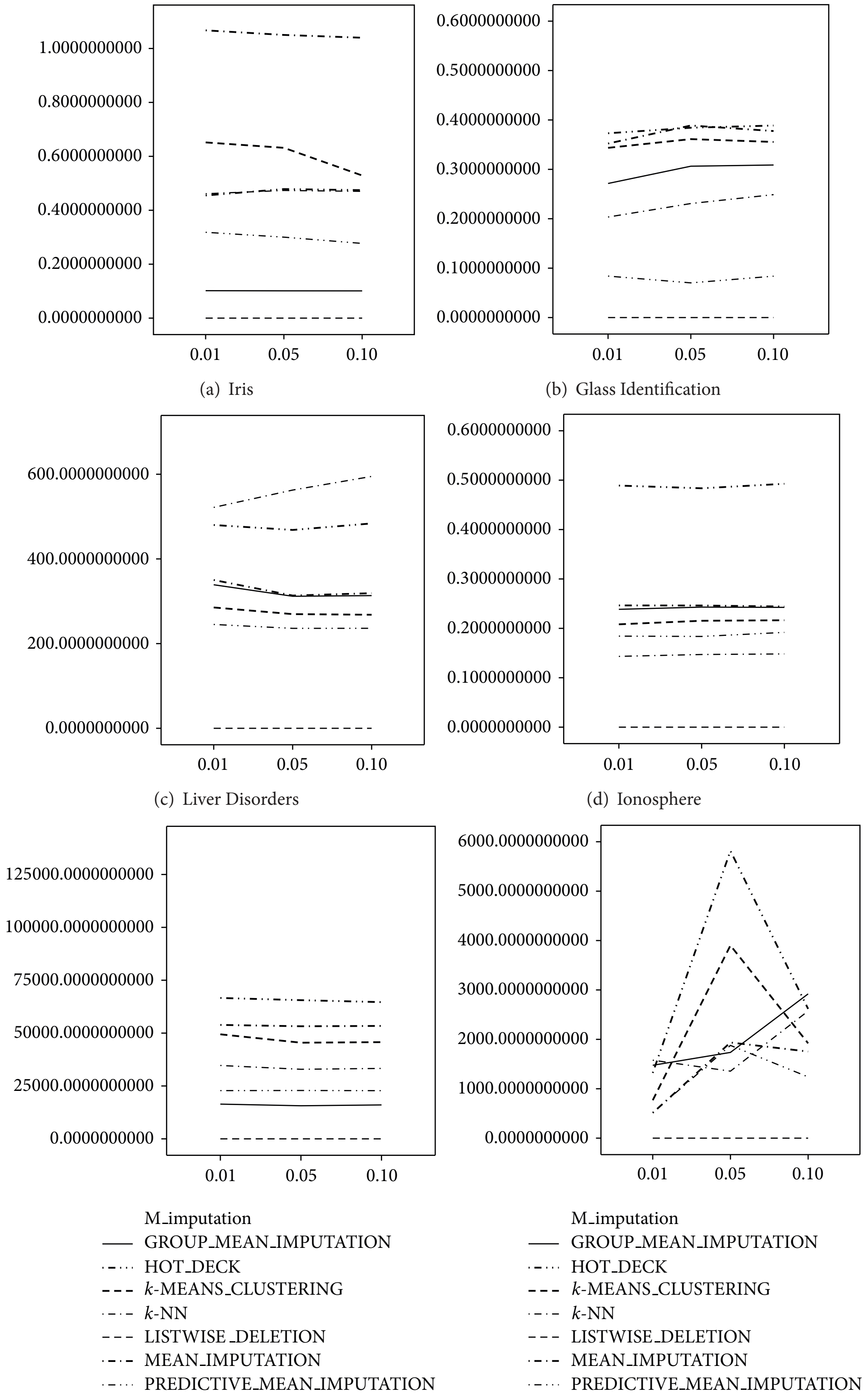
For Liver Disorder data (Figure 3(c)), k-NN was the least effective, and once again, the predictive mean imputation method yielded the best results.

For Ionosphere data (Figure 3(d)), hot-deck was the worst and k-NN the best.

For Wine data (Figure 3(e)), hot-deck was once again the least effective method, and predictive mean imputation the best.

For Statlog data (Figure 3(f)), unlike the other datasets, the results varied based on the missing data ratio. However, predictive mean imputation was still the best method overall and hot-deck the worst.

Figure 3 illustrates that the predictive mean imputation method yielded the best results overall and hot-deck imputation the worst. However, no imputation method was generally superior in all cases with any given dataset. For example, the k-NN method yielded the best performance for the Ionosphere dataset, but for the Liver Disorders dataset, its performance was lowest. In another example, the group mean imputation method performed best for the Iris and Wine datasets, but its performance was only average for other datasets. Therefore, the results were inconsistent, and determining the best imputation method is impossible. Thus, the imputation method cannot be used as an accurate predictor of performance. Rather, the performance must be influenced by other factors, such as the interaction between the characteristics of the dataset in terms of missing data and the chosen imputation method.



(a) Iris

(b) Glass Identification

(c) Liver Disorders

(d) Ionosphere

(e) Wine

(f) Statlog Shuttle

Figure 3: Comparison of performances of imputation methods for each dataset.

Table 6: Factors influencing accuracy (RMSE) for each algorithm (standard beta coefficient): mean imputation.

Data characteristic	trees.J48	BayesNet	SMO	Regression	Logistic	IBk
N_attributes	$-.076^{**}$	$-.075^{**}$	$-.178^{**}$	$-.072^{**}$	$.115^{**}$	.007
N_cases	$-.079^{**}$	$-.049^{**}$	.012	−.017	−.032	$-.048^{**}$
C_imbalance	$.117^{**}$	$.239^{**}$	$.264^{**}$	$.525^{**}$	$.163^{**}$	$.198^{**}$
R_missing	$.051^{*}$	$.078^{**}$	.040	$.080^{**}$	$.076^{**}$	$.068^{**}$
SE_HS	$.249^{**}$	$.285^{**}$	$.186^{**}$	$.277^{**}$	$.335^{**}$	$.245^{**}$
SE_VS	−.009	−.013	−.006	−.013	−.016	−.010
Spread	$-.382^{**}$	$-.430^{**}$	$-.261^{**}$	$-.436^{**}$	$-.452^{**}$	$-.363^{**}$
P_missing_dum1	−.049	−.038	−.038	−.037	−.045	−.038
P_missing_dum2	−.002	.014	.002	.011	.001	.011

Note 1: N_attributes: number of attributes, N_cases: number of cases, C_imbalance: degree of class imbalance, R_missing: missing data ratio, SE_HS: horizontal scatteredness, SE_VS: vertical scatteredness, spread: missing data spread, and missing patterns: univariate (P_missing_dum1 = 1, P_missing_dum2 = 0), monotone (P_missing_dum1 = 0, P_missing_dum2 = 1), and arbitrary (P_missing_dum1 = 1, P_missing_dum2 = 1)
Note 2: RMSE indicates error; therefore, lower values are better.
Note 3: $^{*}P < 0.05$, $^{**}P < 0.01$.

Table 7: Factors influencing accuracy (RMSE) for each algorithm (standard beta coefficient): group mean imputation.

Data characteristic	trees.J48	BayesNet	SMO	Regression	Logistic	IBk
N_attributes	$-.068^{**}$	$-.072^{**}$	$-.179^{**}$	$-.068^{**}$	$.115^{**}$	.010
N_cases	$-.082^{**}$	$-.050^{**}$	.011	−.018	$-.034^{*}$	$-.047^{**}$
C_imbalance	$.115^{**}$	$.228^{**}$	$.260^{**}$	$.517^{**}$	$.156^{**}$	$.197^{**}$
R_missing	$.050^{**}$	$.085^{**}$	.043	$.084^{**}$	$.095^{**}$	$.066^{**}$
SE_HS	$.230^{**}$	$.268^{**}$	$.178^{**}$	$.273^{**}$	$.300^{**}$	$.248^{**}$
SE_VS	−.008	−.012	−.006	−.013	−.013	−.010
Spread	$-.296^{**}$	$-.439^{**}$	$-.264^{**}$	$-.443^{**}$	$-.476^{**}$	$-.382^{**}$
P_missing_dum1	−.043	−.032	−.034	−.035	−.035	−.041
P_missing_dum2	.002	.024	.004	.016	.021	.013

Note 1: N_attributes: number of attributes, N_cases: number of cases, C_imbalance: degree of class imbalance, R_missing: missing data ratio, SE_HS: horizontal scatteredness, SE_VS: vertical scatteredness, spread: missing data spread, and missing patterns: univariate (P_missing_dum1 = 1, P_missing_dum2 = 0), monotone (P_missing_dum1 = 0, P_missing_dum2 = 1), and arbitrary (P_missing_dum1 = 1, P_missing_dum2 = 1)
Note 2: RMSE indicates error; therefore, lower values are better.
Note 3: $^{*}P < 0.05$, $^{**}P < 0.01$.

5.2. Classification Algorithm. Figure 4 shows the performance of the classification algorithms by imputation method and ratio of missing data. As shown in the figure, the performance of each imputation method was similar and did not vary depending on the ratio of missing data, except for listwise deletion. For listwise deletion, as the ratio of missing to complete data increased, the performance deteriorated. In the listwise deletion method, all records are deleted that contain missing data; therefore, the number of deleted records increases as the ratio of missing data increases. The low performance of this method can be explained based on this fact.

The differences in performance between imputation methods were minor. The figure displays these differences by classification algorithm. Using the Bayesian network and logistic classifier methods significantly improved performance compared to other classifiers. However, the relationships among missing data, imputation methods, and classifiers remained to be explained. Thus, a regression analysis was conducted.

In Figure 4, the results suggest the following rules.

(i) IF the missing rate increases AND IBK is used, THEN use the GROUP_MEAN_IMPUTATION method.

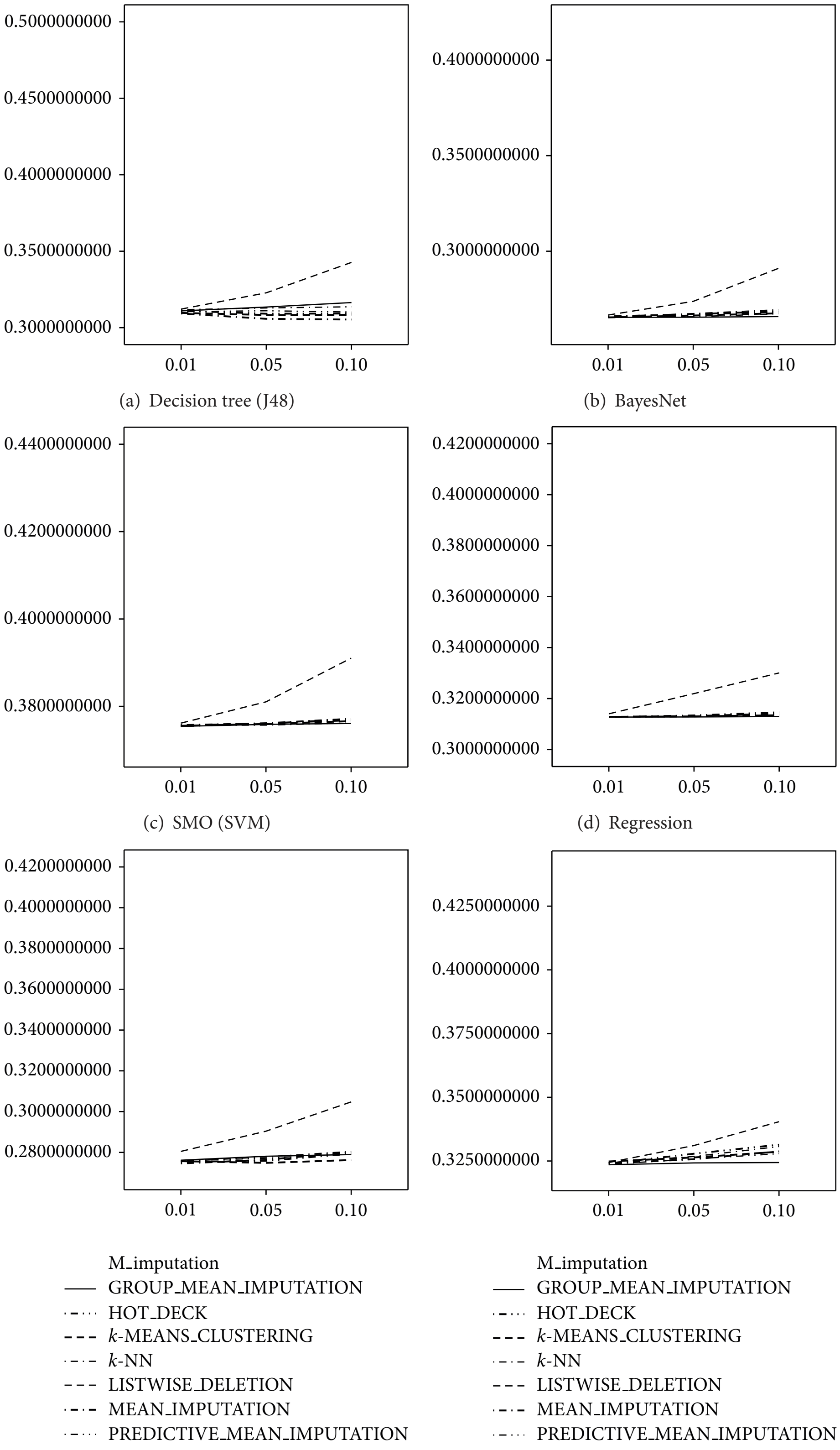
(ii) IF the missing rate increases AND the logistic classifier method is used, THEN use the HOT_DECK method.

(iii) IF the missing rate increases AND the regression method is used, THEN use the GROUP_MEAN_IMPUTATION method.

(iv) IF the missing rate increases AND the BayesNet method is used, THEN use the GROUP_MEAN_IMPUTATION method.

(v) IF the missing rate increases AND the trees.J48 method is used, THEN use the k-NN method.

5.3. Regression. The results of the regression analysis are presented in Tables 6, 7, 8, 9, 10, and 11. The analysis was conducted using 900 datasets (3 missing ratios × 3 missing



(a) Decision tree (J48)

(b) BayesNet

(c) SMO (SVM)

(d) Regression

(e) Logistic

(f) IBk (*k*-nearest neighbor classifier)

Figure 4: Comparison of classifiers in terms of classification performance.

Table 8: Factors influencing accuracy (RMSE) for each algorithm (standard beta coefficient): Predictive_Mean_Imputation.

Data characteristic	trees.J48	BayesNet	SMO	Regression	Logistic	IBk
N_attributes	$-.076^{**}$	$-.076^{**}$	$-.178^{**}$	$-.063^{**}$	$.123^{**}$	.016
N_cases	$-.084^{**}$	$-.049^{**}$	.012	−.017	$-.034^{*}$	$-.047^{**}$
C_imbalance	$.117^{**}$	$.242^{**}$	$.263^{**}$	$.523^{**}$	$.153^{**}$	$.198^{**}$
R_missing	$.050^{*}$	$.079^{**}$	.043	$.085^{**}$	$.080^{**}$	$.068^{**}$
SE_HS	$.223^{**}$	$.279^{**}$	$.182^{**}$	$.268^{**}$	$.322^{**}$	$.242^{**}$
SE_VS	−.008	−.013	−.006	−.013	−.015	−.009
Spread	$-.328^{**}$	$-.432^{**}$	$-.262^{**}$	$-.434^{**}$	$-.465^{**}$	$-.361^{**}$
P_missing_dum1	−.042	−.035	−.034	−.028	−.044	−.036
P_missing_dum2	.008	.012	.004	.018	.007	.011

Note 1: N_attributes: number of attributes, N_cases: number of cases, C_imbalance: degree of class imbalance, R_missing: missing data ratio, SE_HS: horizontal scatteredness, SE_VS: vertical scatteredness, spread: missing data spread, and missing patterns: univariate (P_missing_dum1 = 1, P_missing_dum2 = 0), monotone (P_missing_dum1 = 0, P_missing_dum2 = 1), and arbitrary (P_missing_dum1 = 1, P_missing_dum2 = 1)
Note 2: RMSE indicates error; therefore, lower values are better.
Note 3: $^{*}P < 0.05$, $^{**}P < 0.01$.

Table 9: Factors influencing accuracy (RMSE) for each algorithm (standard beta coefficient): Hot_deck.

Data characteristic	trees.J48	BayesNet	SMO	Regression	Logistic	IBk
N_attributes	$-.080^{**}$	$-.073^{**}$	$-.176^{**}$	$-.071^{**}$	$.115^{**}$	.007
N_cases	$-.081^{**}$	$-.049^{**}$	.012	−.018	$-.034^{*}$	$-.047^{**}$
C_imbalance	$.135^{**}$	$.237^{**}$	$.261^{**}$	$.524^{**}$	$.133^{**}$	$.211^{**}$
R_missing	$.062^{**}$	$.083^{**}$	.044	$.084^{**}$	$.075^{**}$	$.070^{**}$
SE_HS	$.225^{**}$	$.275^{**}$	$.183^{**}$	$.271^{**}$	$.313^{**}$	$.254^{**}$
SE_VS	−.009	−.013	−.006	−.013	−.014	−.010
Spread	$-.365^{**}$	$-.428^{**}$	$-.265^{**}$	$-.427^{**}$	$-.441^{**}$	$-.361^{**}$
P_missing_dum1	−.035	−.037	−.034	−.033	−.048	−.038
P_missing_dum2	.012	.015	.004	.012	−.004	.009

Note 1: N_attributes: number of attributes, N_cases: number of cases, C_imbalance: degree of class imbalance, R_missing: missing data ratio, SE_HS: horizontal scatteredness, SE_VS: vertical scatteredness, spread: missing data spread, and missing patterns: univariate (P_missing_dum1 = 1, P_missing_dum2 = 0), monotone (P_missing_dum1 = 0, P_missing_dum2 = 1), and arbitrary (P_missing_dum1 = 1, P_missing_dum2 = 1)
Note 2: RMSE indicates error; therefore, lower values are better.
Note 3: $^{*}P < 0.05$, $^{**}P < 0.01$.

Table 10: Factors influencing accuracy (RMSE) for each algorithm (standard beta coefficient): k-NN.

Data characteristic	trees.J48	BayesNet	SMO	Regression	Logistic	IBk
N_attributes	$-.085^{**}$	$-.079^{**}$	$-.181^{**}$	$-.068^{**}$	$.122^{**}$	.006
N_cases	$-.083^{**}$	$-.049^{**}$	.011	−.018	$-.034^{*}$	$-.047^{**}$
C_imbalance	$.143^{**}$	$.249^{**}$	$.260^{**}$	$.521^{**}$	$.152^{**}$	$.211^{**}$
R_missing	$.054^{*}$	$.078^{**}$	.041	$.085^{**}$	$.075^{**}$	$.071^{**}$
SE_HS	$.234^{**}$	$.290^{**}$	$.182^{**}$	$.269^{**}$	$.328^{**}$	$.255^{**}$
SE_VS	−.010	−.013	−.006	−.013	−.014	−.011
Spread	$-.332^{**}$	$-.427^{**}$	$-.264^{**}$	$-.431^{**}$	$-.450^{**}$	$-.369^{**}$
P_missing_dum1	−.038	−.041	−.035	−.029	−.057	−.035
P_missing_dum2	.003	.008	.005	.017	.000	.011

Note 1: N_attributes: number of attributes, N_cases: number of cases, C_imbalance: degree of class imbalance, R_missing: missing data ratio, SE_HS: horizontal scatteredness, SE_VS: vertical scatteredness, spread: missing data spread, and missing patterns: univariate (P_missing_dum1 = 1, P_missing_dum2 = 0), monotone (P_missing_dum1 = 0, P_missing_dum2 = 1), and arbitrary (P_missing_dum1 = 1, P_missing_dum2 = 1)
Note 2: RMSE indicates error; therefore, lower values are better.
Note 3: $^{*}P < 0.05$, $^{**}P < 0.01$.

Table 11: Factors influencing accuracy (RMSE) for each algorithm (standard beta coefficient): k-MEANS_CLUSTERING.

Data characteristic	trees.J48	BayesNet	SMO	Regression	Logistic	IBk
N_attributes	$-.080^{**}$	$-.078^{**}$	$-.181^{**}$	$-.068^{**}$	$.117^{**}$	.009
N_cases	$-.079^{**}$	$-.049^{**}$	.012	−.017	−.033	$-.047^{**}$
C_imbalance	$.136^{**}$	$.240^{**}$	$.263^{**}$	$.524^{**}$	$.145^{**}$	$.206^{**}$
R_missing	$.057^{*}$	$.079^{**}$	.041	$.084^{**}$	$.079^{**}$	$.057^{*}$
SE_HS	$.236^{**}$	$.289^{**}$	$.183^{**}$	$.271^{**}$	$.315^{**}$	$.264^{**}$
SE_VS	−.009	−.013	−.006	−.013	−.014	−.011
Spread	$-.362^{**}$	$-.439^{**}$	$-.262^{**}$	$-.440^{**}$	$-.474^{**}$	$-.363^{**}$
P_missing_dum1	−.037	−.042	−.036	−.032	−.038	−.046
P_missing_dum2	.002	.013	.001	.014	.009	.004

Note 1: N_attributes: number of attributes, N_cases: number of cases, C_imbalance: degree of class imbalance, R_missing: missing data ratio, SE_HS: horizontal scatteredness, SE_VS: vertical scatteredness, spread: missing data spread, and missing patterns: univariate (P_missing_dum1 = 1, P_missing_dum2 = 0), monotone (P_missing_dum1 = 0, P_missing_dum2 = 1), and arbitrary (P_missing_dum1 = 1, P_missing_dum2 = 1)
Note 2: RMSE indicates error; therefore, lower values are better.
Note 3: $^{*}P < 0.05$, $^{**}P < 0.01$.

patterns × 100 trials). Each dataset was generated randomly to meet the preconditions. We conducted the performance evaluation by randomly assigning each dataset to test/training sets at a 3 : 7 ratio. The regression analysis included the characteristics of the datasets and the patterns of the missing values as independent variables. Control variables, such as the type of classifier and imputation method, were also included. The effects of the various characteristics of the data and missing values on classifier performance (RMSE) were analyzed. Three types of missing ratios were treated as two dummy variables (P_missing_dum1, 2: 00, 01, 10). Tables 6–11 illustrate the results of the regression analysis of the various imputation methods. The results suggest the following rules regardless of which imputation method is selected:

(i) IF N_attributes increases, THEN use SMO.

(ii) IF N_cases increases, THEN use trees.J48.

(iii) IF C_imbalance increases, THEN use trees.J48.

(iv) IF R_missing increases, THEN use SMO.

(v) IF SE_HS increases, THEN use SMO.

(vi) IF Spread increases, THEN use Logistic.

Figure 5 displays the coefficient pattern of the decision tree classifier for each imputation method. Dataset characteristics are illustrated on the x-axis and the regression coefficients for each imputation method on the y-axis. For all imputation methods except listwise deletion, the classifiers' coefficient patterns seemed similar. However, significant differences were found in the coefficient patterns using other algorithms. For example, for all imputation methods, a higher beta coefficient of the number of attributes (N_attributes) was observed for the logistics algorithm than for any other algorithm. Thus, the logistics algorithm exhibited the lowest performance (highest RMSE) in terms of the number of attributes. In terms of the number of cases (N_cases), SMO performed the worst. When the data were imbalanced, the regression method was the least effective one. For the missing ratio, the regression method showed the lowest performance except in comparison to listwise deletion and mean imputation. For the horizontal scattered standard error (SE_HS), SMO had the lowest performance. For missing data spread, the logistic classifier method had the lowest performance.

Moreover, for each single factor (e.g., spread), even if the results for two algorithms were the same, their performance differed depending on which imputation method was applied. For example, for the decision tree (J48) algorithm, the mean imputation method had the most negative effect on classification performance for horizontal scattered standard error (SE_HS) and spread, while the listwise deletion and group mean imputation methods had the least negative effect.

The similar coefficient patterns shown in Figure 5 indicate that the differences in impact of each imputation method on performance were insignificant. In order to determine the impact of the classifiers, more tests were needed. Figure 6 illustrates the coefficient patterns when the ratio of missing to complete data is 90%. Under these circumstances, the distinction between imputation methods according to dataset characteristics is significant. For example, very high or very low beta coefficients may be observed for most dataset characteristics except the number of instances and class imbalance.

Figure 7 shows the RMSE based on the ratio of missing data for each imputation method. As the ratio increases, the performance drops (RMSE increases); this is not an unexpected result. However, as the ratio of missing to complete data increases, the differences in performance between imputation methods become significant. These results imply that the characteristics of the dataset and missing values affect the performance of the classifier algorithms. Furthermore, the patterns of these effects differ depending on the imputation methods and classifiers used.

Lastly, we estimate the accuracy (RMSE) of each method by conducting a multiple regression analysis. As shown in Table 12, the results confirmed a significant association between the characteristics of the missing data and the method of imputation with the performance of each classification in terms of RMSE. In total, 226,800 datasets (3

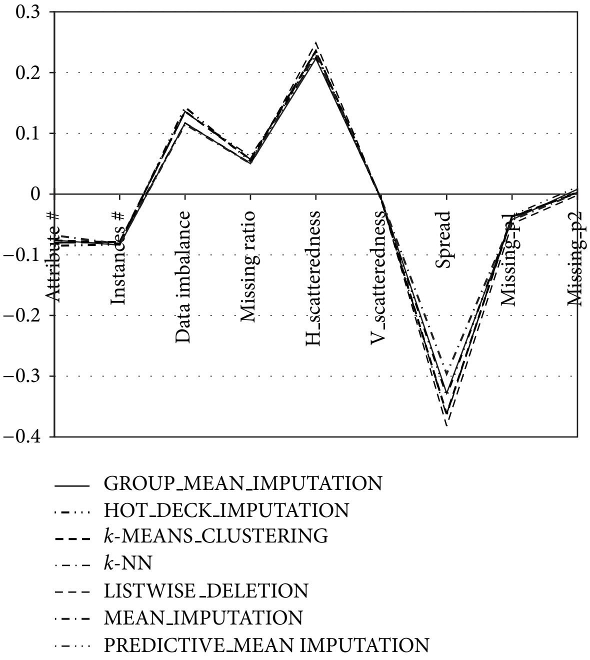


Figure 5: Coefficient pattern of the decision tree algorithm (RMSE).

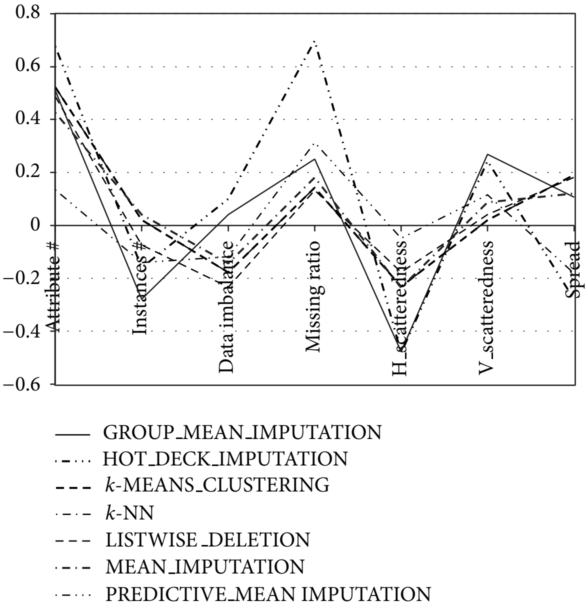


Figure 6: Coefficient pattern of the decision tree algorithm based on a 90% missing ratio (RMSE).

missing ratios × 3 missing patterns × 100 trials × 6 imputation methods × 7 classification methods) were analyzed. The results have at least two implications. First, we can predict the classification accuracy for an unknown dataset with missing data only if the data characteristics can be obtained. Second, we can establish general rules for selection of the optimal combination of a classification algorithm and imputation algorithm.

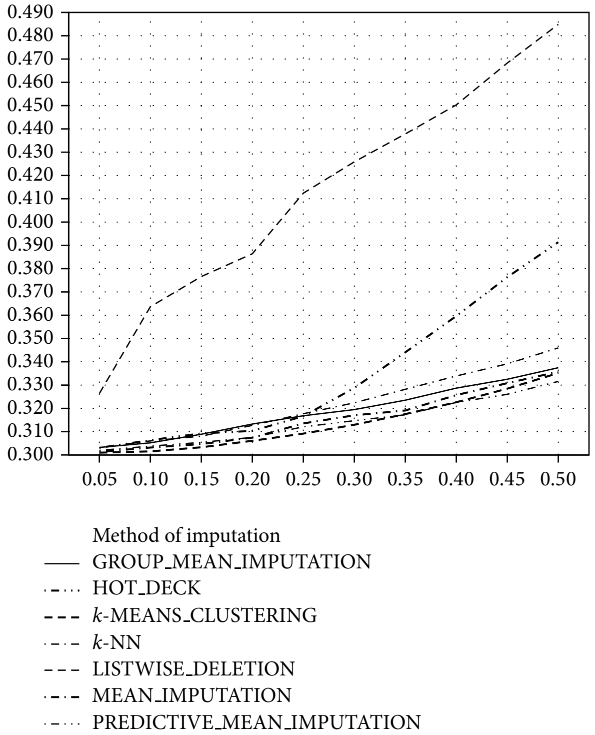


Figure 7: RMSE by ratio of missing data.

Table 12: Factors influencing accuracy (RMSE) of classifier algorithms.

Data characteristic	B	Data characteristic	B
(constant)	.060**	M_imputation_dum1	.012**
R_missing	.083**	M_imputation_dum2	−.001*
SE_HS	−.005**	M_imputation_dum4	000
SE_VS	.000**	M_imputation_dum5	000
Spread	.017**	M_imputation_dum6	.001**
N_attributes	−.008**	M_imputation_dum7	−.001*
C_imbalance	−.003**	P_missing_dum1	−.006**
N_cases	.002**	P_missing_dum3	.000

Note 1: Dummy variables related to imputation methods: LISTWISE DELETION (M_imputation_dum1 = 1, others = 0), MEAN_IMPUTATION (M_imputation_dum2 = 1, others = 0), GROUP_MEAN_IMPUTATION (M_imputation_dum3 = 1, others = 0), PREDICTIVE_MEAN_IMPUTATION (M_imputation_dum4 = 1, others = 0), HOT_DECK (M_imputation_dum5 = 1, others = 0), k-NN (M_imputation_dum6 = 1, others = 0), and k-MEANS_CLUSTERING (M_imputation_dum7 = 1, others = 0). Missing patterns: univariate (P_missing_dum1 = 1, P_missing_dum2 = 0, P_missing_dum3 = 0), monotone (P_missing_dum1 = 0, P_missing_dum2 = 1, P_missing_dum3 = 0), and arbitrary (P_missing_dum1 = 1, P_missing_dum2 = 1, P_missing_dum3 = 1). B: standard beta coefficient.
Note 2: $^{*}P < 0.1$, $^{**}P < 0.05$.

6. Conclusion

So far, the prior research does not fully inform us of the fitness among datasets, imputation methods, and classification algorithms. Therefore, this study ultimately aims to establish a rule set which guides the classification/recommender system developers to select the best classification algorithm based on the datasets and imputation method. To the best of our knowledge, ours is the first study in which the performance of classification algorithms with multiple dimensions (datasets, imputation data, and imputation methods) is discussed. Prior research examines only one dimension [15]. In addition, as shown in Figure 3, since the performance of each method differs according to the dataset, the results of prior studies on imputation methods or classification algorithms depend on the datasets on which they are based.

In this paper, factors affecting the performance of classification algorithms were identified as follows: characteristics of missing values, dataset features, and imputation methods. Using benchmark data and thousands of variations, we found that several factors were significantly associated with the performance of classification algorithms. First, as expected, the results show that the missing data ratio and spread are negatively associated with the performance of the classification algorithms. Second and as a new finding to our best knowledge, we observed that the number of missing cells in each record (SE_HS) was more sensitive in affecting the classification performance than the number of missing cells in each feature (SE_VS). Further, we found it interesting that the number of features negatively affects the performance of the logistic algorithm, while other factors do not.

A disadvantage of logistic regression is its lack of flexibility. The assumption of a linear dependency between predictor variables and the log-odds ratio results in a linear decision boundary in the instance space, which is not valid in many applications. Hence, in the case of data imputation, the logistic algorithm must be avoided. Next, in response to concerns about class imbalance, which has been discussed in data mining research [18, 19], we found that the degree of class imbalance was the most significant data feature to decrease the predicted performance of classification algorithms. In particular, SMO was second to none in predicting SE_HS in any imputation situation; that is, if a dataset has a high number of records in which the number of missing cells is large, then SMO is the best classification algorithm to apply.

The results of this study suggest that optimal selection of the imputation method according to the characteristics of the dataset (especially the patterns of missing values and choice of classification algorithm) improves the accuracy of ubiquitous computing applications. Also, a set of optimal combinations may be derived using the estimated results. Moreover, we established a set of general rules based on the results of this study. These rules allow us to choose a temporally optimal combination of classification algorithm and imputation method, thus increasing the agility of ubiquitous computing applications.

Ubiquitous environments include a variety of forms of sensor data from limited service conditions such as location, time, and status, combining various different kinds of sensors. Using the rules deducted in this study, it is possible to select the optimal combination of imputation method and classification algorithm for environments in which data changes dynamically. For practitioners, these rules for selection of the optimal pair of imputation method and classification algorithm may be developed for each situation depending on the characteristics of datasets and their missing values. This set of rules will be useful for users and developers of intelligent systems (recommenders, mobile applications, agent systems, etc.) to choose the imputation method and classification algorithm according to context while maintaining high prediction performance.

In future studies, the predicted performance of various methods can be tested with actual datasets. Although, in prior research on classification algorithms, multiple benchmark datasets from the UCI laboratory have been used to demonstrate the generality of the proposed method, performance evaluations in real settings would strengthen the significance of the results. Further, for brevity, we used a single performance metric, RMSE, in this study. For example, FP rate, as well as TP rate, is very crucial when it comes to investigating the effect of class imbalance, which is considered in this paper as an independent variable. Although the performance results would be very similar when using other metrics such as misclassification cost and total number of errors [20], more valuable findings may be generated from a study including these other metrics.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by the National Strategic R&D Program for Industrial Technology (10041659) and funded by the Ministry of Trade, Industry, and Energy (MOTIE).

References

[1] J. Augusto, V. Callaghan, D. Cook, A. Kameas, and I. Satoh, "Intelligent environments: a manifesto," *Human-Centric Computing and Information Sciences*, vol. 3, no. 12, pp. 1–18, 2013.

[2] R. Y. Toledo, Y. C. Mota, and M. G. Borroto, "A regularity-based preprocessing method for collaborative recommender systems," *Journal of Information Processing Systems*, vol. 9, no. 3, pp. 435–460, 2013.

[3] G. Batista and M. Monard, "An analysis of four missing data treatment methods for supervised learning," *Applied Artificial Intelligence*, vol. 17, no. 5-6, pp. 519–533, 2003.

[4] R. Shtykh and Q. Jin, "A human-centric integrated approach to web information search and sharing," *Human-Centric Computing and Information Sciences*, vol. 1, no. 1, pp. 1–37, 2011.

[5] H. Ihm, "Mining consumer attitude and behavior," *Journal of Convergence*, vol. 4, no. 2, pp. 29–35, 2013.

[6] Y. Cho and S. Moon, "Weighted mining frequent pattern based customers RFM score for personalized u-commerce recommendation system," *Journal of Convergence*, vol. 4, no. 4, pp. 36–40, 2013.

[7] N. Howard and E. Cambria, "Intention awareness: improving upon situation awareness in human-centric environments," *Human-Centric Computing and Information Sciences*, vol. 3, no. 9, pp. 1–17, 2013.

[8] L. Liew, B. Lee, Y. Wang, and W. Cheah, "Aerial images rectification using non-parametric approach," *Journal of Convergence*, vol. 4, no. 2, pp. 15–21, 2013.

[9] K. J. Nishanth and V. Ravi, "A computational intelligence based online data imputation method: an application for banking," *Journal of Information Processing Systems*, vol. 9, no. 4, pp. 633–650, 2013.

[10] P. Kang, "Locally linear reconstruction based missing value imputation for supervised learning," *Neurocomputing*, vol. 118, pp. 65–78, 2013.

[11] J. L. Schafer and J. W. Graham, "Missing data: our view of the state of the art," *Psychological Methods*, vol. 7, no. 2, pp. 147–177, 2002.

[12] H. Finch, "Estimation of item response theory parameters in the presence of missing data," *Journal of Educational Measurement*, vol. 45, no. 3, pp. 225–245, 2008.

[13] S. J. Press and S. Wilson, "Choosing between logistic regression and discriminant analysis," *Journal of the American Statistical Association*, vol. 73, no. 364, pp. 699–705, 1978.

[14] E. Frank, Y. Wang, S. Inglis, G. Holmes, and I. H. Witten, "Using model trees for classification," *Machine Learning*, vol. 32, no. 1, pp. 63–76, 1998.

[15] O. Kwon and J. M. Sim, "Effects of data set features on the performances of classification algorithms," *Expert Systems with Applications*, vol. 40, no. 5, pp. 1847–1857, 2013.

[16] E. Namsrai, T. Munkhdalai, M. Li, J.-H. Shin, O.-E. Namsrai, and K. H. Ryu, "A feature selection-based ensemble method for arrhythmia classification," *Journal of Information Processing Systems*, vol. 9, no. 1, pp. 31–40, 2013.

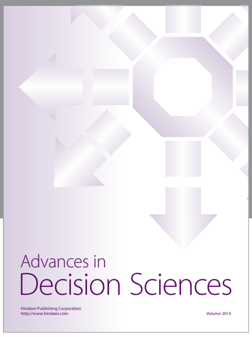
[17] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, Morgan Kaufmann, San Francisco, Calif, USA, 2nd edition, 2005.

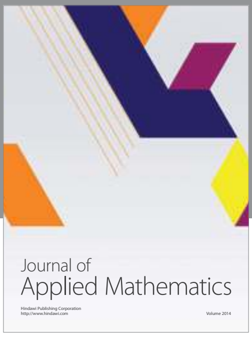
[18] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches," *IEEE Transactions on Systems, Man and Cybernetics C: Applications and Reviews*, vol. 42, no. 4, pp. 463–484, 2012.

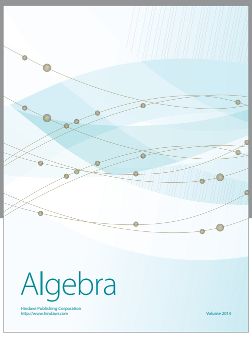
[19] Q. Yang and X. Wu, "10 challenging problems in data mining research," *International Journal of Information Technology & Decision Making*, vol. 5, no. 4, pp. 597–604, 2006.

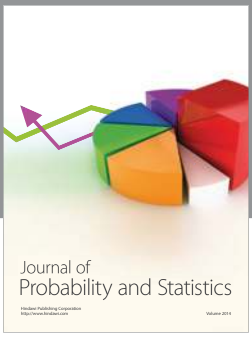
[20] Z.-H. Zhou and X.-Y. Liu, "Training cost-sensitive neural networks with methods addressing the class imbalance problem," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 1, pp. 63–77, 2006.

Advances in
Operations Research

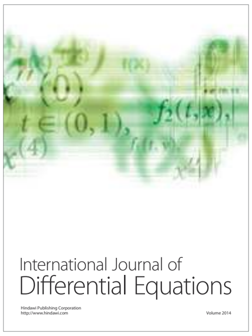

Advances in
Decision Sciences


Journal of
Applied Mathematics

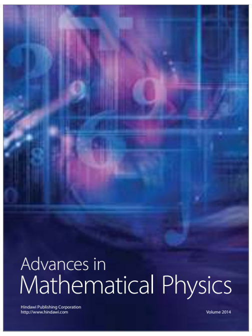

Algebra

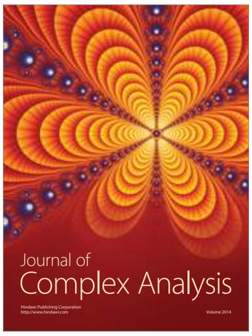

Journal of
Probability and Statistics

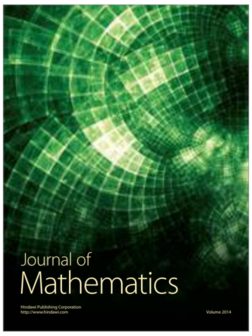
The Scientific
World Journal

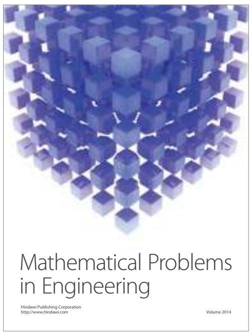

International Journal of
Differential Equations

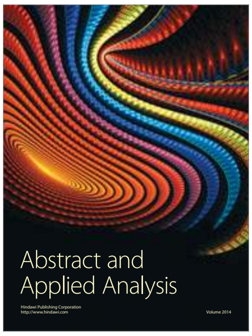
International Journal of
Combinatorics

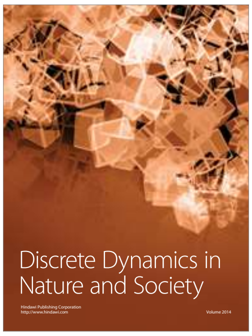

Advances in
Mathematical Physics

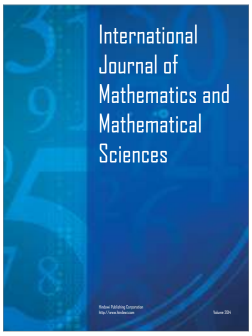

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

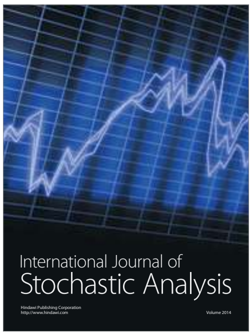

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization