Expert Systems With Applications 46 (2016) 485–493

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Adaptive pairing of classifier and imputation methods based on the characteristics of missing values in data sets

Jaemun Sim [a], Ohbyung Kwon [b,*], Kun Chang Lee [a]

[a] SKKU Business School, Sungkyunkwan University, Seoul 110-745, Republic of Korea
[b] School of Management, Kyung Hee University, Seoul 130-701, Republic of Korea

ARTICLE INFO

Keywords:
Classification algorithms
Imputation methods
Case-based reasoning
Experiments

ABSTRACT

Classifiers and imputation methods have played crucial parts in the field of big data analytics. Especially, when using data sets characterized by horizontal scattering, vertical scattering, level of spread, compound metric, imbalance ratio and missing ratio, how to combine those classifiers and imputation methods will lead to significantly different performance. Therefore, it is essential that the characteristics of data sets must be identified in advance to facilitate selection of the optimal combination of imputation methods and classifiers. However, this is a very costly process. The purpose of this paper is to propose a novel method of automatic, adaptive selection of the optimal combination of classifier and imputation method on the basis of features of a given data set. The proposed method turned out to successfully demonstrate the superiority in performance evaluations with multiple data sets. The decision makers in big data analytics could greatly benefit from the proposed method when it comes to dealing with data set in which the distribution of missing data varies in real time.

1. Introduction

Emerging infrastructures like cloud systems, smart grids, pervasive computing systems and network-related processing are providing managers and practitioners with more flexible utilities for the sake of adopting user-intended applications (Nia, Atani, & Haghi, 2014). Such infrastructures have greatly contributed to producing big data set in a format of massive amounts of streamed information from a wide variety of the network-connected objects (Sowe, Kimata, Dong, & Zettsu, 2014). Correspondingly, data sets in marketing, scheduling, and manufacturing businesses become very large (volume), get rapidly updated by streaming (velocity) (Bifet, 2013), and/or inadvertently tend to be incomplete due to the nature of their sources like IoT (Internet of Things) and social networking services (SNSs) (variety) as well (Chen, Mao, Zhang, & Leung, 2014). It is no surprise that the significant challenges in this type of dataset encompass the unstable data structure and/or characteristics with null value problems caused by either rapidly changing user locations, fault sensors or user's non-responses. The problems like this become more serious when the big data application systems implemented on powerful classifiers tend to repeatedly show poor performances because of the constantly changing patterns of missing data, data volume and data structure embedded in the big data sets.

To cope with these challenges, intelligent applications must be improved in the following ways. First, due to the volume and velocity of data in these data sets, scalable classification is required (Jang, 2014; Liu, Blasch, Chen, Shen, & Chen, 2013). Second, with respect to variety, many null values must be included in order to maintain a satisfactory level of reasoning accuracy (Kim, 2012; Wu, Zhu, Wu, & Ding, 2014). To alleviate these problems, it is necessary to develop a sophisticated method of finding optimal pairs from every possible classifier/imputation method pair in real time.

According to the literature in this area, characteristics of missing data, data sets, and imputation methods may influence the performance of classification algorithms (Sim, Lee, & Kwon, 2015). Research in various data domains has been conducted related to selecting an imputation method that improves the performance of a classifier, and several new imputation methods have been proposed (Farhangfar, Kurgan, & Dy, 2008; Hengpraphrom, Wichian, & Meesad, 2010; Kang, 2013; Liu & Brown, 2013; Luengo, García, & Herrera, 2010; Silva & Hruschka, 2013). Although most imputation methods improve overall classification performance, the magnitude of improvement differs according to the problem domain (Farhangfar et al., 2008; Hengpraphrom et al., 2010; Su, Khoshgoftaar, & Greiner, 2008). The differences in magnitude become clearer as the ratio of missing data increases (Hengpraphrom et al., 2010; Su et al., 2008). To the best of our knowledge, when experimenting with various data sets, no imputation method has always proven superior to other methods in

* Corresponding author. Tel.: +8229612148; fax: +8229610515.
E-mail addresses: deskmoon@gmail.com (J. Sim), obkwon@khu.ac.kr, obyung@gmail.com (O. Kwon), kunchanglee@gmail.com (K.C. Lee).

http://dx.doi.org/10.1016/j.eswa.2015.11.004
0957-4174/

combination with any specific classifiers, because the effect of an imputation method on a classifier differs according to the data set (Farhangfar et al., 2008; Kang, 2013).

If the characteristics of the data set are invariant and fully known beforehand, as prior studies have assumed, identification of an optimal combination of a classifier and imputation method would be possible. However, if the data is collected in real time, the characteristics of the data set will differ depending on the timeline. In this case, the performance of all possible pairs of classifiers and imputation methods for all types of data characteristics must be evaluated in order to select the optimal combination. Moreover, if real-time analysis is needed for an application, an autonomous method of selecting this optimal combination is necessary. However, very few studies have addressed this need for autonomous selection of classifiers and imputation methods based on the characteristics of a data set, especially as regards the structure of null values.

The purpose of this paper is to propose an adaptive method of selecting the optimal classification algorithm/imputation method pair. An autonomous, adaptive selection method should be able to recognize the features of a data set and, if necessary, make changes automatically. To develop this method, we amended case-based reasoning as follows: the original case base is preprocessed to derive a compound metric of a null data structure. Then a candidate set is formed by identifying multiple pairs, and a pair is selected from among the candidate pairs. To demonstrate the feasibility and superiority of the proposed method, we conducted experiments with multiple benchmark data sets and several classifiers and imputation methods that have been deemed suitable in previous studies for reasoning with incomplete data sets.

The paper is organized as follows: Section 2 describes the related works on imputation methods and classifiers. The proposed method and corresponding experiment, which shows the performance of the method, are delineated in Sections 3 and 4, respectively. Finally, in Section 5, we conclude with the implications of the study results to researchers and practitioners.

2. Related works

2.1. Selection of imputation methods

Researchers using supervised learning algorithms, such as those used for classification, have generally assumed that training data sets are complete and that all occurrences contain a value. Missing values are filled in using many imputation methods. Imputation techniques are based on the idea that missing data for a variable are replaced by an estimated value that is drawn from the distribution of existing values. In most cases, attributes of data sets are interdependent; thus, through identification of relationships among attributes, missing values can be determined (Batista & Monard, 2003; Kang, 2013; Li, Li, & Li, 2014).

There is no single superior imputation algorithm for replacing all missing data in a set, because all imputation methods are affected by the characteristics of the data set and the missing values (Kwon & Sim, 2013; Loh & H'ng, 2014). Thus, if the characteristics of a data set and its missing values are changed by some event, then the performance of the selected imputation method may be altered. For example, for sensor-based traffic data, which vary periodically under certain expected conditions such as changed load capacity or altered timeline, robust imputation algorithms using historical information may be prepared (Tan, Wu, Cheng, Wang, & Ran, 2014). However, various data sets, such as those from SNSs, may be changed by uncertain and complex events (Wrzus, 2013); therefore, the characteristics of missing values may also change. Due to this uncertainty, it is impossible to prepare a robust imputation method using data from prior experiments. Moreover, most sensor-based data require real-time decisions. The need for swift execution makes it difficult to select a suitable imputation method fast enough using the techniques outlined in existing studies in which comparative experiments among candidate imputation methods were performed. Considering the two factors of missing data variability and execution time, we assert that only metadata that influence the performance imputation method should be used to select a suitable imputation method. In addition, the following factors with respect to meta-data must be considered.

Missing ratios: When the ratio of missing to present data increases, the error of the imputation also increases and the difference in performance of the imputation method compared to other methods becomes larger. Each imputation method has a different pattern of performance for a given missing ratio (Hengpraphrom et al., 2010; Su et al., 2008).

Missing value distribution: For any given missing ratio, each imputation method has a different performance pattern according to the distribution of missing cells. For example, even if the same imputation method is used repeatedly, its performance may change according to the probability of missing cells in each feature (Wasito & Mirkin, 2006). Various patterns of missing data, such as missing completely at random (MCAR) and missing at random (MAR), can cause differences in the performance of the imputation method (Ghannad-Rezaie, Soltanian-Zadeh, Ying, & Dong, 2010). Here, MCAR refers to a missing data process that does not depend on either observed or missing values, whilst MAR is defined as a situation in which missingness depends on observed values, not on unobserved values (Wang, Xie, & Fisher, 2011).

Data set characteristics: The characteristics of a data set, such as the degree of imbalance, the size of the sample, and the number of features, influence imputation performance (Sim et al., 2015) because an imputation method is a form of machine learning. The performance of a machine learning algorithm depends on the characteristics of the data set (Kwon & Sim, 2013).

2.2. Selection of classifiers

The classification algorithm is one of the most important functions in the analysis of large data sets. Classification algorithms are the most widely used data mining models to extract valuable knowledge from huge amounts of data (Dogan & Zuhal, 2013). Classification is a data mining process that assigns items in a collection to target categories or classes. The goal of classification is to predict a target class for each case in the data set accurately (Akhila, Madhu, Madhu, & Pooja, 2014). Many comparative analyses are used to determine which algorithm is best suited for a particular data set. Classification capability depends on the types of algorithms and the characteristics of the data, such as the degree of imbalance, number of features, number of instances, and number of class types (Kwon & Sim, 2013; Liu & Zhou, 2006; Okamoto, 1963; Raudys & Pikelis, 1980). There is no superior classification algorithm for all types of data sets, because each classification algorithm is affected by the characteristics of the data set (Kwon & Sim, 2013). Moreover, when missing values are treated by a certain imputation method, the classification algorithm is also affected by the imputation method. Thus, each different imputation method/classifier pair results in a different performance, even if they treat the same data with the same missing values (Bastista & Monard, 2003; Farhangfar et al., 2008; Silva and Hruschka, 2013).

The descriptions of this causal relationship in the literature are insufficient. Intuitively, it seems that the cause may be related to the choice of the method of estimation and model of the machine learning algorithm, as both imputation methods and classifiers are forms of machine learning. The machine learning algorithm builds a model via its own method. For example, J48 divides classes with a split point (Safavian, 1991), whereas SVM divides classes with outer boundary points, such as marginal vectors (Suykens, 1999), and k-NN divides classes with similar instances (Zhang, 2007). This means that the chosen imputation method must estimate the determinant point

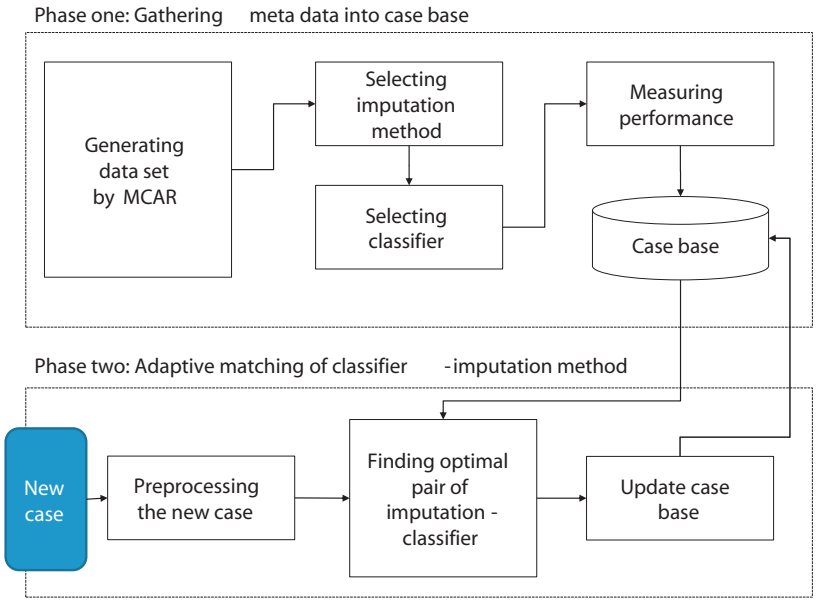


Fig. 1. Overall framework for adaptive matching method.

correctly according to the kind of classifier. Therefore, the imputation method may distort the determinant point depending on the characteristics of the machine-based learning algorithm.

As a result, to select an optimal pair including a classifier and an imputation method, the characteristics of the missing values and those of the data set must be considered. However, previously published methods must consider all possible pairs of classifiers and imputations. This process requires many resources and considerable execution time, and is very costly. To avoid excess consumption of time and resources while still selecting an agile classifier-imputation method combination, we use meta-data to determine the characteristics of the missing data in a given data set. Selection of pairs of algorithms must be able to change rapidly according to the characteristics of the data set, particularly the characteristics of null data. In this study, we refer to this as an adaptive matching of classifier and imputation (AMCI) method.

3. Methods

3.1. Overall architecture

Fig. 1 shows the overall architecture of the proposed method. In phase one, missing data are generated arbitrarily by the MCAR method (Wang et al., 2011) according to each missing ratio. All pairs of algorithms perform the imputation and classification using the generated missing data. Then the meta-data, which includes characteristics of the data set and the selected imputation-classification method combination and is sensitive to the relative accuracy of the classifier, are accumulated in the case base as a case. In phase two, the proposed algorithm finds the situated optimal classifier/imputation method pair from a new set of data. By preprocessing the data set, the algorithm identifies its characteristics and finds the best classifier/imputation method pair. If necessary, the results can be stored in the case base as a new case.

3.2. Data sets, classifiers, and imputation methods

To develop the adaptive pairing method and the relationships among data set features, classifiers, and imputation methods, we performed a pilot test with six data sets gathered from the UCI AI laboratory's repository of benchmarked data sets (see Table 1). Because the six data sets have no missing values, we created some null cells according to a given missing ratio and including horizontal scattering, vertical scattering, and a certain level of spread to test the effects of the imputation methods. In total, 500,000 varied data sets were generated.

Table 1
Benchmarked data set.

Dataset	# of cases	Features	Decision attributes
Iris	150	Numeric (4)	Categorical (3)
Wine	178	Numeric (13)	Categorical (3)
Glass identification	214	Numeric (9)	Categorical (7)
Liver disorder	345	Numeric (6)	Categorical (2)
Ionosphere	351	Numeric (34)	Categorical (2)
Statlog Shuttle	57999	Numeric (7)	Categorical (7)

Table 2
Input features.

Group	Features	
Input feature	Data characteristics	D_Imbalance
	Missing data characteristics	R_missing
		SE_HS
		SE_VS
		Spread
Class	Performance	Relative Accuracy

As the first step in the pilot test, we observed the relationships between the input features and relative accuracy when applying classifiers and imputation methods. The input features considered in this paper are shown in Table 2. Note that we disregarded the number of instances and number of attributes as input features because they had no relationship with the performance of classification algorithms in an earlier study (Kwon, 2013).

The next step was to determine the relative accuracy of the process, after which we attempted to detect any causality within the data set. As shown in Fig. 2(a), the degree of imbalance (D_imbalance) seemed to be independent of the relative accuracy. The missing ratio and level of spread were associated negatively with relative accuracy; the greater the amount of null data or the level of spread, the more difficult it is to achieve relatively accurate estimations (Fig. 2(b)

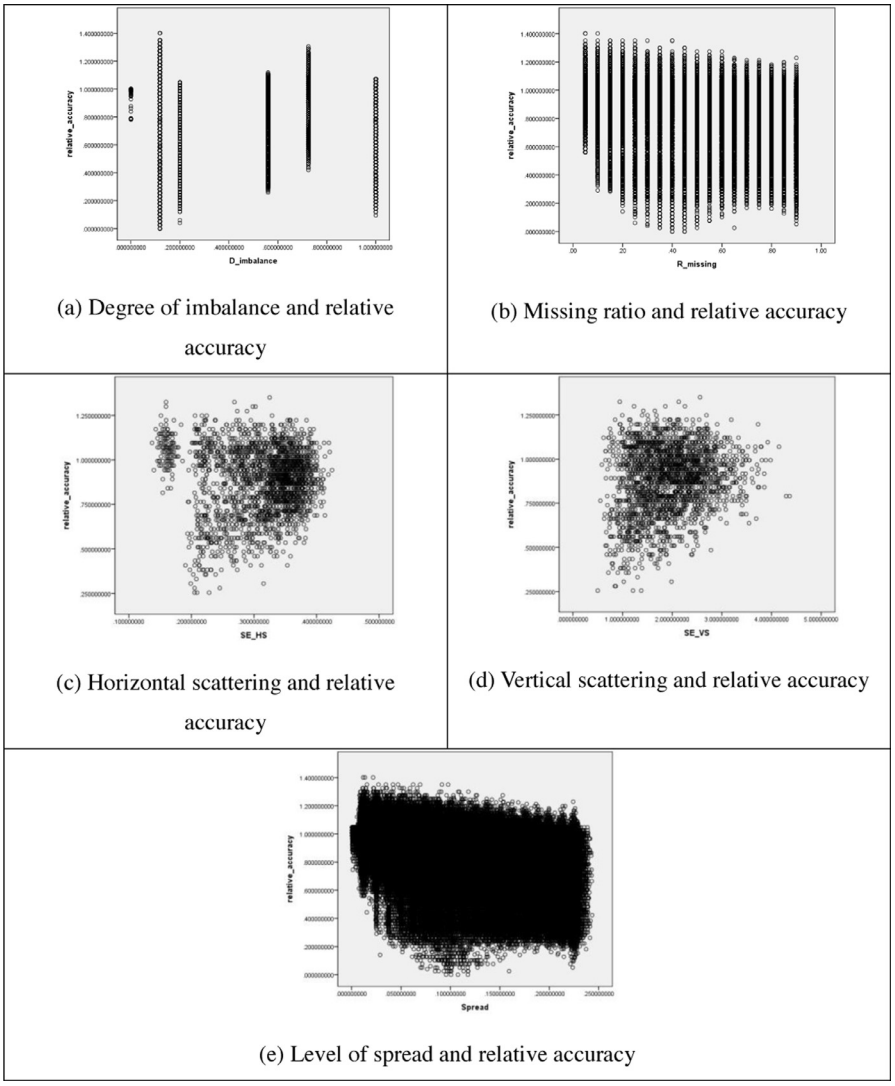


Fig. 2. Data set features and relative accuracy.

and (e)). As shown in Fig. 2(c) and (d), the scattering seemed to be unrelated to the relative accuracy. However, as shown in Fig. 3, we also found that if we bipartitioned the results of horizontal and vertical scattering based on the missing rate, the missing rate of both subgroups was greater than 0.5. In both cases, horizontal and vertical scattering were associated positively with relative accuracy only when the missing ratio was greater than 0.5. These findings imply that we need another input feature that combines horizontal and vertical scattering. This will be addressed in the next section.

3.3. Adaptive matching of classifiers and imputation methods

The algorithm of matching in the AMCI consists of three phases: [1] preprocessing the case base (Fig. 4); [2] reasoning the neighborhoods (Fig. 5), and [3] identifying the optimal of classifier/imputation method pair using Eq. (3). The process for each phase is outlined below. Fig. 4 states the first phase, in which cases are preprocessed. Our method uses primarily case-based reasoning to explore candidate pairs of imputation methods and classification algorithms and to compute the similarity, imbalance rate, missing rate, and compound metric of the degree of HS(v1), VS(v2), and spread(v3). To derive the compound metric, we normalized the values of v1, v2, and v3 as:

$$x_j = \frac{v_j - m(v_j)}{M(v_j) - m(v_j)} \tag{1}$$

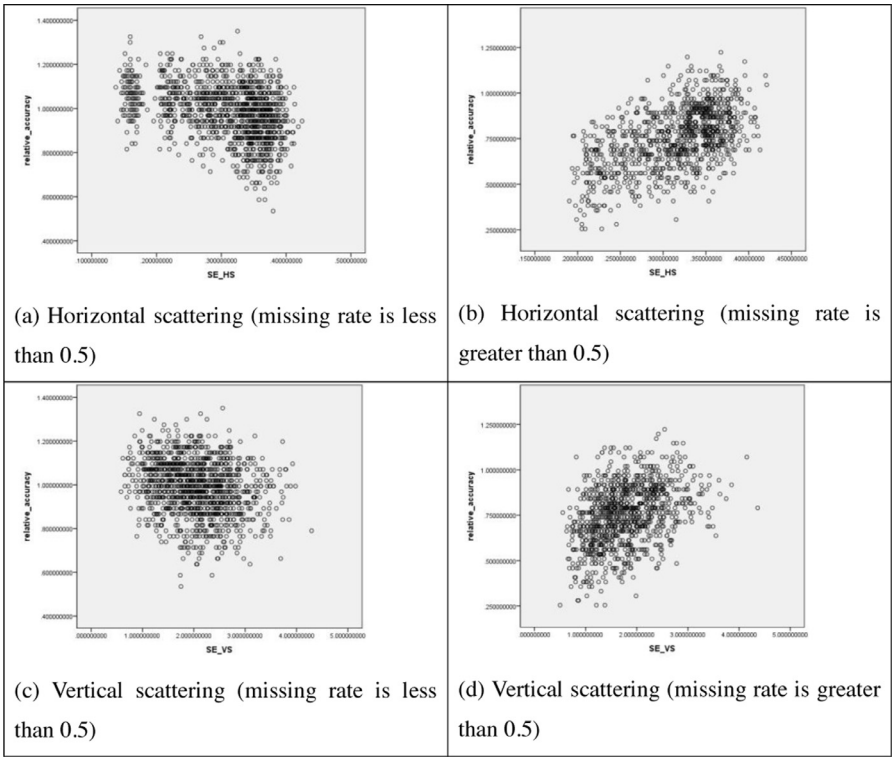
where $M(v_j)$ and $m(v_j)$ indicate the maximum and minimum values of all elements in v_j, respectively. Then, the compound metric (cm) is calculated as follows:

$$cm = \sqrt{x_1^2 + x_2^2 + x_3^2} \tag{2}$$

Next, using the imbalance rate, missing rate, and *cm* as input features, and the pairs of classification algorithms and imputation methods (CLIM) as classes, the proposed method identifies a similar set from the case base (i.e., the number of neighbors); the number of neighbors may vary.

As shown in Fig. 5, reasoning the neighborhoods is a form of case-based reasoning. The primary difference is that it selects M neighbors among the amended training cases (AC), in which the conceptual distance from the test case is shorter than in any other N-M cases. The greater the value of *M*, the more likely it is that the performance measures (RMSE, AC) will improve. However, increasing the number of *M* would require more elapsed time, which then has a negative effect on the scalability of the algorithm. The issue of scalability is of great importance, as the amount of training data is very large. Therefore, the best *M* with respect to performance measures and elapsed time must be computed.

Identifying an optimal classifier/imputation pair from a set of neighbors (NE) consists simply of selecting a pair in which the combined value of RMSE and relative accuracy (RA) is higher than the



(a) Horizontal scattering (missing rate is less than 0.5)

(b) Horizontal scattering (missing rate is greater than 0.5)

(c) Vertical scattering (missing rate is less than 0.5)

(d) Vertical scattering (missing rate is greater than 0.5)

Fig. 3. Scatterings and relative accuracy.

Input: training cases (TR) **Output**: amended cases (AC)
Process: Get meta data of case base; Compute compound metric (cm) from horizontal scattering (v1), vertical scattering (v2), and level of spread (v3); Newly develop amended cases (AC) using cm, imbalance ratio, and missing ratio as input features and CLIM, RMSE, and relative accuracy as classes; Return AC ;

Fig. 4. Phase 1: Preprocessing Cases.

values of any other neighbors. The combined value of RMSE and RA is computed as follows:

$$CV = (1 - RMSE) + RA \tag{3}$$

4. Experiments

4.1. Implementation

All experiments were performed in an implementation environment (see Table 3). The algorithms were developed as a Java application using the Weka library.

4.2. Data sets

Six data sets were used for classification. A description of each data set is provided in Table 1. The classification data sets were selected from the UCI AI laboratory's repository, which is accessible at http://archive.ics.uci.edu/ml/. UCI benchmark data sets have been widely used to test the new data mining algorithm or methods (Gao, Liu, Peng, & Jian, 2015; Xiang, Yu, & Kang, 2015; Zhang, Fang, Ma, & Zhao, 2016). The number of instances varied from 150 (Iris) to 57,999 (Statlog), while the number of attributes varied from 4 (Iris) to 60 (Sonar). Two datasets (Liber disorder, Ionosphere) were binary classification problems, while others were multiclass classification problems.

To quantify how missing data affected performances, we generated the following synthetic missing cells. First, we selected 18 missing instance ratios: 5%, 10%, 15%, 20%, 25%, 30%, ~80%, 85%, and 90%. For each missing data set, the cell that was located at a randomly selected instance and its attribute were marked as missing. For each missing ratio, we generated different missing data to ensure that the experimental results were statistically acceptable.

4.3. Design

We tested the proposed method, adaptive matching of classifier and imputation (AMCI), against seven imputation methods and eight classification algorithms. To compare performance among the competing methods, the same test data set was utilized in all methods for

Input: amended cases (AC), level of optimal number (M)

Output: neighbors (NE)

Process:

While (!end of amended cases ($ac \in AC$)) {

Compute conceptual (Euclidean) distance (d) of ac and test case;

If d is less than any of the currently selected M cases, then include the new case, ac, into the M neighbors; Otherwise, set 0;

} // End while

Return IR;

Fig. 5. Phase 2: Reasoning the neighborhoods.

Table 3
Implementation environment.

Category		Description
Java SDK environment	Version: 1.7	
Classifier	Using weka 3.7.2 release	
Imputation algorithm implementation	GROUP_MEAN_IMPUTATION	Developed
	MEAN_IMPUTATION	
	HOT_DECK	
	PREDICTIVE_MEAN_IMPUTATION	Developed using regression library in Weka 3.7.2
	kMEANS_CLUSTERING	Developed using k-Means library in Weka 3.7.2
	KNN	Developed using kNN library in Weka 3.7.2
Proposed algorithm	Developed	

each experiment. To generate data for performance of classification algorithms, we adopted the Weka software tool release 3.7.2.

The experiment was iterated 8400 times with different random sampling. For each experiment, out of 102,906 samples, 95% were used randomly for training, and the remaining 5% were used to test the methods. For experimental simulation, a Java application program was developed.

We used two performance measures and the traditional accuracy measurements. In the literature, overall performance was computed using two global metrics: root mean squared error (RMSE) and relative accuracy (RA). RMSE is one of the most widely accepted metrics for testing a classifier's reasoning accuracy regardless of the type of class (numeric or nominal) (Fire & Elovici, 2015). In addition, RA refers to the ratio of the quotient of the observed result to the true value. RA shows the accuracy and stability of the difference between the true value and the estimated value provided by a classifier; it is also related to consistency (van Leeuwen & Cardinaels, 2015). RMSE and RA can be computed as (4) and (5), respectively:

$$RMSE = \sqrt{\sum_{\forall i} (y_i - \hat{y}_i)^2 / N} \tag{4}$$

and

$$RA = \xi(j|S(i)) / \xi(j|P) \tag{5}$$

where $\xi(j|S(i))$indicates the overall accuracy of the classification algorithm j when using the data set imputed by the imputation method, i, and $\xi(j|P)$indicates the overall accuracy of the classification algorithm j when using a perfect data set (P).

The proposed method (AMCI) was compared with two other methods: best case and average case. Best case (BEST) indicates the CLIM that shows the best performance in terms of RMSE and RA, while average case indicates the average (AVERAGE) value of all CLIMs. As we considered eight classification algorithms and seven imputation methods, 56 CLIMs in total were evaluated and compared to the proposed method.

4.4. Overall results: comparison of all CLIMs

The proposed method is compared with a variety of conventional data mining algorithms such as J48, Bayes Net, SMO (Sequential Minimal Optimization), Regression, Logistic, IBk (Instance-Based k neighbors), JRip (Repeated Incremental Pruning to Produce Error Reduction), and RBF (Radial Basis Function) Network. These should increase the validity of the result of the experiment. They are available in the open source data mining software WEKA (downloadable at http://www.cs.waikato.ac.nz/ml/weka/). We show the results of the evaluation in terms of RMSE and RA in Tables 4 and 5, respectively. Fifty-six baseline approaches were included to compare performance from J48, Group Mean Imputation to RBFNetwork, KNN. For RMSE, the hot deck imputation method is second to none among the imputation methods available. However, our proposed method (mean = 0.272207, standard error = 0.028857) outperformed the best pair (BayesNet and hot deck method; mean = 0.281878, standard error = 0.003480). Based on the value of the standard error, we concluded that further statistical comparison was not needed. As for RA, the group mean imputation method worked best when BayesNet, SMO, Regression, and RBFNetwork were selected as classifiers. When J48, Logistic, IBK, and JRip were used, the hot deck imputation method performed better than the group mean imputation method. However, the proposed method showed the best performance in terms of RA (mean = 0.879858, standard error = 0.083301). In summary, the proposed method, AMCI, outperformed all other pairs of classifiers and imputation methods when the pairs were applied constantly to all data sets regardless of characteristics of the data, such as rate of imbalance, ratio of null data, and class.

4.5. Results: RMSE

As the first efficiency test to compare the proposed method with AVERAGE and BEST—which consist of 56 pairs of classifiers and imputation methods—assessment by RMSE was performed. RMSE for the proposed method ranged from 0.2530 (number of neighbors = 100) to 0.3435 (number of neighbors = 1). The RMSE of BEST and AVERAGE were approximately 0.2800 and 0.3400, respectively. The comparison

Table 4
Performance comparison (RMSE).

	J48	Bayes net	SMO	Regression	Logistic	IBK	JRip	RBF network
Group_Mean_Imputation	0.356591	0.309359	0.343969	0.320050	0.364371	0.346020	0.358097	0.329209
	0.006165	0.006075	0.004079	0.003524	0.003773	0.003760	0.005787	0.006578
Likewise_Deletion	0.397702	0.413210	0.394057	0.385942	0.412258	0.416624	0.362461	0.382729
	0.010202	0.005129	0.011740	0.014517	0.007545	0.006525	0.010916	0.014858
Mean_Imputation	0.354487	0.310060	0.343285	0.314655	0.321278	0.363741	0.334816	0.356787
	0.006486	0.003408	0.004572	0.002889	0.002020	0.004096	0.005956	0.006404
Predictive_Mean_Imputation	0.353228	0.295936	0.361079	0.339308	0.349956	0.382560	0.327603	0.314058
	0.007453	0.003681	0.004388	0.004310	0.003869	0.005010	0.004386	0.004559
Hot_Deck	0.311324	0.281878	0.325576	0.297620	0.301338	0.332451	0.301968	0.290486
	0.004124	0.003480	0.002388	0.002703	0.003178	0.005507	0.004475	0.002827
k-Means_Clustering	0.361323	0.312874	0.340721	0.314721	0.324852	0.357812	0.334016	0.360273
	0.004760	0.004739	0.002702	0.002933	0.001851	0.003862	0.004712	0.007063
k-NN	0.367720	0.310929	0.358374	0.327542	0.326594	0.391663	0.332802	0.345255
	0.007731	0.005506	0.006125	0.002154	0.003829	0.005229	0.001632	0.005678
Proposed	0.272207							
	0.028857							

Note. In each cell, upper and lower values indicate average and standard deviation, respectively.

Table 5
Performance comparison (RA).

	J48	Bayes Net	SMO	Regression	Logistic	IBK	JRip	RBF network
Group_Mean_Imputation	0.797864	0.819528	0.771519	0.769424	0.768161	0.822462	0.775541	0.815556
	0.007561	0.009134	0.006228	0.005467	0.004642	0.007214	0.008278	0.007532
Likewise_Deletion	0.605180	0.437839	0.586239	0.593827	0.615006	0.460395	0.605915	0.660473
	0.038375	0.007948	0.021766	0.031800	0.029011	0.018158	0.032796	0.036940
Mean_Imputation	0.758118	0.722010	0.729093	0.749978	0.775583	0.773486	0.745613	0.717603
	0.010556	0.012384	0.008643	0.005932	0.008515	0.009407	0.010883	0.013526
Predictive_Mean_Imputation	0.761350	0.754095	0.703034	0.726566	0.726512	0.759400	0.760016	0.771316
	0.015391	0.017579	0.008888	0.008029	0.013440	0.009882	0.010179	0.011386
Hot_Deck	0.823016	0.770412	0.715117	0.740577	0.782080	0.829999	0.787352	0.807142
	0.010726	0.021393	0.011654	0.012927	0.015046	0.009522	0.015400	0.010059
k-Means_Clustering	0.752997	0.729183	0.726600	0.752337	0.770295	0.781152	0.752209	0.715990
	0.009641	0.012672	0.009051	0.009745	0.007808	0.009713	0.009246	0.013290
k-NN	0.737167	0.717503	0.710794	0.729881	0.747705	0.741664	0.741056	0.721692
	0.012029	0.011768	0.013983	0.009430	0.010821	0.014871	0.008295	0.010977
Proposed	0.879858							
	0.083301							

Note. In each cell, upper and lower values indicate average and standard deviation, respectively.

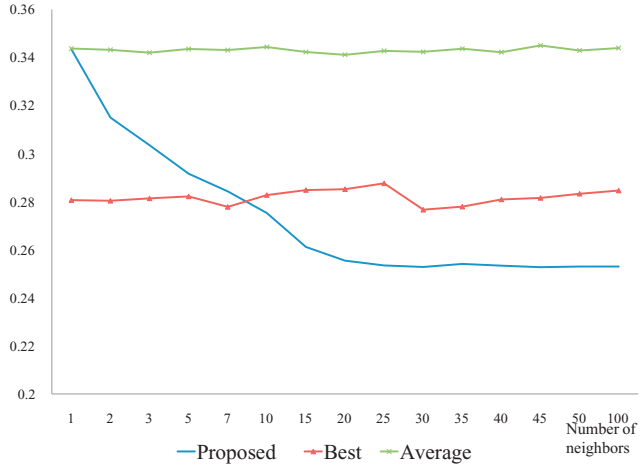


Fig. 6. RMSE.

with BEST and AVERAGE is shown in Fig. 6. Hence, we can conclude that the proposed method was superior to BEST when the number of neighbors was greater than a certain threshold (10 in this study) and that it always outperformed AVERAGE. Moreover, the RMSE of the proposed method stabilized as the number of neighbors exceeded 20. This implies that the number of neighbors need not increase to improve performance; this is a good feature when scalability issues are considered.

4.6. Results: RA

In terms of RA, which is the ratio of the accuracy of the imputed data set and that of a perfect data set when the same classification algorithm is applied, the accuracy of the proposed method ranged from 0.7339 to 0.9370, which was always superior to AVERAGE (approximately 0.7300; see Fig. 7). The proposed method outperformed BEST when we increased the number of neighbors to 10. The RA of the proposed method also improved as the number of neighbors increased. However, the performance stabilized when the number of neighbors exceeded 25, which also shows that the proposed method is scalable. Note that no more than 25 neighbors needed to be collected. Thus, the overconsumption of computation resources was reduced.

5. Discussion and future work

In this study, we have proposed an adaptive algorithm to find the best combination of classifier and imputation method on the basis of the missing value characteristic of the target data set. Such an effort of developing the proposed adaptive algorithm is significant in the big data application domains where target data sets are continuously suffering from fast changing data structures, random occurrence of missing values, and exponentially increasing data volumes. The typical trends of the big data analysis are easily observed in the fields of Internet of Things (IoT) and digital banking. Endlessly changing locations of users and things involved in those applications lead to the

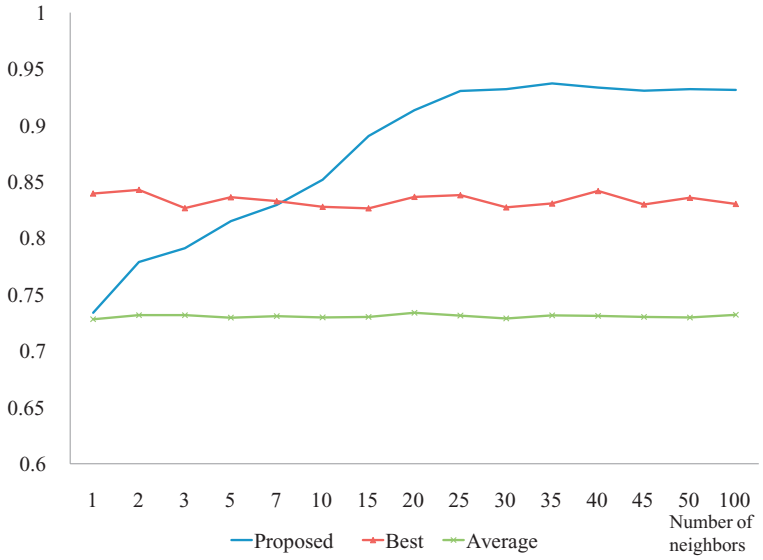


Fig. 7. Performance comparison of relative accuracy.

necessity with which we have successfully developed the proposed AMCI method.

The main advantages of the proposed method over traditional methods include the novelty in covering the patterns of imputation, and the superiority in terms of scalability and reasoning accuracy. First, our study, to the best of our knowledge, is the first to show how the relationship among the characteristics of a data set, the chosen imputation method, and the chosen classifier can affect classification performance in terms of overall accuracy. Many previous researchers were unable to find a combination of classifier and imputation method that is always superior to other methods in all cases and applications (Sim et al., 2015; Farhangfar, 2008; Kang, 2013). A lot of studies test classifiers with multiple data sets assuming no missing values; hence, the results of their works are not applicable in most situations (Jones, Johnstone, & Wilson, 2015). Recent literature discusses the impact of the choice of imputation method on classification performance. However, usually only one classifier is considered, such as the nearest neighbor rule (Sarez et al., 2015; Jiang & Yang, 2015; Orczyk & Porwik, 2015) or genetic algorithms (Tran, Andreae, & Zhang, 2015). Other studies explore a variety of classifiers to identify the most appropriate solution to a specific pattern of missing values (Xiao, Zhu, Teng, He, & Liu, 2014). Recently, Sim and Kwon (2015) find a relationship between the performance of classifiers and data characteristics. However, they did not mention how to cope with changes in the combination. Based on the results of our study, the proposed pairing of classifier–imputation method, AMCI, is very adaptive and successful. Once the characteristics of a data set are identified, AMCI adaptively selects the optimal combination of imputation and classification methods to ensure adequate performance. The results of our experiments clearly show the superiority of the proposed adaptive method compared to other methods of pairing classifiers and imputation methods.

Secondly, our study is also the first to consider full-fledged characteristics of the pattern of missing values. Basically, literature shows that missing values patterns include a wide variety of ones such as horizontal scattering, vertical scattering, level of spread, compound metric, imbalance ratio and missing ratio. To the best of our knowledge, there is no study proposing a combination of classifier and imputation method to tackle all the missing values patterns. The values of our proposed method are therefore clear when considering other works in which such an effort is not tried (Sarez et al., 2015; Jiang & Yang, 2015; Orczyk & Porwik, 2015, Tran, 2015).

Thirdly, the proposed AMCI method is also scalable. Performance in terms of RMSE and RA remained superior to other current methods regardless of the number of neighbors. The results indicate that no increase in the computational effort required to select the optimal pair is necessary to improve the performance of the proposed method. Hence, AMCI is very practical and usable in large data analytic settings. Large data sets have high volume, high updating velocity, and/or may be incomplete due to the nature of the sources. For example, the variety of social data from SNSs inevitably leads to missing values, and data from sensory networks required to implement the Internet of Things architecture are notoriously unwieldy. Due to the high volume and velocity of such data sets, scalable classification is definitely required. As for variety, large numbers of null values must be dealt with in order to maintain a satisfactory level of reasoning accuracy. Hence, very rapid methods of finding optimal pairs from among every possible pair of classifiers and imputation methods in real time must be made possible and available. Based on the results of our experiments, we posit that the proposed method can also be applied in large data analytics.

In conclusion, we believe that our proposed AMCI method can serve, in the field of real-time data anlaytics, as a valuable tool with appropriate accuracy, and reduced costs to work with. We know that there exist a number of issues that need to be overcome in the future studies. Firstly, use of a real data set may strengthen the implications of the results of the experiment. Secondly, though this study proposes a core principle how to combine the classifiers and imputation methods, we still need to strengthen the adaptive pairing of classifier and imputation method so that the proposed method can be applied more actively to the real-time online big data applications (Jiang & Yang, 2015). Such a further study effort may be based on vouching the valid significance of statistics for original population of data set (Carlin, 2002; Rubin, 1987). Finally, we need to consider the ensemble of the classifiers when selecting optimal combination with imputation methods. A state of the art research is beneficial to such efforts (Orczyk & Porwik, 2015; Twala & Cartwright, 2005; Xiao et al., 2014).

Acknowledgments

This work was supported by ICT R&D program of MSIP/IITP. [R0126-15-1007, Curation commerce based global open market system development for personal happiness enhancement].

References

Akhila, G. S., Madhu, G. D., Madhu, M. H., & Pooja, M. H. (2014). Comparative study of classification algorithms using data mining. *Discovery Science, 9*(20), 17–21.

Batista, G. E., & Monard, M. C. (2003). An analysis of four missing data treatment methods for supervised learning. *Applied Artificial Intelligence, 17*(5--6), 519–533.

Bifet, A. (2013). Mining big data in real time. *Informatica, 37*(1), 15–20.

Chen, M., Mao, S., Zhang, Y., & Leung, V. C. (2014). Big data applications. *Big Data* (pp. 59–79). Springer International Publishing.

Dogan, N., & Zuhal, T. (2013). A comparative analysis of classification algorithms in data mining for accuracy, speed and robustness. *Information Technology and Management, 14*(2), 105–124.

Farhangfar, A., Kurgan, L., & Dy, J. (2008). Impact of imputation of missing values on classification error for discrete data. *Pattern Recognition, 41*(12), 3692–3705.

Fire, M., & Elovici, Y. (2015). Data mining of online genealogy datasets for revealing lifespan patterns in human population. *ACM Transactions on Intelligent Systems and Technology (TIST), 6*(2), 1–22 28.

Gao, H., Liu, X. W., Peng, Y. X., & Jian, S. L. (2015). Sample-based extreme learning machine with missing data. *Mathematical Problems in Engineering, 2015*(2015), 1–11 145156.

Ghannad-Rezaie, M., Soltanian-Zadeh, H., Ying, H., & Dong, M. (2010). Selection–fusion approach for classification of datasets with missing values. *Pattern Recognition, 43*(6), 2340–2350.

Hengpraphrom, K., Wichian, A. N., & Meesad, P. (2010). A comparative study of microarray data classification with missing values imputation. *International Journal of Computer Science and Information Security, 8*(2), 29–32.

Jang, Y. J., & Kwak, J. (2014). Social network service real time data analysis process research. *Frontier and innovation in future computing and communications* (pp. 643–652). Springe.

Jiang, C., & Yang, Z. (2015). CKNNI: an improved knn-based missing value handling technique. *Advanced intelligent computing theories and applications* (pp. 441–452). Springer International Publishing.

Jones, S., Johnstone, D., & Wilson, R. (2015). An empirical evaluation of the performance of binary classifiers in the prediction of credit ratings changes. *Journal of Banking & Finance, 56*, 72–85.

Kang, P. (2013). Locally linear reconstruction based missing value imputation for supervised learning. *Neurocomputing, 118*(22), 65–78.

Kim, K. H., & Tsai, W. (2012). Social comparison among competing firms. *Strategic Management Journal, 33*(2), 115–136.

Kwon, O., & Sim, J. M. (2013). Effects of data set features on the performances of classification algorithms. *Expert Systems with Applications, 40*, 1847–1857.

Li, Y., Li, Z., & Li, L. (2014). Missing traffic data: Comparison of imputation methods. *IET Intelligent Transport Systems, 8*(1), 51–57.

Liu, B., Blasch, E., Chen, Y., Shen, D., & Chen, G. (2013). Scalable sentiment classification for Big Data analysis using Naïve Bayes Classifier. In *Big Data, 2013 IEEE International Conference on* (pp. 99–104). IEEE.

Liu, X. Y., & Zhou, Z. H. (2006). The influence of class imbalance on cost-sensitive learning: an empirical study. In *Data Mining, 2006. ICDM'06. Sixth International Conference on* (pp. 970–974). IEEE.

Liu, Y., & Brown, Steven D. (2013). Comparison of five iterative imputation methods for multivariate classification. *Chemometrics and Intelligent Laboratory Systems, 120*, 106–115.

Loh, W. P., & H'ng, C. W. (2014). Data treatment effects on classification accuracies of bipedal running and walking motions. *Recent Advances on Soft Computing and Data Mining* (pp. 477–485). Johor, Malaysia: Springer International Publishing.

Luengo, J., García, S., & Herrera, F. (2010). A study on the use of imputation methods for experimentation with Radial Basis Function Network classifiers handling missing attribute values: the good synergy between RBFNs and EventCovering method. *Neural Networks, 23*(3), 406–418.

Nia, M. A., Atani, R. E., & Haghi, A. K. (2014). Ubiquitous IoT structure via homogeneous data type modeling. In *Telecommunications (IST), 2014 7th International Symposium on* (pp. 283–288). IEEE.

Okamoto, M. (1963). An asymptotic expansion for the distribution of the linear discriminant function. *The Annals of Mathematical Statistics, 34*(4), 1286–1301.

Orczyk, T., & Porwik, P. (2015). Investigation of the Impact of missing value imputation methods on the k-NN classification accuracy. *Computational Collective Intelligence* (pp. 557–565). Madrid, Spain: Springer International Publishing.

Raudys, S., & Pikelis, V. (1980). On dimensionality, sample size, classification error, and complexity of classification algorithm in pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 2*(3), 242–252.

Rubin, D. (1987). *Multiple Imputation for Nonresponse in Surveys*. New York: John Wiley & Sons.

Safavian, S. R., & Landgrebe, D. (1991). A survey of decision tree classifier methodology. *IEEE Transactions on Systems, Man, and Cybernetics, 21*(3), 660–674.

Silva, J. D. A., & Hruschka, E. R. (2013). An experimental study on the use of nearest neighbor-based imputation algorithms for classification tasks. *Data & Knowledge Engineering, 84*, 47–58.

Sim, J., Lee, J. S., & Kwon, O. (2015). missing values and optimal selection of an imputation method and classification algorithm to improve the accuracy of ubiquitous computing applications. *Mathematical Problems in Engineering, 2015*(538613), 1–14.

Sowe, S. K., Kimata, T., Dong, M., & Zettsu, K. (2014). Managing heterogeneous sensor data on a big data platform: Iot services for data-intensive science. In *Computer Software and Applications Conference Workshops (COMPSACW)* (pp. 295–300). IEEE 38th International.

Su, X., Khoshgoftaar, T. M., & Greiner, R. (2008). Using imputation techniques to help learn accurate classifiers. In *Proceedings of the 20th IEEE international conference tools with artificial intelligence 2008, ICTAI'08*, (pp. 437–444). IEEE.

Suykens, J. A. K., & Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural Processing Letters, 9*(3), 293–300.

Tan, Huachun, Wu, Y., Cheng, B., Wang, W., & Ran, B. (2014). Robust missing traffic flow imputation considering nonnegativity and road capacity. *Mathematical Problems in Engineering*, 1–8 article id 763469.

Tran, C. T., Andreae, P., & Zhang, M. (2015). Impact of imputation of missing values on Genetic programming based multiple feature construction for classification. In *Evolutionary Computation (CEC), 2015 IEEE Congress* (pp. 2398–2405). IEEE.

Twala, B., & Cartwright, M. (2005). Ensemble imputation methods for missing software engineering data. In *Software Metrics 2005, 11th IEEE International Symposium* (pp. 1–10). IEEE.

van Leeuwen, M., & Cardinaels, L. (2015). VIPER–visual pattern explorer. *Machine learning and knowledge discovery in databases* (pp. 333–336). Leuven, Belgium: Springer International Publishing.

Wang, J., Xie, H., & Fisher, J. F. (2011). *Multilevel models: applications using SAS®*. Berlin: Walter de Gruyter.

Wasito, I., & Mirkin, B. (2006). Nearest neighbours in least-squares data imputation algorithms with different missing patterns. *Computational Statistics & Data Analysis, 50*(4), 926–949.

Wrzus, C., Hänel, M., Wagner, J., & Neyer, F. J. (2013). Social network changes and life events across the life span: A meta-analysis. *Psychological Bulletin, 139*(1), 53–80.

Wu, X., Zhu, X., Wu, G. Q., & Ding, W. (2014). Data mining with big data. *IEEE Transactions on Knowledge and Data Engineering, 26*(1), 97–107.

Xiang, Z. L., Yu, X. R., & Kang, D. K. (2015). Experimental analysis of naïve Bayes classifier based on an attribute weighting framework with smooth kernel density estimations. *Applied Intelligence*, 1–10. doi:10.1007/s10489-015-0719-1.

Xiao, J., Zhu, B., Teng, G., He, C., & Liu, D. (2014). One-step dynamic classifier ensemble model for customer value segmentation with missing values. *Mathematical Problems in Engineering, 2014*(2014), 1–15 869628.

Zhang, M. L., & Zhou, Z. H. (2007). ML-KNN: a lazy learning approach to multi-label learning. *Pattern recognition, 40*(7), 2038–2048.

Zhang, Q., Fang, L., Ma, L., & Zhao, Y. (2016). Research on parameters optimization of SVM based on improved fruit fly optimization algorithm. *International Journal of Computer Theory and Engineering, 8*(6), 500–505.