Expert Systems With Applications 115 (2019) 68–94

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Missing value imputation using a novel grey based fuzzy c-means, mutual information based feature selection, and regression model

Amir Masoud Sefidian, Negin Daneshpour*

Faculty of Computer Engineering, Shahid Rajaee Teacher Training University, Tehran, Iran

ARTICLE INFO

Article history:
Received 6 January 2018
Revised 26 May 2018
Accepted 28 July 2018
Available online 30 July 2018

Keywords:
Missing data imputation
Grey relational analysis
Fuzzy c-means
Mutual information
Regression

ABSTRACT

The presence of missing values in real-world data is not only a prevalent problem but also an inevitable one. Therefore, missing values should be handled carefully before the mining or learning process. This paper proposes a novel technique to impute missing data. It employs a new version of Fuzzy c-Means clustering algorithm which benefits from advantages of Grey Relational Grade over Minkowski-like similarity measures. To impute a missing value more accurately, it also performs a local mutual information based feature selection in each cluster to select only highly relevant features. Briefly, missing values are imputed in the following steps. First, the algorithm finds the importance of each missing attribute. Next, input instances are separated into several fuzzy clusters. Then, the algorithm selects clusters which satisfy a minimum condition. After that, it chooses highly dependent features of instances within each cluster using a mutual information based feature selection approach. When the features are selected, regression models will be applied to the selected features of the selected clusters to provide estimations for a missing value. Finally, the missing value is imputed through a weighted average of estimated values obtained from the previous step.

Three well-known evaluation criteria and the accuracy of classification task are used to assess the performance of the proposed method. The experimental results for seven UCI data sets with different missing ratios and strategies indicate that the proposed algorithm outperforms five other imputation methods in general.

1. Introduction

Knowledge discovery has become crucial nowadays for various decision-making processes of modern organizations (Rahman & Islam, 2016). The first step in the knowledge discovery process is gathering data from a single source or multiple sources (Tsai & Chang, 2016). High-quality knowledge is highly dependent on high-quality data (Aydilek & Arslan, 2013). In many applications, collected data sets may suffer from incompleteness, i.e. a data set contains instances that have no value recorded for some attributes, due to the failure in data collection, measurement errors, missing observations, random noise, etc. (Li, Gu, & Zhang, 2010). Imputation algorithms attempt to replace the missing values using estimations derived from statistics of known values or obtained using more sophisticated algorithms (Garciarena & Santana, 2017).

The imputation of missing data is of fundamental importance in many areas such as industry, commerce, and medicine. The inappropriate management of missing data in the analysis may introduce bias and can result in misleading conclusions being drawn from a research study. It can also limit the generalizability and imperil the validity of findings of a research (Wang & Wang, 2010). There is consensus on the importance of the application of imputation methods, especially when data sets with missing data are used as a basis for learning or mining algorithms. The imputation of missing values can enhance capabilities of the model obtained when applying data mining techniques. For instance, results of experiments of Carmona, Luengo, González, and Del Jesus (2012) revealed that the completeness of information presented to the subgroup discovery algorithm, which is a descriptive technique for induction of interesting rules, is crucial. The reason is that most of evolutionary fuzzy systems which are appropriated for subgroup discovery cannot work directly with incomplete data sets. In a comprehensive analysis, Garciarena and Santana (2017) showed that the type of missing data and the choice of the imputation method can directly influence the quality of predictions of different classification algorithms applied to the data. Accurate imputation of missing values in the field of unsupervised learning has always been a challenging problem. For instance,

* Corresponding author.
E-mail addresses: amirmasoud.sefidian@sru.ac.ir (A.M. Sefidian), ndaneshpour@sru.ac.ir (N. Daneshpour).

https://doi.org/10.1016/j.eswa.2018.07.057
0957-4174/

Li et al. (2010) introduced a modified version of fuzzy c-means algorithm which was specially designed to cluster the incomplete data. Since missing data is a prevalent problem in the medical domain, imputation methods are also widely applied in this area (Purwar & Singh, 2015). For example, Mohammed, Naugler, and Far (2016) used multiple imputation (Rubin, 2004) to improve the accuracy of breast tumor classification. Di Nuovo (2011) applied fuzzy c-means imputation in a psychological research environment. The result showed that the fuzzy c-means imputation results in a more effective data imputation than deleting incomplete instances. Guessoum, Laskri, and Lieber (2014) proposed five imputation techniques to enhance the accuracy of results of their case-based decision support system dedicated to the diagnosis of a respiratory disease.

In the last decade, a number of new strategies based on clustering methods have been proposed to solve the problem of missing data imputation. The basic idea of these techniques is to estimate a missing value within a record using the information of the cluster which the missing record located in that cluster. Many of these researches employed hard clustering algorithms such as k-means (Ankaiah & Ravi, 2011; Bu, Chen, Zhang, & Yang, 2016; Gautam & Ravi, 2015; Ishay & Herman, 2015; Patil, Joshi, & Toshniwal, 2010; Zhang, Zhang, Zhu, Qin, & Zhang, 2008). However, fuzzy clustering methods could be a better choice when instances do not belong to a category definitively, as is the case for missing data. Therefore, many soft clustering based imputation methods, such as fuzzy c-means imputation, have been introduced. In the most of these techniques, simple distance metrics such as the Euclidean distance are used as the similarity function. In the authors' opinion, the performance of the fuzzy c-means algorithm can be improved in the context of missing data imputation using a more suitable distance measure. We propose the use of Grey System Theory (GST) (Ju-Long, 1982; Julong, 1989) concepts for this purpose. Moreover, we believe that only highly related features in relation to a missing feature should be used to impute that missing feature. Hence, a feature selection procedure based on the Mutual Information (MI) measure is used to select highly correlated attributes.

To deal with missing values, a novel imputation method, called Grey based Fuzzy c-Means (GFCM) and Mutual Information (MI) based feature selection Imputation method (GFCMI) is developed and proposed in this paper. The main assumption in this paper is that an incomplete instance is more likely to be located in the same cluster of complete instances which can precisely predict that incomplete instance. Compared with the previous works, the main contribution of this paper could be described as follows. (1) To improve the clustering accuracy, a new version of the fuzzy clustering algorithm is proposed that combines the original fuzzy c-means algorithm and grey system theory concepts. The grey relational grade in GST is employed to measure the similarity between input instances and cluster prototypes during the clustering process. This is because there are some advantages for GRG, especially in the situations that there are unknown factors as the problem of missing values. To the best of our knowledge, no such combination had previously been proposed in the context of missing values imputation. (2) We perform a mutual information based feature subset selection in each cluster, locally. The purpose is to find the most related features to a missing feature in each cluster. Other methods usually perform feature subset selection once before starting the imputation process. However, we believe that feature selection in subsets of input data is expected to result in better performance. (3) Two well-known regression-based prediction models, namely, Multiple Linear Regression (MLR) and Support Vector Regression (SVR) are applied to the selected features of instances of each fuzzy cluster. The final imputed value is obtained through a weighted average of predicted values of these models.

The order of selecting missing attributes for imputation can affect the quality of imputation. Hence, we also use a ranking procedure of missing attributes which is employed in van Stein and Kowalczyk (2016) to improve the accuracy of the proposed approach.

The remainder of the paper is organized as follows. Section 2 provides a review of diverse literature about missing value imputation. Section 3 describes the proposed method in detail. Section 4 and Section 5 present the conducted experiments and analyze results given by the proposed approach against five other imputation methods. Finally, Section 6 concludes the paper and highlights possible future developments.

2. Literature review

This section reviews some missing data handling methods. Before doing so, however, it is worth taking some time to review different missing data categories. Generally, missing data can arise from three types of mechanisms (Little & Rubin, 2002; Silva-Ramírez, Pino-Mejías, & López-Coello, 2015; Tian, Yu, Yu, & Ma, 2014):

(i) MCAR (Missing Completely At Random): The probability that the value of an attribute is observed or missed for any record does not depend on any attribute.
(ii) MAR (Missing At Random): The probability that the value of an attribute is observed or missed for any record depends on the value of the other attributes, but not on the value of the missing attribute itself.
(iii) MNAR (Missing Not At Random): Occurs when the probability of the presence of a missing value in an attribute could depend on the value of that attribute itself.

In the past few years, missing value imputation has attracted more and more attentions of researchers. The investigations encompass a broad spectrum of techniques, from simple statistical methods to complex data mining techniques. The most popular imputation method in the statistics is regression imputation (Gelman & Hill, 2006; Zhang, 2011). It is applied by establishing a regression equation for each missing feature, using other features as predictors. Exploitation of existing relationships between the features to approximate the missing data is the main advantage of these methods. However, when there is no relation between independent variables and dependent variables, the performance is poor (Zhang et al., 2008). Different regression models are proposed to impute missing values: multiple linear regression for quantitative variables, logistic regression for a dichotomous dependent variable, and multinomial logistic regression is used to handle categorical variables with more than two categories (Silva-Ramírez, Pino-Mejías, López-Coello, & Cubiles-de-la Vega, 2011). van Stein and Kowalczyk (2016) applied a series of regression models which replace missing values in an iterative manner. It also used the class label of each sample as an extra predictor variable.

Strategies based on artificial neural networks are another type of imputation methods. Multilayer Perceptron (MLP) neural network is the most popular network in this category (García-Laencina, Sancho-Gómez, & Figueiras-Vidal, 2013; Gupta & Lam, 1996; Sharpe & Solly, 1995; Silva-Ramírez et al., 2011). First, it is trained as a nonlinear regression model on the complete portion of input data set. Then, the incomplete data set is imputed by passing each missing instance as the input of the trained network. Silva-Ramírez et al. (2015) applied MLP networks on eighteen real data sets to fill missing values and investigated the impact of different learning rules and parameters of MLP networks on the final performance. Singh, Javeed, Chhabra, and Kumar (2015) and Samad and Harp (1992) employed Self-Organizing Map (SOM) neural networks to fill missing values. Lengthy training times and the challenge

of finding the optimal network's topology and its parameters are main disadvantages of these imputation methods (Han, Kamber, & Pei, 2011).

The Nearest Neighbors Imputation (NNI) algorithm (Batista, Monard et al., 2002; Bose, Das, Dutta, & Chattopadhyay, 2012; García-Laencina, Sancho-Gómez, Figueiras-Vidal, & Verleysen, 2009; Pan & Li, 2010; Tutz & Ramzan, 2015) is another prevailing nonparametric imputation technique. In this method, each missing value in a record is inferred from the closest instances of that missing record in the whole data set. Usually, the most frequent value among all neighbors is selected for nominal features, and the mean value is used for numerical attributes (Jiang & Yang, 2015). It can be used for both numerical and categorical data simply by selecting an appropriate distance function. The NNI is simple since it does not require to create predictive models such as neural networks. Therefore, it avoids computational time on modeling (Jiang & Yang, 2015; Wu, Wun, & Chou, 2004). Moreover, it is easy to implement and often provides good performance compared to other imputation strategies (Jiang & Yang, 2015). Krishnamoorthy, Kumar, and Neelagund (2014) applied 1NN imputation method on records which have missing degree less than defined threshold. Zhang, Zhu, Zhang, Qin, and Zhang (2007) and Huang and Lee (2004a) employed a grey based kNN method to fill the missing values. They used grey relational grade to determine the nearest neighbors of a record containing missing values. A variation of nearest neighbors imputation is weighted nearest neighbors imputation in which the distance of each missing instance from its neighbors is used to approximate the missing value in some way (Troyanskaya et al., 2001). Pan, Yang, Cao, Lu, and Zhang (2015) proposed a method called, Feature Weighted Grey k Nearest Neighbors (FWGKNN) in which it uses a mutual information weighted grey relational analysis as the similarity metric in the kNN method to determine the nearest neighbors of a missing instance. Beretta and Santaniello (2016) evaluated the performance of various types of NNI over data sets with different patterns and degrees of missingness. The results have shown that kNN usually outperforms 1NN in terms of accuracy of imputation.

The NNI method can be expensive for large data sets since it must perform an exhaustive search in the data set for each missing record. Moreover, the selection of an appropriate similarity metric and an optimal value for k in kNNI can be challenging (Batista & Monard, 2003). Since kNNI usually uses only complete cases to fill missing values, the performance of this method is limited when the amount of missing data is high. Van Hulse and Khoshgoftaar (2014) proposed an alternative version of kNNI which allows some incomplete instances to participate in the nearest neighbors imputation procedure.

Clustering methods as one of the most wide-spreading techniques in data mining have helped to improve imputation results (Fujikawa & Ho, 2002). Given a set of objects, the objective of clustering is to divide the data set into groups based on similarity of objects and to minimize the intra-cluster dissimilarity (Aydilek & Arslan, 2013). The basic idea of these techniques is to estimate a missing record using similar records to that record which are located in the same cluster. In the k-means clustering imputation, all of the records are clustered, first. Next, the information of instances belonging to the same cluster of the missing record is used to impute missing attributes. Jiang and Yang (2015) integrated the k-means clustering algorithm and kNN algorithm to impute missing values in a data set. Patil et al. (2010) applied the k-means clustering method on input instances at first. Then, they used the nearest neighbors method to detect the nearest instance and its distance to the missing instance in the same cluster. Finally, it took the average of the centroid value and the weighted distance as the estimated value of the missing attribute.

Li, Deogun, Spaulding, and Shuart (2004) borrowed the idea of a fuzzy version of the k-means algorithm and applied it to impute the missing data. In the fuzzy clustering, the likelihood that a sample belongs to a certain cluster is expressed by a membership function (Li et al., 2004). Experiments denote that the fuzzy c-means often outperforms the basic k-means method. Rahman and Islam (2016) performed a fuzzy clustering on input samples in order to find similar records, first. Then, they applied a fuzzy expectation maximization algorithm for the imputation. The main idea is that instances in the same cluster are very similar to each other and the correlations for attributes are high. Hence, the imputation accuracy is likely to be high when an imputation algorithm uses the records which are in the missing record's cluster rather than the whole set of records.

In Tian et al. (2014) incomplete instances were divided into several clusters. Then, a record with the least number of missing values was assigned to the closest cluster using a grey based distance metric. Finally, each missing value of a missing record was estimated by an entropy-based multiple imputation. Bu et al. (2016) performed a hierarchical clustering-based feature selection to reduce dimensions of a high dimensional data set. Next, it clustered data using a parallel k-means algorithm. Finally, samples in the same cluster with the missing record were utilized to estimate missing features of that record. Ankaiah and Ravi (2011) employed a two-step imputation method. In the first step, the k-means algorithm was employed to replace missing values with cluster centers. The second stage refined the estimated values using MLP networks. The fuzzy c-means algorithm was replaced in the first step of this method in Nishanth, Ravi, Ankaiah, and Bose (2012). Ayuyev, Jupin, Harris, and Obradovic (2009) employed a new clustering algorithm that involves the number of common neighbors among k nearest neighbors of two records in distance calculation between those records.

Hybrid techniques also have been proposed to fill in missing values in order to improve the imputation accuracy. Nelwamondo, Golding, and Marwala (2013) combined dynamic programming, neural networks, and genetic algorithm. Aydilek and Arslan (2013) hybridized the fuzzy c-means, support vector regression and genetic algorithm to estimate missing values. Saravanan and Sailakshmi (2015) employed fuzzy possibilistic c-means to handle noisy data effectively in the former method. Gautam and Ravi (2015) proposed two hybrid imputation methods involving Particle Swarm Optimization (PSO), Evolving Clustering Method (ECM), and Auto-Associative Extreme Learning Machine (AAELM). Tsai and Chang (2016) proposed two strategies which use an instance selection procedure to filter out noisy or outliers data from a given data set to achieve better imputation results.

There are many other imputation techniques which have been proposed in the literature such as genetic algorithm (Lobato et al., 2015), genetic programming (Tran, Zhang, & Andreae, 2016), Bayesian networks (Di Zio, Scanu, Coppola, Luzi, & Ponti, 2004; Duan, Yue, Qian, & Liu, 2013; Hruschka, Hruschka, & Ebecken, 2007), k-decision tree based imputation (Rahman & Islam, 2013), recursive partitioning (Doove, Buuren, & Dusseldorp, 2014), and so on. However, Kwon and Sim (2013); Loh and H'ng (2014) argue that there is no single superior imputation algorithm to replace all missing data in a data set because all imputation methods are affected by characteristics of the data set and the missing values. Sim, Kwon, and Lee (2016) listed the missing ratio, distribution of missing values, and data set characteristics (such as the degree of imbalance, the size of the data set, and the number of features (Sim, Lee, & Kwon, 2015)) as main factors which can affect the performance of imputation.

This paper proposes an imputation strategy which fills up missing values using a combination of Grey based Fuzzy c-Means (GFCM), mutual information based feature selection, and regres-

sion models. The complete details of the proposed method are explained in next sections.

3. The proposed method

This section explains steps of the proposed imputation method which makes use of a modified version of fuzzy clustering algorithm and a mutual information based feature selection approach. Before delving into the details of the proposed method, however, a brief overview of basic concepts which are used in the proposed method is covered in Section 3.1. Section 3.3 discusses the complexity order of the proposed method.

3.1. Preliminary

3.1.1. The fuzzy c-Means algorithm

Clustering techniques are mostly unsupervised methods that can be used to decompose data into subgroups or clusters based on similarities among the instances (Pinzon-Morales, Baquero-Duarte, Orozco-Gutierrez, & Grisales-Palacio, 2011). They can be grouped into two categories, namely hard (crisp) clustering and fuzzy (soft) clustering. In the hard clustering, such as the k-means algorithm, an instance x_i belongs to one and only one cluster to which x_i is the most similar (Rahman & Islam, 2016). In the fuzzy clustering methods, however, instances on the boundaries between several clusters are not forced to fully belong to one of the clusters. They are allowed to belong to several clusters simultaneously (Szczepaniak & Lisboa, 2012). Each instance has a degree of membership between 0 and 1 indicating its partial membership.

There are some advantages for fuzzy clustering in comparison with original k-means to impute the missing data. In many situations, the fuzzy clustering is more natural than the hard clustering. It provides a better description tool when instances are not well-separated, as is the case for missing data problem. Furthermore, the original k-means algorithm may be trapped in a local minimum status if the initial points are not selected properly. By contrast, continuous membership values in the fuzzy clustering make the resulting algorithms less sensitive to get stuck in a local minimum (García, Luengo, & Herrera, 2015; Li et al., 2004).

The Fuzzy c-Means (FCM) algorithm (Bezdek, Ehrlich, & Full, 1984) is the most well-known soft clustering technique. The FCM algorithm partitions a set of input data $\{x_1, x_2, \ldots, x_n\}$ into c fuzzy clusters $\{C_1, C_2, \ldots, C_c\}$ by minimizing the following distance-based objective function:

$$\mathbf{J}(\boldsymbol{\delta}, \mathbf{V}) = \sum_{i=1}^{n}\sum_{k=1}^{c}(\delta_{ik})^{m'}\|x_i - v_k\|^2 \tag{1}$$

where $x_i = [x_i^1, x_i^2, \ldots, x_i^m]^T$ represents an input instance and x_i^j refers to the jth attribute value of x_i. $\mathbf{V} = [v_{ij}]_{c\times m}$ is a matrix of cluster prototypes (centroids) and v_k denotes the kth cluster prototype. $\|.\|$ denotes the Euclidean norm. It is used to measure the similarity of data object x_i to the center vector v_k. $m' \in (1, \infty)$ is a fuzzification parameter. It specifies how much the clusters can overlap with one another. $\boldsymbol{\delta} = [\delta_{ik}]_{n\times c}$ indicates the partition (membership) matrix and δ_{ik} is the likelihood value that expresses the degree to which x_i belongs to the kth cluster (C_k), $\forall i, k$: $\delta_{ik} \in [0, 1]$. The higher value of δ_{ik} expresses the higher association between x_i and C_k. The total association of x_i with c clusters is equal to 1; that is, δ_{ik} satisfies the following condition:

$$\sum_{k=1}^{c}\delta_{ik} = 1, \quad \text{for } i = 1, \ldots, n \tag{2}$$

In the FCM algorithm, the objective function is iteratively optimized over the membership degrees and the cluster prototypes. The required conditions for minimizing (1) with the constraint (2) are following updating equations for the prototypes and the membership matrices, respectively (Bezdek, 2013):

$$v_k = \frac{\sum_{i=1}^{n}\delta_{ik}^{m'}.x_i}{\sum_{i=1}^{n}\delta_{ik}^{m'}}, \quad \text{for } k = 1, 2, \ldots, c \tag{3}$$

and

$$\delta_{ik} = \left[\sum_{j=1}^{c}\left(\frac{\|x_i - v_k\|}{\|x_i - v_j\|}\right)^{\frac{2}{m'-1}}\right]^{-1}, \quad \text{for } k = 1, 2, \ldots, c, \quad i = 1, 2, \ldots, n \tag{4}$$

The required precision for the membership matrix determines the number of iterations completed by the FCM algorithm. This precision is calculated using the membership matrix from one iteration to the next iteration as follows:

$$\|\boldsymbol{\delta}^{(r+1)} - \boldsymbol{\delta}^{(r)}\| \le \varepsilon \tag{5}$$

where $\boldsymbol{\delta}^{(r)}$ and $\boldsymbol{\delta}^{(r+1)}$ are the partition matrix at iteration r and $r+1$, respectively. $\|.\|$ denotes the matrix norm operator. The iterations will be stopped when the difference between two successive partitions is less than a predefined level of accuracy, ε.

The conventional FCM uses the Euclidean distance as a distance measure. In this paper, a modified version of the original FCM which employs Grey Relational Analysis is used to improve imputation results.

3.1.2. Grey system theory

Grey System Theory (GST), introduced by Deng (Ju-Long, 1982; Julong, 1989), has been proposed for tackling uncertain systems with partially known and partially unknown information. It can extract valuable information from partial data (Pan et al., 2015). Hence, it has been widely applied to solve problems of fields containing unknown factors. As a measurement method in GST, Grey Relational Analysis (GRA) can determine relationships between a referential observation (instance) and a set of compared observations by calculating Grey Relational Coefficient (GRC) and Grey Relational Grade (GRG) for finite sequences. Consider a set of observations $\{x_0, x_1, x_2, \ldots, x_n\}$, where x_0 is a reference instance and $x_1, x_2, \ldots, x_n$ are the compared instances. Each observation x_i has m attributes and is denoted by $x_i = (x_i(1), x_i(2), \ldots, x_i(m))$, $i = 0, 1, 2, \ldots, n$. The GRC between two instances is defined as follows:

$$GRC(x_0(p), x_i(p)) = \frac{\Delta_{\min} + \zeta\Delta_{\max}}{|x_0(p), x_i(p)| + \zeta\Delta_{\max}} \tag{6}$$

where $\Delta_{\min} = \min_{\forall j}\min_{\forall k}|x_0(k), x_j(k)|$ and $\Delta_{\max} = \max_{\forall j}\max_{\forall k}|x_0(k), x_j(k)|$ for $i = j = 1, 2, \ldots, n$ and $k = p = 1, 2, \ldots, m$. $\zeta \in [0, 1]$ is a distinguishing coefficient that controls the level of differences with respect to the relational coefficient (Tian et al., 2014). For categorical features, the value of the GRC is defined as follows (Pan et al., 2015):

$$GRC(x_0(p), x_i(p)) = \begin{cases} 1, & \text{if } x_0(p) \text{ and } x_i(p) \text{ are the same.} \\ 0, & \text{if } x_0(p) \text{ and } x_i(p) \text{ are different.} \end{cases} \tag{7}$$

Finally, GRG can be calculated as follows:

$$GRG(x_0, x_i) = \frac{1}{m}\sum_{k=1}^{m}GRC(x_0(k), x_i(k)) \tag{8}$$

where $i = 1, 2, \ldots, n$. It is clear that GRG takes a value between zero and one. It represents the level of similarity of two instances on a set of features. If $GRG(x_0, x_i) > GRG(x_0, x_j)$, it shows that the level of similarity between x_0 and x_i is larger than that of x_0 and x_j.

Generally, similarity functions such as Euclidean distance can be used to determine the "nearness" between two instances. However,

Minkowski-like distances are mainly suitable for some application domains (Zhang et al., 2007). GRA offers some advantages rather than other metrics. For example, it offers a normalized measuring function (Normality). Moreover, GRA gives whole relational orders because of its wholeness over the entire relational space (Ju-Long, 1982; Pan et al., 2015). The Minkowski-like distances only consider two compared instances to calculate the similarity between them. By contrast, GRG takes the whole space of all instances into account in order to calculate the distance between two instances (Wholeness). Experimental results demonstrate that GRA is superior to Euclidean distance and its variants as a nearness measure (Huang & Lee, 2004b; Pan et al., 2015; Zhang, 2012). Therefore, it is expected that GRG measures the distance between two samples more precisely than Euclidean distance.

3.1.3. Mutual information

The Mutual Information (MI) between two random variables X and Y, denoted by $I(X; Y)$, is a symmetric quantity that measures the mutual dependency between X and Y from the perspective of information theory (Folch-Fortuny, Villaverde, Ferrer, & Banga, 2015; Lv, Zhao, Liu, & Wang, 2016). Intuitively, it measures how much one random variable tells about another. The main advantage of MI is that it does not assume any property of the dependence between variables, such as linearity or continuity. Hence, it is more general than linear measures such as the correlation coefficient (Folch-Fortuny et al., 2015).

On a discrete domain, the formal definition of the mutual information of two random variables X defined on an alphabet χ and Y defined on an alphabet γ is given by:

$$I(X;Y) = \sum_{x\in\chi}\sum_{y\in\gamma} p(x,y)\log\frac{p(x,y)}{p(x)p(y)} \tag{9}$$

where $p(x) = Pr\{X = x\},\ x\in\chi$ and $p(y) = Pr\{Y = y\},\ y\in\gamma$ are the marginal probability mass function (pmf) of X and Y, respectively. The $p(x, y)$ denotes the joint pmf of X and Y. It also can be straightforwardly extended to a continuous domain:

$$I(X;Y) = \int_x\int_y p(x,y)\log\frac{p(x,y)}{p(x)p(y)}dxdy \tag{10}$$

where $p(x, y)$ is the joint probability density function (pdf) of X and Y, and $p(x)$ and $p(y)$ are the marginal pdf of X and Y, respectively. The higher mutual information value indicates the stronger dependency between X and Y.

By computing the MI value for each pair of m attributes in a data set $\mathfrak{D}$, a symmetric $m\times m$ mutual information matrix can be constructed in which entry i, j of the matrix denotes MI value between attributes $\mathcal{A}_i$ and $\mathcal{A}_j$ in $\mathfrak{D}$. The following notational conventions are used in the rest of this paper. Symbol $\mu(\mathfrak{D})$ denotes the mutual information matrix for data set $\mathfrak{D}$. Symbol $\mu(\mathfrak{D})_{ij}$ refers to the entry in the ith row and jth column of the mutual information matrix of data set $\mathfrak{D}$. Eq. (11) shows how the mutual information matrix is calculated.

$$\mu(\mathfrak{D}) = \begin{bmatrix} 1 & \mu_{12} & \cdots & \mu_{1m} \\ \mu_{21} & 1 & \cdots & \mu_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \mu_{m1} & \mu_{m2} & \cdots & 1 \end{bmatrix}_{m\times m} \tag{11}$$

where μ_{ij} is equal to the mutual information value between two attributes $\mathcal{A}_i$ and $\mathcal{A}_j$ in $\mathfrak{D}$, i.e., $I(\mathcal{A}_i; \mathcal{A}_j)$.

3.2. The main steps of the proposed method

This section describes details of the proposed imputation method. There are two main points of the proposed method. First, the imputation accuracy for a missing record is likely to be high when the records which participate in the imputation process are very similar to that missing record. Second, identification of features of data which have high correlations and applying regression models within these features is expected to produce a better imputation result. Therefore, the main aim of the proposed method is to find a set of similar records with high dependencies for a missing record and then apply regression imputation techniques within the group to estimate missing values for that record.

In order to find similar records, we perform a modified version of fuzzy c-means clustering which employs GST concepts, called Grey based Fuzzy c-Means (GFCM), on the input data set. As mentioned in Section 3.1.2, recent research studies have shown that GRA can provide more effective results for nearness measuring in comparison with Euclidean distance. To find a set of highly correlated features for a missing feature, a mutual information based feature subset selection is performed.

Suppose that an incomplete data set $\mathfrak{D}$, consists of n instances where each instance $x\in\mathfrak{D}$ is described by m numerical attributes $\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_m$. The x_i^j symbol refers to jth attribute of ith instance in $\mathfrak{D}$. Algorithm 1 shows the general procedure of the Grey based Fuzzy c-Means and Mutual Information based feature selection Imputation method (GFCMI). To impute missing values in a data set, the proposed method consists of steps which are explained in next subsections.

3.2.1. Calculating the importance of each missing attribute

The order of imputation of missing features can affect the imputation results. Therefore, the proposed feature ranking system in van Stein and Kowalczyk (2016) is used. The algorithm finds the priority of each missing attribute for imputation process using a Random Forest model (Breiman, 2001) (lines 1 and 2 of Algorithm 1). The importance score for an attribute $\mathcal{A}_i$ is computed by averaging the difference in Out-of-Bag (OOB) error before and after the permutation of values of $\mathcal{A}_i$ in the training data of all trees placed in a random forest model. The feature with the highest priority is imputed first, the attribute with the lowest priority is imputed last.

3.2.2. Preliminary imputation, data clustering, and cluster selection

So far, it has been determined that records should be imputed in what order. Henceforth, it is explained that how each missing record is imputed. As mentioned before, missing values of an instance are filled up with those plausible values that are generated by applying regression models on the instances belong to a set of fuzzy clusters. In order to impute a missing value, the proposed method firstly divides the incomplete data set into c fuzzy clusters using a new version of Fuzzy c-Means (FCM).

Like many other learning algorithms, the FCM algorithm is not able to handle incomplete data sets. Therefore, it is necessary to provide a preliminary estimation of missing values of the input data set $\mathfrak{D}$ and then pass it to the FCM algorithm. Hence, the incomplete input data set must be imputed with the help of an imputation method (line 3). In this study, the basic mean imputation method is used to obtain the preliminary imputed data set. That is, a missing value on a certain feature $\mathcal{A}_j$ is replaced by the mean of available values in the feature $\mathcal{A}_j$.

After performing a preliminary imputation, the imputed data set is divided into c fuzzy clusters (line 4). As mentioned before, a new version of fuzzy clustering technique (Grey based Fuzzy c-Means) is used for this purpose. The GFCM algorithm is based on the conventional FCM which utilizes GRG to assign each input instance to each of c cluster prototypes with a membership degree. To compute GRG, we use a cluster prototype as the reference instance and the input data as comparative instances.

Algorithm 2 shows the GFCM algorithm. The steps of the GFCM algorithm are as follows: First, it randomly initializes cen-

Algorithm 1: General steps of the proposed method.

Input:
$\mathfrak{D}$: A data set with missing values containing n instances, defined on schema $(\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_m)$
$\delta_{\min}$: Minimum membership threshold to involve a cluster in the imputation process
c: Number of fuzzy clusters
θ: Minimum mutual information value for feature selection process

Output:
$\mathfrak{D}_{imputed}$: Imputed data set

$\mathbb{A} \leftarrow$ Find a set of attributes in $\mathfrak{D}$ containing missing values
Calculate the priority of each missing attribute $\mathcal{A} \in \mathbb{A}$ using a Random Forest model
$\mathfrak{D}_{simpleImputed} \leftarrow$ Impute $\mathfrak{D}$ using the mean imputation method
$\boldsymbol{\delta}, \mathbf{V} \leftarrow$ GFCM($\mathfrak{D}_{simpleImputed}$, c, m')
Assign each instance x_i to the most probable cluster using the partition matrix
$\mathfrak{D}_{imputed} \leftarrow \mathfrak{D}$
while $\mathbb{A} \neq \emptyset$ **do**
  $j \leftarrow$ Index of the most important attribute in $\mathbb{A}$
  $\mathfrak{D}_j \leftarrow \{x_i \in \mathfrak{D}_{imputed} | x_i^j = NULL\}$
  foreach $x_i \in \mathfrak{D}_j$ **do**
    $estimations \leftarrow [\,]$
    $memberships \leftarrow [\,]$
    $c' \leftarrow 0$
    for $k \leftarrow 1$ **to** c **do**
      if $\delta_{ik} \geq \delta_{\min}$ **then**
        $c' \leftarrow c' + 1$
        $memberships[k] \leftarrow \delta_{ik}$
        $selectedFeatures \leftarrow \{\}$
        $\mu \leftarrow$ Compute the mutual information matrix for cluster C_k
        for $p \leftarrow 1$ **to** m **do**
          if $p \neq j$ **then**
            if $\mu_{jp} \geq \theta$ **then**
              $selectedFeatures \leftarrow (p \cup selectedFeatures)$
            end
          end
        end
        if $selectedFeatures = \emptyset$ **then**
          $selectedFeatures \leftarrow \{1, 2, \ldots, m\}$
        end
        Fit a regressor model using $\mathcal{A}_j$ as dependent variable and $\mathcal{A}_t (\forall t \in selectedFeatures)$ as independent variables on C_k
        $estimations[k] \leftarrow$ Estimate x_i^j using the fitted regressor
      end
    end
    $x_i^j \leftarrow \frac{\sum_{i=1}^{c'} memberships[i] \times estimations[i]}{\sum_{i=1}^{c'} memberships[i]}$
  end
  $\mathbb{A} \leftarrow \mathbb{A} \backslash \mathcal{A}_j$ ▷ Delete $\mathcal{A}_j$ from $\mathbb{A}$
end
return $\mathfrak{D}_{imputed}$

Algorithm 2: Grey based Fuzzy c-Means (GFCM).

Input:
$\mathfrak{D} = \{x_1, x_2, \ldots, x_n\}$: Input data set
m': Fuzzification parameter, $m' \in [1, \infty)$
c: Number of clusters
maxIter: Maximum number of iterations

Output:
$\boldsymbol{\delta}$: Partition (membership) matrix
$\mathbf{V}$: Cluster prototypes

1 Randomly initialize the partition matrix, $\boldsymbol{\delta}$, such that $\sum_{i=1}^{c} \delta_{ik} = 1, \quad \text{for} \quad k = 1, \ldots, n$
2 Randomly initialize set of cluster centers, $\mathbf{V} = \{v_1, v_2, \ldots, v_c\}$
3 $r \leftarrow 1$
4 **repeat**
5 Calculate the c centers using: $v_k \leftarrow \frac{\sum_{i=1}^{n} \delta_{ik}^{m'} . x_i}{\sum_{i=1}^{n} \delta_{ik}^{m'}}, \qquad k = 1, 2, \ldots, c$
6 Update the partition matrix as follows: $\delta_{ik}^{(r+1)} \leftarrow \left[\sum_{j=1}^{c} \left(\frac{1 - GRG(v_i, x_k)}{1 - GRG(v_i, x_j)} \right)^{\frac{2}{m'-1}} \right]^{-1}, \qquad k = 1, 2, \ldots, c, \quad i = 1, 2, \ldots, n$
7 $r \leftarrow r + 1$
8 **until** $\|\boldsymbol{\delta}^{(r+1)} - \boldsymbol{\delta}^{(r)}\| \leq \varepsilon$ ***or*** $r \geq maxIter$;
9 **return** $\boldsymbol{\delta}$, $\mathbf{V}$

ter prototypes ($\boldsymbol{V}$) and memberships matrix ($\boldsymbol{\delta}$) (lines 1 and 2 of Algorithm 2). Second, it iteratively tries to minimize the sum of distances for each instance from the centroid of the cluster to which the instance belongs to. This is done by updating the memberships and prototypes matrices (lines 5 and 6 of Algorithm 2). Finally, the process is terminated once the difference between two successive partitions is less than a predefined threshold ε or when it reaches to a maximum predefined number of iterations specified by *maxIter* parameter. As mentioned in Section 3.1.2, GRA has been experimentally proven to be more appropriate for determining the similarity between two instances than Euclidean-like distances because of its properties. Hence, we use GRG to measure the similarity between cluster prototypes and input instances during the updating process of prototypes and memberships matrices. The GRG value between a reference instance and a compared instance quantifies the similarity between those instances, while the FCM algorithm needs the dissimilarity value of two samples in its iterative steps. Therefore, the value (1-GRG) has been replaced as the distance function in the updating formula of partition matrix.

After the end of the clustering process, a membership matrix is obtained which describes the membership degree of each instance to each cluster. Each instance can be assigned to the cluster which has the most likelihood. That is, having the membership matrix, each instance x_i is assigned to cluster C_{k^*} where

$$k^* = \underset{k}{\operatorname{argmax}} (\delta_{ik}), \quad k \in [1, c] \tag{12}$$

In next steps, each missing instance will be estimated using plausible values generated from the formed fuzzy clusters. However, the algorithm does not use all of the formed clusters' information. A missing instance is estimated only using the clusters which the missing instance belongs to them with a minimum membership degree. This is because we want to use only clusters which contain instances with highly related information according to the missing instance. Using highly related instances' information leads to improve the performance of imputation. The minimum

membership degree is determined using a threshold parameter. It is denoted by $\delta_{\min}$ in the algorithm. The proper value for $\delta_{\min}$ directly depends on the number of clusters. We set the $\delta_{\min}$ value to $1/c$ throughout this work, where c is the number of clusters. This assignment ensures that all of the clusters have a chance to participate in the imputation process of a missing instance when it equally belongs to all clusters. In summary, the output of this step is a subset of clusters that satisfies the minimum membership condition.

3.2.3. Feature subset selection in each selected cluster

After choosing the desired clusters, a feature subset selection method is applied to each cluster. Indeed, the algorithm does not use the whole set of features of data in each cluster. Rather, it employs a mutual information based approach to select highly associated attributes. In this approach, only the features which have a minimum mutual information dependence in proportionate to the missing attribute are selected to involve in the imputation process. To do this, a minimum mutual information threshold is used, which is denoted by θ, to select appropriate attributes. Specifically, in each selected cluster, the algorithm only selects features that have a mutual dependence greater than θ value. The desired features of instances within a cluster is selected with the help of the mutual information matrix computed for that cluster.

Lines 18–26 of Algorithm 1 show the feature selection procedure. In a certain situation, if there is no feature that satisfies the minimum mutual information condition (usually occurs in high values of θ), the whole feature set will be participated in the imputation process (lines 27–29). There is an advantage for local feature subset selection in each cluster; after the clustering process, the instances in a cluster probably are more correlated on a subset of their features, whereas it might not be true for that subset in another cluster. Therefore, it is better to perform the feature subset selection in each cluster, separately.

3.2.4. Fitting a regression model on the selected clusters and features

The next step of the proposed algorithm is the estimation of each missing value using the selected features in each selected cluster. The algorithm replaces the missing value by calculating a weighted average of estimated values which are obtained from each cluster selected in the cluster selection step. To be more precise, if there are c' ($c' \leq c$) numbers of candidate clusters, c' numbers of imputed values will be obtained for x_j^i. The missing value of a missing instance in each elected cluster is estimated by fitting a regression model on the selected features of that cluster. The missing attribute ($\mathcal{A}_j$) is used as the dependent variable and the other selected features are used as independent variables to fit the regression model (line 30).

In this study, two different types of regression models are used as the estimator model: Multiple Linear Regression (MLR) and Support Vector Regression (SVR). The linear regression is the simplest method to solve the regression problem where the regression is a linear function of the input. It often outputs fair results compared with other regression algorithms. Support Vector Machines (SVM) are widely used in classification and regression problems. They have been successfully applied to various real-world problems including bioinformatics, computer vision, data mining and knowledge discovery (Chang, Guo, Lin, & Lu, 2010). Support Vector Regression (SVR) is a kernel-based prediction model that extends the linear regression to nonlinear regression (Wang, Yeung, & Lochovsky, 2008).

3.2.5. Weighted voting for the final estimation

The last step of the GFCMI algorithm is the prediction of a missing value using the estimated values obtained from the previous step. A weighted voting approach is employed to find the final imputed value for the missing value x_i^j. The weight of each estimated value for a missing instance in a cluster is equal to what extent the missing instance belongs to that cluster. This means that the final imputation value is computed through a weighted average of all c' imputed values which are estimated from each selected cluster and membership degrees as the weights (line 34). Therefore, the weight of kth($1 \leq k \leq c'$) estimation is equal to δ_{ik}. The stronger the membership of x_i in C_k, the more impact of kth estimation on the final imputed value.

The repetitions of algorithm's steps are continued until all of the missing attributes are imputed. As the algorithm suggests, after the imputation of each missing attribute, that attribute will be used for estimation of other missing attributes in next iterations. Indeed, the input data set ($\mathfrak{D}$) is updated after each iteration of the **while** loop.

3.3. Time complexity analysis

This section analyses the time complexity of the GFCMI algorithm considering both types of GFCMI algorithm, i.e., SVR and MLR. Let n be the number of input instances, m be the number of features, and c be the number of clusters. Suppose that there are n_i missing records and n_c complete records ($n = n_i + n_c$). The computational complexity of GFCMI is a combination of three main processes as follows: (1) Calculating importance of each missing attribute (line 2 of Algorithm 1). (2) Clustering data using the proposed GFCM algorithm (line 4)). (3) Imputing missing data using information of each fuzzy cluster (lines 7- 37).

(1) As mentioned in Section 3.2.1, the GFCMI algorithm uses a random forest model to find the importance of each missing attribute. The building of a random forest has the complexity order of $O(mn\log n)$.

(2) The next step of GFCMI algorithm is dividing the data set into c fuzzy clusters using the proposed grey based fuzzy c-means (GFCM). The time complexity of the original FCM algorithm is $O(nc^2m)$ (Kolen & Hutcheson, 2002). The original FCM uses the Euclidean distance to update each entry of partition matrix which requires m operations. We use GRG instead of Euclidean distance to update partition matrix and the time complexity of calculating GRG is $O(mn)$ (Pan et al., 2015). Hence, the time complexity of the GFCM clustering method is $O(n^2c^2m)$.

(3) In the imputation step, the algorithm replaces missing values with the estimated values obtained from each selected cluster. The computational complexity of imputation in each cluster consists of two main components. One is that of computing the mutual information matrix to perform feature selection in the cluster. The size of mutual information matrix is $m \times m$ and each entry demands a computational time of $O(n_1\log n_1)$ (Evans, 2008), where n_1 represents size of that cluster. The second component is that of applying one of Multiple Linear Regression (MLR) or Support Vector Regression (SVR) models to the selected features of instances of the cluster. The computational complexity of computing the coefficients of MLR and SVR regression equations for a cluster is $O(n_1m_1^2)$ and $O(n_1^3)$, respectively (Wang et al., 2008). m_1 denotes the number of selected features in the cluster. Assuming that c' ($c' \leq c$) clusters are selected in the cluster selection process, this step incurs a cost of $O(c'(n_1m^2\log n_1 + n_1m_1^2))$ and $O(c'(n_1m^2\log n_1 + n_1^3))$ for MLR type and SVR type of GFCMI, respectively. Since the imputation process in each cluster does not need to the other clusters' information, these complexities could be reduced to $O(n_1m^2\log n_1 + n_1m_1^2)$ and $O(n_1m^2\log n_1 + n_1^3)$ when the computations of each cluster are performed in parallel.

Feature ranking step and clustering step (steps 1 and 2) are done just once throughout the whole imputation process. The third step is applied repeatedly n_i times. From the above analysis, it

Table 1
The data sets used in the experiments.

Data set	#Instances	#Attributes
Iris	150	4
Wine	178	13
Glass	214	10
Haberman	306	3
Wholesale Customers	440	8
Chess	3196	36
Adult	48,842	14

is clear that the complexity of the proposed algorithm is about $O(mn\log n + n^2c^2m + n_i(n_1m^2\log n_1 + n_1m_1^2))$ for the MLR type and $O(mn\log n + n^2c^2m + n_i(n_1m^2\log n_1 + n_1^3))$ for the SVR type. In the worst case scenario, $m_1 = m$, $n_i = n$, and $n_1 = n$. Typically, since $c << n$ and $m << n$ in practical cases, the overall time complexity of GFCMI is about $O(n^2m^2\log n)$ and $O(n^4)$ for the MLR type and the SVR type, respectively. The complexities of basic mean imputation, kNNI, MLPI, FCMI, and IARI (i.e., the comparative techniques that are used in the experiments of this study) are estimated as $O(m)$, $O(mn^2)$, $O(nmoh^k)$ (for an MLP with k hidden layers, each containing h neurons, and o output neurons), $O(nmc^2)$, and $O(n^4)$, respectively. Although the GFCMI method needs higher computation time compared to the other methods, better imputation quality generally has a higher priority in missing value imputation (Deb & Liew, 2016; Gan, Liew, & Yan, 2006) because it can substantially enhance the overall quality of obtained results and/or the time required for the actual mining (Han et al., 2011). Moreover, it is done only once before the main mining or learning process.

4. Experimental design

This section presents the details of our experimental design. It includes methods, data sets, and criteria that have been used in the comparisons. The source codes employed in this work for all the different imputation methods were implemented in Python with the help of some Scikit-Learn (Pedregosa et al., 2011) packages. To eliminate the randomness factor, each experiment was repeated 20 times with different random seeds. The averaged performance of these 20 repetitions was reported as the final result in this paper.

As mentioned in Section 3.2.4, we use two different regression models for the prediction step, i.e., Multiple Linear Regression and Support Vector Regression. The Linear Ridge Regression (Hoerl & Kennard, 1970) which uses the linear least squares function as the cost function and performs L_2 regularization with regularization parameter $\lambda = 1$ was used as the Multiple Linear Regression (MLR) model to avoid the overfitting problem. For the SVR model, default parameters $\varepsilon = 0.1$ and $C = 1$ were set in the implementations. Moreover, the popular Radial Basis Function (RBF) kernel was selected to train SVR models.

4.1. Data sets and data preprocessing

Our experiments were conducted on seven real data sets from UCI machine learning repository (Lichman, 2013). The data sets are Iris, Wine, Glass, Haberman, and Wholesale Customers, Chess, and Adult. These data sets have been frequently used to test imputation methods. A brief description of each data set is provided in Table 1. None of these data sets have missing values. The complete data sets were used so as to allow a comparison of results.

To make the comparison of results over various data sets meaningful, values for each attribute of a data set were transformed using the Min-Max normalization (Han et al., 2011). It scales each attribute individually to the range [0, 1]. To do this, each entry of a data set, i.e. x_i^j, was transformed as follows:

$$_{(normalized)}x_i^j = \frac{x_i^j - x^{j,min}}{x^{j,max} - x^{j,min}} \tag{13}$$

where $x^{j,\ min}$ and $x^{j,\ max}$ correspond to the mean and standard deviation of the attribute $\mathcal{A}_j$, respectively.

The common One Hot Encoding scheme was employed to convert categorical features of Adult and Chess data sets to numerical features. This encoding allows algorithms which expect numerical features to use categorical features.

4.2. Simulation of missing values

To examine the effectiveness of the GFCMI method, we first artificially injected missing values into the pure data sets. Then, they were imputed by different imputation algorithms. Since the original values of the artificially created missing data are known, the performance of different methods can be evaluated precisely.

Both the amount and type of missing values affect the imputation precision (Junninen, Niska, Tuppurainen, Ruuskanen, & Kolehmainen, 2004). Therefore, to verify the accuracy of the proposed method in different situations, missing values were randomly inserted in 30%, 40%, 50%, and 60% of records, according to MAR and MNAR mechanisms as described by van Stein and Kowalczyk (2016). That is, for MAR pattern, values were removed uniformly at random, and for MNAR pattern, only values higher than the median value of the feature were removed. Accordingly, we have a total of eight missing combinations (2 missingness mechanisms $\times$ 4 missing ratios). Missing values for each data set were introduced into half of attributes. That is, half of the attributes were randomly chosen to place missing values in them. Therefore, each missing instance may have at most $m/2$ missing values.

4.3. Imputation approaches

The proposed method (GFCMI) was compared with five comparative imputation techniques namely mean imputation, kNNI, MLPI (Silva-Ramírez et al., 2015), FCMI (Raja & Thangavel, 2016), and IARI (van Stein & Kowalczyk, 2016). In the mean imputation, missing values of each attribute are replaced by the average of all values of that attribute. The kNNI method uses the closest objects of a missing record to impute that missing record. MLPI firstly trains a Multilayer Perceptron network using the complete data. Next, it uses the trained network to impute each missing record. The FCMI approach performs the fuzzy c-means algorithm on the input data. Missing values are then imputed according to information about the values of cluster centroids and membership degrees. The IARI method applies a sequence of regression models that iteratively replace all missing values. It incrementally repairs attributes with missing values, one by one, in the order of their importance.

4.4. The evaluation criteria

The imputation efficiency of GFCMI was evaluated using three well-known performance indicators: Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Coefficient of Determination (CoD) or R^2.

Root Mean Squared Error (RMSE) is the most commonly used performance measurement for quantitative values. It measures the average difference between the actual values and the estimated values by an imputation model (Rahman & Islam, 2016). The Coefficient of Determination (CoD) or R^2 expresses the association between the real values and the imputed values. It takes a value between 0 and 1. MAE measures the average magnitude of the errors in a set of predictions, without considering their direction (Deb &

Liew, 2016). The higher CoD, the lower RMSE and MAE show better imputation result. Eqs. (14)–(16) show how RMSE, CoD, and MAE are calculated, respectively.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2} \tag{14}$$

$$\text{CoD} = 1 - \frac{\sum_{i=1}^{n}(\tilde{y}_i - y_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{15}$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \tilde{y}_i| \tag{16}$$

where y_i is the original value, $\tilde{y}_i$ is the predicted value by the imputation model, $\bar{y}$ is the mean of observed data, and n is the total number of predictions.

5. Experimental results and analysis

This section presents extensive experimental results on validating the performance of the proposed algorithm. The investigations include (a) experimentally investigating the convergence of the GFCM algorithm (b) comparing the performance of GFCMI method with other imputation methods in terms of RMSE, MAE, and CoD (c) comparing the influence of different imputation methods on classification task accuracy (d) inspecting the effect of input parameters c and θ on the imputation performance of GFCMI. Next subsections provide the results of conducted experiments.

5.1. The convergence of GFCM algorithm

In this section, we conduct experiments to empirically verify the validity and convergence of the grey based fuzzy c-means clustering algorithm. To do this, values of the objective function (**J**) were calculated and recorded for the different number of clusters and various number of iterations of the GFCM algorithm.

Fig. 1 depicts values of **J** for cluster numbers $2 \le c \le 10$ in different data sets. The value of **J** is recorded after the completion of iterations 0 to 90. As the figures show, for all data sets, after iterating a few number of GFCM algorithm (almost ten iterations), the value of **J** had been greatly reduced by almost a monotonic downward trend. These changes have occurred for all values of c. After this downward tendency, for all data sets other than Haberman, the value of the objective function almost remained unchanged with some minor variations. Although the value of the objective function has some unstable fluctuations after the initial decline in the Haberman data set, the range of these fluctuations was relatively small. For the Wholesale Customers data set, after the aforementioned changes, although the value of the objective function has slightly increased for low values of c, it has converged to a constant value.

To summarize, changes for the objective function value empirically demonstrate that GFCM algorithm can minimize the objective function very well, even after a few number of iterations. This trend of changes meets our expectation which is the convergence of GFCM algorithm and its ability to minimize the objective function.

5.2. Comparisons with other imputation methods

In order to assess the validity of GFCMI algorithm, the performance of GFCMI was compared with five existing imputation methods, including mean imputation, k Nearest Neighbors imputation (kNNI), Multilayer Perceptron imputation (MLPI) (Silva-Ramírez et al., 2015), Fuzzy c-Mean imputation (FCMI) (Raja & Thangavel, 2016), and a recently regression-based imputation called Incremental Attribute Regression Imputation (IARI) (van Stein & Kowalczyk, 2016). By imputing the missing values at different mechanisms and ratios, the performance of these methods was recorded in terms of RMSE, MAE, and CoD. Tables 2–22 show RMSE, MAE, and CoD values for different data sets under different mechanisms with various missing rates. Bold values mark the best imputation result in comparison with other imputation methods for each case. Lower values in the parentheses indicate standard deviation. To determine whether the differences obtained among the GFCMI method and the other methods were significant or not, standard t-tests were applied on the results. "T" columns in tables indicate significant tests of the columns before them against the GFCMI method. Differences that were significant at 90% (p-value < 0.1), 95% (p-value < 0.05), and 99% (p-value < 0.01) confidence levels are marked with "*", "**", and "***", respectively. For example, when the missingness mechanism is MAR and the missing rate is 40% in the Iris data set, GFCMI performs significantly better than IARI at a confidence level of 95% in terms of RMSE. When no symbols are placed on a method it implies that there is no statistically significant difference with respect to GFCMI.

The results in Tables 2–4 illustrate the performance comparison between the proposed method and other methods for Iris data set. It can be seen that our GFCMI imputation method performs better than other methods. For this data set, mean imputation obtained the worst result. This is because there are high correlations between the features in this data set, while the mean imputation entirely ignores these correlations. The results also show significant differences between RMSE, MAE, and CoD values obtained by the grey based fuzzy c-means imputation (GFCMI) and the basic fuzzy c-means imputation (FCMI). For example, when the missing rate is 60% under MAR and MNAR mechanisms, the differences are 0.054 and 0.034 for the RMSE metric, respectively. This shows the superiority of the grey based fuzzy c-means method over the basic fuzzy c-means in the field of data imputation. What is more, the RMSE for the proposed method and the IARI method is more stable in comparison to the other methods. That is, the RMSE value for GFCMI and IARI methods increases slighter than the other methods while the missing rate increases. Considering MAE measure, kNNI performs better than IARI. The SVR type of GFCMI performs better than the MLR type for this data set in all of the cases except two cases of CoD.

Tables 5–7 compare the performance of GFCMI with that of kNNI, FCMI, MLPI and IARI, and the mean imputation methods on Wholesale Customers data set. Regardless of the missingness mechanism and missing ratio, it can be seen that GFCMI offers lower RMSE and MAE values and higher CoD values than the other imputation methods. IARI is a second best at handling missing data in this data set, followed by kNNI, FCMI, MLPI, and mean imputation in terms of RMSE and CoD. For MAE measure, however, MLPI operates worse than the mean imputation. For this data set, the SVR model presented better results than the MLR model.

Tables 8–10 presents obtained results for Wine data set in different missingness mechanisms with various missing rates. In terms of RMSE and CoD, it is clear that MLR type of GFCMI performs better than the other approaches for all the combinations of missing patterns and missing ratios. Even though SVR type of GFCMI provides a weaker performance than MLR type, it operates better than the best method among the other compared methods, i.e., IARI. Conversely, the SVR type is the best one for MAR pattern, considering the mean absolute error evaluation criterion. For this data set, the mean imputation obtained the worst result again.

The obtained results for the Glass data set are provided inTables 11–13. Based on both evaluation criteria of CoD and RMSE, the IARI method outperforms all the others, for missing rates varying from 30% to 50% at MAR pattern. For 60% of missing-

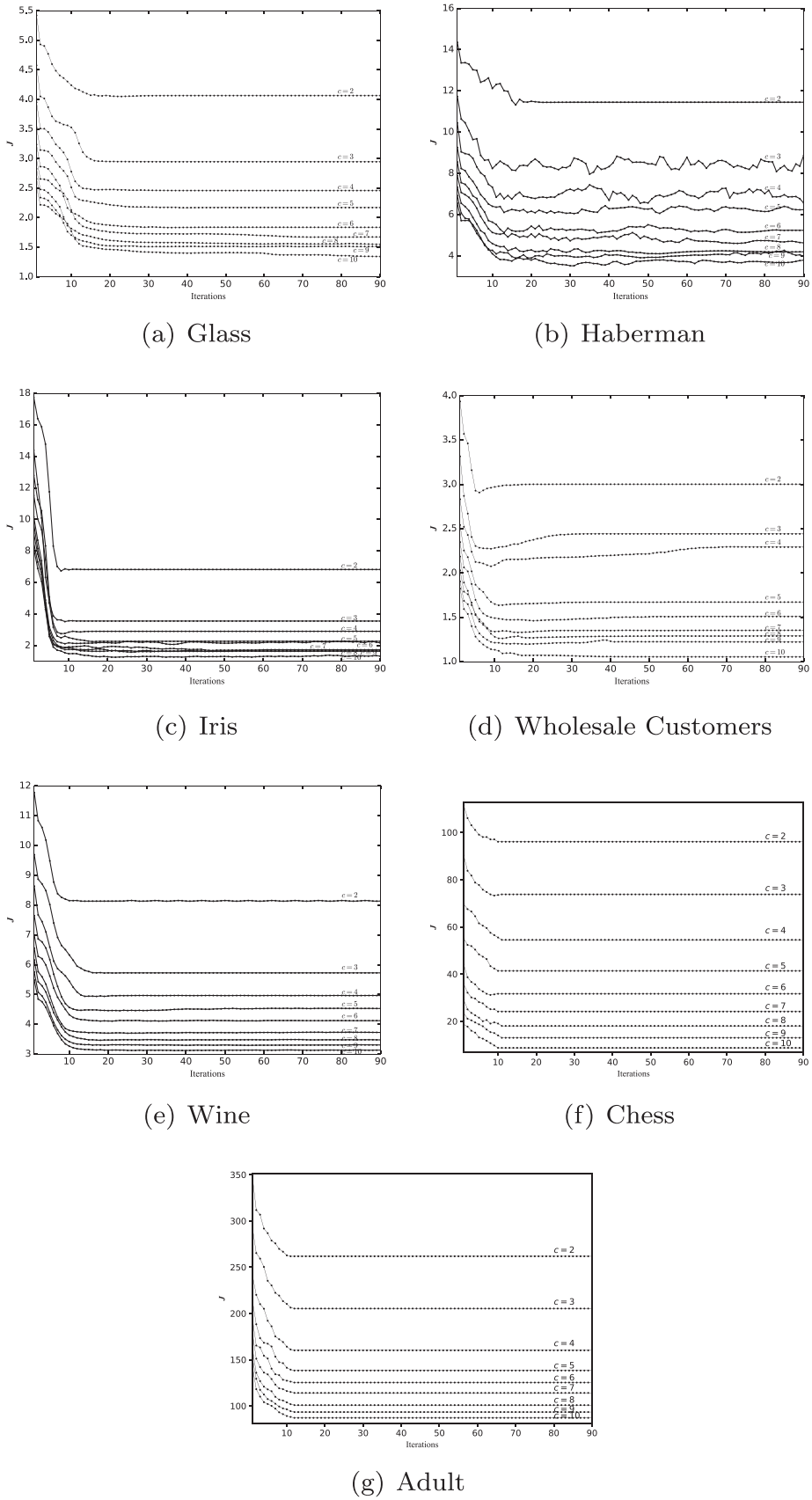

Fig. 1. Objective function values and the iteration count for (a) Glass, (b) Haberman, (c) Iris, (d) Wholesale Customers, (e) Wine, (f) Chess, and (g) Adult data set.

Table 2
RMSE for Iris data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.091236	***	0.034863	***	0.053599	***	0.064343	***	0.029823	***	0.026096	**0.024713**
		(0.004956)		(0.005260)		(0.007059)		(0.005978)		(0.004768)		(0.003327)	(0.003163)
	40%	0.104716	***	0.038434	***	0.059999	***	0.073202	***	0.033643	**	0.031683	**0.030501**
		(0.004706)		(0.005042)		(0.006760)		(0.004873)		(0.004432)		(0.004054)	(0.004095)
	50%	0.117367	***	0.044635	***	0.067326	***	0.083416	***	0.039206	***	0.037280	**0.036023**
		(0.004469)		(0.003342)		(0.005320)		(0.004123)		(0.004029)		(0.003671)	(0.003200)
	60%	0.128874	***	0.049241	***	0.073726	***	0.093255	***	0.042593	**	0.040197	**0.038876**
		(0.004690)		(0.004170)		(0.005137)		(0.004755)		(0.004574)		(0.004398)	(0.004161)
MNAR	**30%**	0.064010	***	0.031001	***	0.036070	***	0.040032	***	0.024963	**	0.022802	**0.021282**
		(0.006168)		(0.004717)		(0.005299)		(0.005239)		(0.004949)		(0.004402)	(0.004178)
	40%	0.076211	***	0.034733	***	0.044099	***	0.048088	***	0.028284		0.027935	**0.026552**
		(0.007177)		(0.005684)		(0.005879)		(0.006456)		(0.005158)		(0.005325)	(0.005413)
	50%	0.088241	***	0.040029	***	0.048724	***	0.056550	***	0.031888	*	0.030234	**0.028842**
		(0.005813)		(0.005682)		(0.004751)		(0.005456)		(0.005196)		(0.004505)	(0.004847)
	60%	0.100462	***	0.044546	***	0.056387	***	0.065918	***	0.035572	**	0.033347	**0.031822**
		(0.006920)		(0.005403)		(0.007232)		(0.006723)		(0.004561)		(0.004373)	(0.004526)

Table 3
MAE for Iris data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.023348	***	0.006184	***	0.012332	***	0.015609	***	0.006201	***	0.005565	**0.005399**
		(0.001562)		(0.000644)		(0.001516)		(0.001562)		(0.000812)		(0.000545)	(0.000605)
	40%	0.030674	***	0.008778	***	0.015974	***	0.020358	***	0.008134	**	0.007709	**0.007540**
		(0.001379)		(0.000817)		(0.001571)		(0.001467)		(0.000943)		(0.000801)	(0.000804)
	50%	0.038663	***	0.010194	***	0.019834	***	0.026163	***	0.010600	***	0.009801	**0.009220**
		(0.001520)		(0.000972)		(0.001683)		(0.001481)		(0.001102)		(0.000953)	(0.000948)
	60%	0.046295	***	0.012513	***	0.023732	***	0.032100	***	0.012683	***	0.011803	**0.010578**
		(0.001198)		(0.001061)		(0.001911)		(0.001403)		(0.001651)		(0.001667)	(0.001821)
MNAR	**30%**	0.011857	***	0.003366	***	0.005802	***	0.006628	***	0.003939	***	0.002617	**0.002330**
		(0.001725)		(0.000888)		(0.001117)		(0.001156)		(0.000851)		(0.000722)	(0.000711)
	40%	0.016359	***	0.004511		0.008025	***	0.009247	***	0.005148		0.004987	**0.004793**
		(0.002186)		(0.000991)		(0.001211)		(0.001624)		(0.000988)		(0.001013)	(0.001006)
	50%	0.021409	***	0.006082	**	0.010010	***	0.012387	***	0.006591	***	0.005680	**0.005438**
		(0.002103)		(0.000992)		(0.001101)		(0.001623)		(0.001027)		(0.001012)	(0.001014)
	60%	0.026714	***	0.007621	***	0.012734	***	0.015983	***	0.007984	***	0.006123	**0.005981**
		(0.002762)		(0.001168)		(0.001683)		(0.002277)		(0.001193)		(0.001081)	(0.001144)

Table 4
CoD for Iris data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.878237 (0.013265)	***	0.980787 (0.002289)	***	0.957512 (0.010654)	***	0.939097 (0.011081)	***	0.986698 (0.004271)	***	0.989888 (0.002282)	**0.990196** (0.002155)
	40%	0.839869 (0.011364)	***	0.978030 (0.004503)	***	0.947157 (0.011022)	***	0.921499 (0.010421)	***	0.983200 (0.004523)		0.985101 (0.003890)	**0.985251** (0.003961)
	50%	0.798915 (0.011892)	***	0.952349 (0.004433)	***	0.933204 (0.010336)	***	0.898267 (0.010112)	***	0.977338 (0.004646)		0.977304 (0.004425)	**0.979673** (0.004584)
	60%	0.757660 (0.010116)	***	0.947398 (0.005423)	***	0.920310 (0.010934)	***	0.872956 (0.010276)	***	0.963208 (0.005645)	***	0.975822 (0.005497)	**0.976248** (0.005555)
MNAR	**30%**	0.939686 (0.011391)	***	0.983393 (0.003198)	***	0.980554 (0.005546)	***	0.976227 (0.006215)	***	0.990557 (0.003858)	*	0.992126 (0.002904)	**0.992501** (0.002875)
	40%	0.914538 (0.016175)	***	0.970852 (0.003805)	***	0.970793 (0.007065)	***	0.965664 (0.009267)	***	0.978090 (0.003629)	***	0.988321 (0.003500)	**0.989425** (0.003522)
	50%	0.885941 (0.015096)	***	0.976588 (0.006135)	***	0.964765 (0.006914)	***	0.952924 (0.009127)	***	0.974923 (0.006871)	***	**0.983346** (0.005488)	0.982765 (0.006530)
	60%	0.852102 (0.019946)	***	0.962432 (0.006403)	***	0.953244 (0.012195)	***	0.935966 (0.012814)	***	0.971301 (0.006257)		**0.974559** (0.006209)	0.973983 (0.006391)

Table 5
RMSE for Wholesale Customers data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.020691 (0.005713)	***	0.015604 (0.004887)	***	0.020098 (0.005969)	***	0.018431 (0.006027)	***	0.013842 (0.004815)	*	0.012713 (0.004502)	**0.011096** (0.004585)
	40%	0.024958 (0.005746)	***	0.017603 (0.004654)	***	0.025692 (0.005510)	***	0.022408 (0.005956)	***	0.016467 (0.004159)	***	0.015683 (0.004134)	**0.012801** (0.003917)
	50%	0.027446 (0.005061)	***	0.020071 (0.004413)	***	0.029130 (0.004946)	***	0.024785 (0.005277)	***	0.019145 (0.004911)	**	0.018280 (0.004350)	**0.016023** (0.004351)
	60%	0.030793 (0.004473)	***	0.022856 (0.004936)	***	0.033716 (0.004852)	***	0.028014 (0.004655)	***	0.021546 (0.004357)	**	0.020597 (0.004306)	**0.018176** (0.004385)
MNAR	**30%**	0.019220 (0.006270)	***	0.014128 (0.005311)	**	0.017988 (0.004923)	***	0.017485 (0.006476)	***	0.013292 (0.004928)	*	0.012448 (0.004993)	**0.010257** (0.004865)
	40%	0.023393 (0.006344)	***	0.016077 (0.004944)	**	0.023033 (0.005314)	***	0.021440 (0.006409)	***	0.016489 (0.004928)	***	0.015043 (0.004462)	**0.012432** (0.004115)
	50%	0.025853 (0.005536)	***	0.018187 (0.004677)	**	0.025243 (0.004640)	***	0.023812 (0.005627)	***	0.018768 (0.004723)	***	0.016803 (0.004639)	**0.014294** (0.004465)
	60%	0.029269 (0.004875)	***	0.020866 (0.004604)	**	0.029783 (0.005039)	***	0.027090 (0.004931)	***	0.020284 (0.004671)	**	0.019731 (0.004558)	**0.017147** (0.004421)

Table 6
MAE for Wholesale Customers data set.

Missingness mechanism (MLR) (SVR)	Missing rate GFCMI	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI	
MAR	**30%**	0.002872 (0.000337)	***	0.001965 (0.000275)	***	0.003246 (0.000510)	***	0.002400 (0.000323)	***	0.001790 (0.000282)	**	0.001777 (0.000270)	**0.001566** (0.000259)
	40%	0.004001 (0.000377)	***	0.002639 (0.000392)	***	0.004585 (0.000509)	***	0.003364 (0.000360)	***	0.002487 (0.000314)	***	0.002403 (0.000306)	**0.002169** (0.000252)
	50%	0.004881 (0.000360)	***	0.003294 (0.000357)	***	0.005720 (0.000457)	***	0.004130 (0.000353)	***	0.003137 (0.000349)	***	0.003094 (0.000328)	**0.002733** (0.000304)
	60%	0.005942 (0.000400)	***	0.004028 (0.000453)	***	0.007242 (0.000444)	***	0.005070 (0.000388)	***	0.003778 (0.000408)	***	0.003800 (0.000406)	**0.003411** (0.000368)
MNAR	**30%**	0.001658 (0.000374)	***	0.001363 (0.000299)	***	0.001863 (0.000299)	***	0.001477 (0.000354)	***	0.001234 (0.000298)	**	0.001140 (0.000283)	**0.001009** (0.000276)
	40%	0.002318 (0.000428)	***	0.001827 (0.000304)	***	0.002689 (0.000390)	***	0.002071 (0.000400)	***	0.001689 (0.000292)	**	0.001568 (0.000293)	**0.001464** (0.000261)
	50%	0.002830 (0.000423)	***	0.002293 (0.000300)	***	0.003359 (0.000453)	***	0.002534 (0.000398)	***	0.002049 (0.000341)		0.001984 (0.000311)	**0.001937** (0.000302)
	60%	0.003509 (0.000516)	***	0.002836 (0.000392)	***	0.004123 (0.000519)	***	0.003150 (0.000475)	***	0.002545 (0.000383)		0.002507 (0.000405)	**0.002440** (0.000382)

Table 7
CoD for Wholesale Customers data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.951196 (0.025953)	***	0.972532 (0.016547)		0.955519 (0.028955)	***	0.960172 (0.024570)	***	0.978164 (0.015582)		0.978153 (0.014731)	**0.980073** (0.014998)
	40%	0.930522 (0.030219)	***	0.966029 (0.017727)		0.927108 (0.029127)	***	0.943055 (0.028190)	***	0.970223 (0.014723)		0.970019 (0.015213)	**0.973992** (0.014292)
	50%	0.917500 (0.030043)	***	0.955881 (0.020100)	*	0.907306 (0.028984)	***	0.931986 (0.028540)	***	0.958613 (0.024567)		0.963018 (0.019422)	**0.967131** (0.019315)
	60%	0.897446 (0.030648)	***	0.943630 (0.021388)	*	0.877256 (0.027022)	***	0.914579 (0.029255)	***	0.948900 (0.022435)		0.951047 (0.021478)	**0.955537** (0.021356)
MNAR	**30%**	0.956709 (0.026323)	***	0.977130 (0.016721)		0.961735 (0.018152)	***	0.963177 (0.024936)	***	0.979206 (0.013447)		0.980947 (0.014948)	**0.981580** (0.014190)
	40%	0.937773 (0.030919)	***	0.971566 (0.017657)		0.938830 (0.023161)	***	0.946962 (0.028774)	***	0.972955 (0.014197)		0.973921 (0.015213)	**0.976146** (0.014427)
	50%	0.925956 (0.030853)	***	0.963732 (0.020239)		0.929429 (0.022893)	***	0.936587 (0.029178)	***	0.966752 (0.019536)		0.968813 (0.019471)	**0.969767** (0.019133)
	60%	0.906743 (0.031971)	***	0.952897 (0.021639)		0.902944 (0.031040)	***	0.919690 (0.030155)	***	0.954183 (0.020089)		0.956563 (0.021561)	**0.959099** (0.020661)

Table 8
RMSE for Wine data set.

Missingness mechanism (MLR) (SVR)	Missing rate GFCMI	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI	
MAR	**30%**	0.036360 (0.002430)	***	0.023976 (0.002743)	***	0.033730 (0.002940)	***	0.030167 (0.002794)	***	0.022538 (0.002893)	***	**0.020096** (0.002564)	0.021713 (0.002638)
	40%	0.041661 (0.002593)	***	0.028334 (0.002604)	***	0.039631 (0.003268)	***	0.034952 (0.002735)	***	0.026348 (0.002550)	**	**0.024683** (0.002435)	0.025501 (0.002503)
	50%	0.046409 (0.002888)	***	0.031679 (0.002822)	***	0.042908 (0.002983)	***	0.039003 (0.003081)	***	0.029118 (0.002948)		**0.028280** (0.002743)	0.028823 (0.002964)
	60%	0.050661 (0.002894)	***	0.035406 (0.002915)	***	0.046779 (0.003326)	***	0.042911 (0.003053)	***	0.032450 (0.003002)	**	**0.030197** (0.002861)	0.031176 (0.002963)
MNAR	**30%**	0.027071 (0.004420)	***	0.017629 (0.002871)	**	0.024798 (0.003325)	***	0.022375 (0.004001)	***	0.017585 (0.003466)	**	**0.015482** (0.002908)	0.016583 (0.003100)
	40%	0.031450 (0.004694)	***	0.021199 (0.003409)		0.028471 (0.004272)	***	0.026277 (0.004313)	***	0.020695 (0.003651)		**0.019523** (0.003112)	0.019750 (0.003315)
	50%	0.034953 (0.004515)	***	0.023116 (0.003399)	**	0.032038 (0.004250)	***	0.029126 (0.004231)	***	0.022611 (0.003450)	**	**0.020444** (0.003268)	0.021770 (0.003383)
	60%	0.038725 (0.003840)	***	0.026067 (0.003244)	**	0.035859 (0.003803)	***	0.032557 (0.003633)	***	0.025427 (0.002950)		**0.023882** (0.003011)	0.024185 (0.003326)

Table 9
MAE for Wine data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.004955 (0.000308)	***	0.003013 (0.000276)	***	0.004448 (0.000426)	***	0.004028 (0.000294)	***	0.002849 (0.000330)	**	0.002760 (0.000267)	**0.002621** (0.000280)
	40%	0.006549 (0.000416)	***	0.004120 (0.000303)	***	0.006009 (0.000522)	***	0.005356 (0.000435)	***	0.003862 (0.000335)		0.003719 (0.000315)	**0.003695** (0.000324)
	50%	0.008121 (0.000426)	***	0.005093 (0.000467)	***	0.007198 (0.000511)	***	0.006666 (0.000433)	***	0.004717 (0.000430)		0.004622 (0.000428)	**0.004542** (0.000438)
	60%	0.009680 (0.000455)	***	0.006194 (0.000453)	***	0.008570 (0.000687)	***	0.007991 (0.000458)	***	0.005733 (0.000486)		0.005598 (0.000451)	**0.005512** (0.000476)
MNAR	**30%**	0.002598 (0.000591)		0.001640 (0.000338)	***	0.002308 (0.000411)	***	0.002066 (0.000499)	***	0.001630 (0.000394)		**0.001511** (0.000322)	0.001520 (0.000324)
	40%	0.003487 (0.000680)	**	0.002289 (0.000391)	***	0.003077 (0.000476)	***	0.002790 (0.000578)	***	0.002200 (0.000416)		**0.002040** (0.000371)	0.002083 (0.000373)
	50%	0.004299 (0.000735)	*	0.002754 (0.000446)	***	0.003793 (0.000637)	***	0.003433 (0.000640)	***	0.002649 (0.000433)		**0.002521** (0.000426)	0.002564 (0.000424)
	60%	0.005234 (0.000705)	**	0.003388 (0.000439)	***	0.004669 (0.000590)	***	0.004208 (0.000612)	***	0.003244 (0.000429)		**0.003053** (0.000426)	0.003089 (0.000437)

Table 10
CoD for Wine data set.

Missingness mechanism (MLR) (SVR)	Missing rate GFCMI	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI	
MAR	**30%**	0.967889 (0.003731)	***	0.985899 (0.003061)	**	0.972275 (0.004417)	***	0.977855 (0.003172)	***	0.986501 (0.003182)	*	**0.988349** (0.002663)	0.987718 (0.002858)
	40%	0.957826 (0.005147)	***	0.980367 (0.003425)	***	0.961761 (0.005816)	***	0.970248 (0.004593)	***	0.981043 (0.003268)	***	**0.984233** (0.003195)	0.982946 (0.003295)
	50%	0.947717 (0.005528)	***	0.975561 (0.004899)	***	0.955174 (0.005896)	***	0.963016 (0.004866)	***	0.979283 (0.004522)		**0.980570** (0.004553)	0.980725 (0.004561)
	60%	0.937771 (0.005099)	***	0.969484 (0.004562)	***	0.946659 (0.007276)	***	0.955308 (0.004653)	***	0.974294 (0.004619)		**0.975870** (0.004485)	0.975990 (0.004617)
MNAR	**30%**	0.981788 (0.005670)	***	0.982250 (0.002637)	***	0.984621 (0.003578)	***	0.987495 (0.004340)	***	0.992222 (0.003144)	*	**0.993935** (0.002320)	0.992609 (0.002516)
	40%	0.975526 (0.006977)	***	0.985901 (0.003513)	***	0.979934 (0.005778)	***	0.982837 (0.005443)	***	0.989315 (0.003850)		**0.990558** (0.003186)	0.989723 (0.003328)
	50%	0.969935 (0.007284)	***	0.986811 (0.003715)	*	0.974751 (0.006188)	***	0.979033 (0.005760)	***	0.987337 (0.003913)		**0.988891** (0.003470)	0.987423 (0.003677)
	60%	0.963344 (0.007039)	***	0.984329 (0.003701)		0.968418 (0.006186)	***	0.974024 (0.005615)	***	0.984140 (0.003709)		**0.985194** (0.003559)	0.984689 (0.003888)

Table 11
RMSE for Glass data set.

Missingness mechanism (MLR) (SVR)	Missing rate GFCMI	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI	
MAR	**30%**	0.031926 (0.005194)	***	0.028065 (0.005298)	*	0.036946 (0.005194)	***	0.030478 (0.005273)	***	**0.024559** (0.004778)		0.025402 (0.004665)	0.026711 (0.004784)
	40%	0.037667 (0.004675)	***	0.033988 (0.004795)	**	0.043886 (0.005490)	***	0.035865 (0.004734)	***	**0.029146** (0.004426)	*	0.030077 (0.004423)	0.031547 (0.004526)
	50%	0.041271 (0.004862)	***	0.038739 (0.004875)	***	0.047221 (0.004722)	***	0.039348 (0.004916)	***	**0.033161** (0.004618)		0.033211 (0.004737)	0.034956 (0.004779)
	60%	0.044692 (0.004720)	***	0.043048 (0.004644)	***	0.052514 (0.004799)	***	0.042803 (0.004758)	***	0.036080 (0.004257)		**0.036026** (0.004398)	0.038199 (0.004330)
MNAR	**30%**	0.028558 (0.005893)	***	0.022917 (0.006046)		0.029990 (0.006063)	***	0.026987 (0.005974)	***	**0.019868** (0.005511)		0.020229 (0.005442)	0.022716 (0.005747)
	40%	0.034110 (0.005774)	***	0.027318 (0.005710)	***	0.036256 (0.006469)	***	0.032194 (0.005430)	***	0.023563 (0.005455)		**0.022447** (0.005387)	0.024734 (0.005530)
	50%	0.036956 (0.006310)	***	0.030252 (0.006664)		0.039985 (0.006095)	***	0.034798 (0.006181)	***	**0.026979** (0.005945)		0.027873 (0.005962)	0.029618 (0.006038)
	60%	0.039692 (0.005785)	***	0.033272 (0.005901)	**	0.042746 (0.005769)	***	0.037518 (0.005845)	***	0.029512 (0.005568)		**0.029430** (0.005470)	0.031680 (0.005561)

Table 12
MAE for Glass data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.004039 (0.000441)	***	0.003002 (0.000431)	**	0.004790 (0.000465)	***	0.003713 (0.000462)	***	**0.002660** (0.000451)		0.002847 (0.000351)	0.002682 (0.000370)
	40%	0.005585 (0.000459)	***	0.004196 (0.000431)	***	0.006621 (0.000624)	***	0.005108 (0.000460)	***	0.003728 (0.000427)		0.004015 (0.000356)	**0.003706** (0.000379)
	50%	0.006855 (0.000535)	***	0.052267 (0.000553)	***	0.008014 (0.000589)	***	0.006270 (0.000532)	***	0.004578 (0.000540)	**	0.004902 (0.000435)	**0.004222** (0.000489)
	60%	0.008282 (0.000578)	***	0.006525 (0.000585)	*	0.009776 (0.000757)	***	0.007636 (0.000570)	***	0.005591 (0.000547)	**	0.005526 (0.000493)	**0.005250** (0.000508)
MNAR	**30%**	0.002219 (0.000516)	***	0.001597 (0.000455)		0.002541 (0.000513)	***	0.002083 (0.000515)	***	**0.001424** (0.000435)		0.001546 (0.000403)	0.001875 (0.000422)
	40%	0.003097 (0.000546)	***	0.002245 (0.000490)	**	0.003464 (0.000690)	***	0.002896 (0.000536)	***	0.001958 (0.000491)		**0.001894** (0.000461)	0.002027 (0.000489)
	50%	0.003759 (0.000678)	***	0.002729 (0.000617)		0.004303 (0.000749)	***	0.003494 (0.000671)	***	**0.002465** (0.000583)		0.002655 (0.000571)	0.002865 (0.000602)
	60%	0.004472 (0.000710)	***	0.003310 (0.000625)		0.005099 (0.000845)	***	0.004170 (0.000705)	***	**0.002962** (0.000591)		0.003108 (0.000573)	0.003394 (0.000602)

Table 13
CoD for Glass data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.964872 (0.011322)	***	0.972769 (0.010091)		0.953004 (0.013224)	***	0.967879 (0.010907)	***	**0.979049** (0.009871)		0.977604 (0.008228)	0.975219 (0.008700)
	40%	0.951631 (0.011633)	***	0.962560 (0.010394)	**	0.934284 (0.015621)	***	0.956060 (0.011113)	***	**0.970817** (0.008945)		0.968971 (0.008645)	0.965820 (0.009066)
	50%	0.942019 (0.013419)	***	0.953744 (0.012009)	**	0.924029 (0.014978)	***	0.947205 (0.012979)	***	**0.962392** (0.011072)		0.962215 (0.009218)	0.958129 (0.010323)
	60%	0.932190 (0.014285)	***	0.944492 (0.012833)	***	0.906177 (0.016114)	***	0.937727 (0.013863)	***	0.955709 (0.010216)		**0.956289** (0.008648)	0.950324 (0.009786)
MNAR	**30%**	0.971451 (0.011494)	***	0.981551 (0.010003)		0.968813 (0.011257)	***	0.974349 (0.011083)	**	**0.985830** (0.009149)		0.982423 (0.008344)	0.976794 (0.009035)
	40%	0.960001 (0.011868)	***	0.974105 (0.010699)	**	0.954727 (0.015556)	***	0.964210 (0.011431)	***	0.980393 (0.009472)		**0.982547** (0.009391)	0.967262 (0.010583)
	50%	0.952972 (0.014334)	***	0.968059 (0.013434)	*	0.944778 (0.015184)	***	0.958061 (0.014072)	***	**0.974369** (0.009212)		0.970737 (0.010225)	0.960766 (0.011802)
	60%	0.946020 (0.015209)	***	0.961869 (0.013385)	*	0.937446 (0.017154)	***	0.951592 (0.014870)	***	0.969790 (0.011465)		**0.967830** (0.011303)	0.953775 (0.012852)

ness, GFCMI (MLR) presents the best result, followed by the IARI method. For 40% and 60% of missing values at MNAR type, the best performance is achieved by GFCMI (MLR), again. However, for 30% and 50% of missing values, IARI provides the optimal RMSE and CoD value. Considering the MAE criterion, when the missingness mechanism is MAR and the missing rate in the incomplete data set is larger than 30%, GFCMI (SVR) performs the best. For MAR pattern at 30% and MNAR pattern at 30%, 50%, and 60%, the IARI method presents the least MAE value.

Tables 14–16 compare the performance of different imputation methods for Haberman data set. For this data set, although IARI, kNNI, MLPI, and FCMI perform worse than the mean substitution method, the results show that MLR type of GFCMI method has a slight superiority over the mean imputation. The relative weakness of the other methods compared with the mean imputation is due to the small number of predictor variables in this data set. Moreover, the small number of instances for imputation in this data set results in less biased estimations. Thus, in spite of being a simplistic method, the mean imputation could be a better choice to perform imputation when there is a small number of features or instances, because of its less computational cost.

The results for the Chess data set are presented inTables 17–19. It is clear that the GFCMI (MLR) method provides better results than other methods in terms of RMSE, regardless of the missingness mechanism and missing ratio. This is also the case for MAE and CoD measures when the missingness mechanism is MAR. For 30% and 40% of missing values at MNAR type, GFCMI (MLR) outperforms Mean, FCM, and MLP imputation methods in terms of MAE. However, when the missingness pattern is MNAR, GFCMI (MLR) perform worse than IARI and kNNI at missing rates 30% and 40%, respectively. At 50% and 60% missing rates GFCMI is the best imputation method followed by kNNI at managing missing data. Considering the CoD evaluation criterion, the GFCMI (MLR) method outperforms all the others when the missingness pattern is MAR. However, at MNAR missingness pattern, kNNI and IARI outperform the GFCMI algorithm.

The obtained results for Adult data set (Tables 20–22) offer that the MLR type of GFCMI provides lower RMSE and MAE values than other imputation method. In terms of CoD, it can be seen that, except for three cases that IARI achieves higher CoD values than GFCMI (50% and 60% at MAR pattern and 60% at MNAR pattern), GFCMI operates better than other imputation methods.

In summary, the experimental results illustrate some phenomena that could be discussed as follows:

(a) Increasing proportion of missing values degrades the imputation performance considering all of the imputation algorithms. In other words, more available information could improve the precision of final approximations. This is rational because we lose more useful knowledge as the number of missing values increases.

(b) Considering different missing ratios and different missingness mechanisms, the IARI method outperforms kNNI, FMCI, MLPI, and mean imputation, in most cases. Similarly, for all data sets except the Glass data set and some cases in Chess data set, GFCMI provides lower imputation errors in comparison with IARI. Therefore, it can be said that GFCMI is superior to the other methods, and IARI is a second best at missing value imputation for the tested data sets. It is worth noticing that the IARI method uses an extra predictor variable, i.e., class label of records, as well as other predictor variables by default.

(c) Statistical analysis of the results (t-tests) reveals significant differences in the means of RMSE values, which is the most reliable performance indicator in the missing value imputation, between the GFCMI algorithm and the Mean imputation, kNNI, FCMI, and MLPI algorithms, in most cases. It can also be observed that there are significant differences in the means of RMSE values between

Table 14
RMSE for Haberman data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.073809	**	0.090718	***	0.076416	***	0.074974	***	0.082576	***	**0.070841**	0.073411
		(0.004513)		(0.005145)		(0.004612)		(0.004646)		(0.006619)		(0.004500)	(0.004589)
	40%	0.088862		0.110032	***	0.093408	***	0.090205	**	0.098874	***	**0.086889**	0.088400
		(0.00407)		(0.0041490)		(0.004412)		(0.004085)		(0.004549)		(0.003933)	(0.004084)
	50%	0.100338		0.123904	***	0.105968	***	0.100741	*	0.113329	***	**0.098367**	0.099492
		(0.003796)		(0.003909)		(0.004777)		(0.003807)		(0.004352)		(0.003760)	(0.003803)
	60%	0.111476	***	0.135649	***	0.117827	***	0.113404	***	0.126812	***	**0.100697**	0.110591
		(0.004688)		(0.005015)		(0.004745)		(0.004792)		(0.004711)		(0.004629)	(0.004628)
MNAR	**30%**	0.058072		0.069678	***	0.061373	**	0.057250		0.064925	***	**0.056799**	0.059307
		(0.005973)		(0.006585)		(0.006618)		(0.006255)		(0.006561)		(0.005963)	(0.006096)
	40%	**0.070920**		0.084997	***	0.073530		0.071205		0.078437	***	0.070930	0.073661
		(0.005510)		(0.006261)		(0.006012)		(0.005845)		(0.006162)		(0.005354)	(0.005542)
	50%	0.081309		0.097023	***	0.085299	***	0.083630	*	0.090038	***	**0.079532**	0.084130
		(0.006279)		(0.007195)		(0.007073)		(0.007414)		(0.006359)		(0.006114)	(0.006293)
	60%	0.091819		0.107095	***	0.096246	**	0.095756	*	0.100241	***	**0.090704**	0.094387
		(0.007436)		(0.008106)		(0.008803)		(0.008552)		(0.007572)		(0.007434)	(0.007660)

Table 15
MAE for Haberman data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.018002 (0.001197)	***	0.021412 (0.001256)	***	0.019068 (0.001314)	***	0.017740 (0.001110)	***	0.019884 (0.001522)	***	**0.016006** (0.001097)	0.017193 (0.001108)
	40%	0.025404 (0.001065)	***	0.030644 (0.001093)	***	0.027170 (0.001114)	***	0.025140 (0.001132)	***	0.027844 (0.001523)	***	**0.023456** (0.001017)	0.024712 (0.001120)
	50%	0.032405 (0.001100)	***	0.038785 (0.001198)	***	0.034811 (0.001526)	***	0.032105 (0.001187)	***	0.035945 (0.001108)	***	**0.030497** (0.001176)	0.031888 (0.001203)
	60%	0.039709 (0.001596)	***	0.046719 (0.001639)	***	0.042956 (0.001652)	***	0.039483 (0.001622)	***	0.044271 (0.001511)	***	**0.036636** (0.001659)	0.039396 (0.001721)
MNAR	**30%**	0.009761 (0.001599)	***	0.011767 (0.001591)	***	0.010635 (0.001593)	***	0.009604 (0.001547)	***	0.010902 (0.001644)	***	**0.007485** (0.001477)	0.010330 (0.001545)
	40%	0.013927 (0.001574)	**	0.016605 (0.001616)	***	0.014685 (0.001512)	***	0.014023 (0.001622)	**	0.015273 (0.001611)	***	**0.012911** (0.001453)	0.014902 (0.001575)
	50%	0.018087 (0.002092)	**	0.021271 (0.002147)	***	0.019340 (0.002194)	***	0.018757 (0.002356)	***	0.019820 (0.001963)	***	**0.016739** (0.001983)	0.018278 (0.002076)
	60%	0.022555 (0.002623)	**	0.025961 (0.002743)	***	0.024145 (0.002955)	***	0.023753 (0.002951)	***	0.024214 (0.002645)	***	**0.020350** (0.002647)	0.022856 (0.002764)

Table 16
CoD for Haberman data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.881742 (0.012804)	**	0.831435 (0.015965)	***	0.880756 (0.014055)	**	0.881247 (0.012989)	**	0.860367 (0.022034)	***	**0.891634** (0.013101)	0.889919 (0.013441)
	40%	0.832639 (0.013323)	**	0.752718 (0.015076)	***	0.821772 (0.016588)	***	0.831378 (0.014233)	**	0.800664 (0.018074)	***	**0.842517** (0.013966)	0.840664 (0.014614)
	50%	0.778956 (0.014582)	***	0.686665 (0.014518)	***	0.771137 (0.020386)	***	0.777291 (0.015553)	***	0.738527 (0.016810)	***	**0.798813** (0.015262)	0.798269 (0.016249)
	60%	0.741297 (0.019395)		0.624871 (0.023731)	***	0.717118 (0.022889)	***	0.747035 (0.021335)		0.672536 (0.019157)	***	**0.750239** (0.019977)	0.750720 (0.019958)
MNAR	**30%**	0.920657 (0.013089)	***	0.899785 (0.015611)	***	0.922570 (0.015197)	**	0.922517 (0.013470)	**	0.913357 (0.016863)	***	**0.933180** (0.013042)	0.927425 (0.014918)
	40%	0.887069 (0.015193)	**	0.851818 (0.018515)	***	0.889420 (0.017560)		0.886146 (0.016388)	**	0.874036 (0.019363)	***	**0.897052** (0.015060)	0.888752 (0.016972)
	50%	0.854709 (0.019622)	*	0.807012 (0.023887)	***	0.851208 (0.023022)	**	0.846578 (0.024497)	***	0.834220 (0.021213)	***	**0.867120** (0.019408)	0.855017 (0.020052)
	60%	0.817338 (0.026712)		0.764807 (0.030154)	***	0.810130 (0.032657)	*	0.819490 (0.032328)		0.794457 (0.028202)	***	**0.827154** (0.026771)	0.817232 (0.027033)

Table 17
RMSE for Chess data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.026695 (0.002398)	***	0.016703 (0.001786)	**	0.025181 (0.001444)	***	0.023245 (0.001398)	***	0.017542 (0.001592)	***	**0.015524** (0.001392)	0.015964 (0.001517)
	40%	0.028907 (0.002479)	***	0.018774 (0.001822)	***	0.028356 (0.001669)	***	0.027473 (0.001479)	***	0.018816 (0.001660)	***	**0.016352** (0.001492)	0.017374 (0.001542)
	50%	0.032556 (0.003020)	***	0.019294 (0.001807)	***	0.029165 (0.001712)	***	0.031352 (0.001442)	***	0.020988 (0.001691)	***	**0.017528** (0.001548)	0.018626 (0.001578)
	60%	0.035410 (0.003120)	***	0.022535 (0.001965)		0.030680 (0.002092)	***	0.034687 (0.001620)	***	0.023494 (0.001777)	***	**0.021627** (0.001692)	0.022388 (0.001767)
MNAR	**30%**	0.026327 (0.002751)	***	0.017761 (0.001762)	**	0.026642 (0.001723)	***	0.025427 (0.001693)	***	0.018074 (0.001536)	***	**0.016455** (0.001556)	0.016774 (0.001537)
	40%	0.030822 (0.002942)	***	0.019205 (0.001856)	*	0.030197 (0.001800)	***	0.029384 (0.001646)	***	0.020127 (0.001640)	***	**0.018128** (0.001612)	0.018972 (0.001667)
	50%	0.035818 (0.002869)	***	0.022606 (0.001861)	**	0.035555 (0.001860)	***	0.034517 (0.001685)	***	0.023020 (0.001657)	***	**0.021270** (0.001624)	0.022248 (0.001645)
	60%	0.039620 (0.003035)	***	0.025821 (0.001952)	*	0.038278 (0.001971)	***	0.037284 (0.001751)	***	0.026599 (0.001763)	***	**0.024673** (0.001734)	0.023460 (0.001719)

Table 18
MAE for Chess data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.002515 (0.000323)	***	0.000869 (0.000219)	***	0.002447 (0.000188)	***	0.002104 (0.000220)	***	0.000923 (0.000124)	***	**0.000431** (0.000134)	0.000541 (0.000157)
	40%	0.002754 (0.000447)	***	0.001241 (0.000225)	***	0.002716 (0.000218)	***	0.002269 (0.000237)	***	0.001437 (0.000146)	***	**0.000637** (0.000127)	0.000794 (0.000169)
	50%	0.003210 (0.000479)	***	0.001637 (0.000324)	***	0.002984 (0.000229)	***	0.002837 (0.000251)	***	0.001758 (0.000188)	***	**0.000873** (0.000165)	0.000998 (0.000184)
	60%	0.003497 (0.000490)	***	0.002038 (0.000404)	***	0.003052 (0.000299)	***	0.003119 (0.000269)	***	0.001954 (0.000209)	***	**0.001282** (0.000177)	0.001329 (0.000208)
MNAR	**30%**	0.003092 (0.000226)	***	0.001739 (0.000196)		0.002928 (0.000153)	***	0.002732 (0.000246)	***	**0.001722** (0.000115)		0.001737 (0.000108)	0.001862 (0.000160)
	40%	0.003946 (0.000337)	***	**0.002302** (0.000254)		0.003993 (0.000247)	***	0.003857 (0.000263)	***	0.002364 (0.000138)		0.002331 (0.000109)	0.002463 (0.000193)
	50%	0.005263 (0.000404)	***	0.003056 (0.000284)		0.005120 (0.000322)	***	0.004898 (0.000229)	***	0.003079 (0.000216)		**0.003021** (0.000159)	0.003236 (0.000234)
	60%	0.006422 (0.000449)	***	0.003722 (0.000398)		0.006204 (0.000337)	***	0.006067 (0.000279)	***	0.003744 (0.000256)		**0.003678** (0.000205)	0.003820 (0.000211)

Table 19
CoD for Chess data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.985306 (0.012364)	***	0.998554 (0.004281)		0.989862 (0.005742)	***	0.985306 (0.005152)	***	0.998370 (0.003133)		**0.999301** (0.003547)	0.999235 (0.003352)
	40%	0.983612 (0.013454)	***	0.998035 (0.005803)		0.988776 (0.006656)	***	0.983612 (0.006212)	***	0.997819 (0.003200)		**0.999051** (0.003374)	0.999003 (0.004358)
	50%	0.981842 (0.013690)	***	0.997423 (0.004615)		0.986706 (0.007449)	***	0.981842 (0.004210)	***	0.997222 (0.004104)		**0.998512** (0.003785)	0.998347 (0.004473)
	60%	0.980349 (0.014778)	***	0.996754 (0.005115)		0.975655 (0.005351)	***	0.980349 (0.004229)	***	0.996764 (0.005317)		**0.998190** (0.004693)	0.998184 (0.004233)
MNAR	**30%**	0.971514 (0.012134)	***	**0.982497** (0.006667)		0.981069 (0.006325)		0.981514 (0.004973)		0.982223 (0.003789)		0.982213 (0.003845)	0.981511 (0.004082)
	40%	0.964669 (0.011167)	***	**0.980108** (0.005700)	***	0.974044 (0.006342)		0.974669 (0.005067)		0.979193 (0.003685)	***	0.974242 (0.003775)	0.969247 (0.003900)
	50%	0.957306 (0.012294)	***	0.973140 (0.005692)		0.966296 (0.006758)	***	0.967306 (0.005386)	***	**0.975868** (0.003681)		0.975252 (0.003949)	0.968799 (0.004274)
	60%	0.950234 (0.011687)	***	0.970819 (0.004798)		0.958870 (0.006412)	***	0.960234 (0.005549)	***	**0.972526** (0.004467)		0.971655 (0.004252)	0.965381 (0.004537)

Table 20
RMSE for Adult data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.022635 (0.003276)	***	0.020278 (0.002253)	***	0.019552 (0.002262)	***	0.019528 (0.002288)	***	0.018027 (0.002004)	**	**0.016643** (0.001905)	0.017642 (0.001978)
	40%	0.026890 (0.003567)	***	0.021321 (0.002184)	***	0.020569 (0.002227)	*	0.021774 (0.002271)	***	0.020765 (0.001980)	**	**0.019374** (0.001941)	0.020347 (0.001964)
	50%	0.028544 (0.003575)	***	0.023651 (0.002317)	***	0.023122 (0.002308)	***	0.024414 (0.002374)	***	0.022921 (0.002187)	***	**0.020832** (0.001967)	0.021639 (0.002032)
	60%	0.031354 (0.003762)	***	0.026247 (0.002490)	***	0.025823 (0.002484)	***	0.027212 (0.002462)	***	0.025280 (0.002338)	***	**0.023186** (0.002224)	0.023910 (0.002257)
MNAR	**30%**	0.016338 (0.003198)	***	0.014144 (0.002527)	*	0.014087 (0.002042)	**	0.015246 (0.002200)	***	0.013300 (0.002159)		**0.012762** (0.001951)	0.013428 (0.001997)
	40%	0.018693 (0.003191)	***	0.015768 (0.002661)	*	0.015805 (0.002172)	**	0.017585 (0.002389)	***	0.016123 (0.002211)	**	**0.014373** (0.001995)	0.015761 (0.001904)
	50%	0.020682 (0.003478)	***	0.017317 (0.002639)	**	0.017186 (0.002357)	**	0.018560 (0.002476)	***	0.017529 (0.002261)	***	**0.015561** (0.002069)	0.016017 (0.002023)
	60%	0.022766 (0.004194)	***	0.019123 (0.003063)	*	0.018714 (0.002411)		0.020331 (0.003190)	***	0.019269 (0.002351)	**	**0.017611** (0.002170)	0.018530 (0.002126)

Table 21
MAE for Adult data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.002077 (0.000539)	***	0.001936 (0.000439)	***	0.001907 (0.000336)	***	0.001948 (0.000440)	***	0.001821 (0.000243)	***	**0.001604** (0.000220)	0.001757 (0.000245)
	40%	0.002763 (0.000546)	***	0.002356 (0.000368)	***	0.002195 (0.000335)	**	0.002538 (0.000546)	***	0.002109 (0.000307)	*	**0.001953** (0.000249)	0.002004 (0.000263)
	50%	0.003422 (0.000636)	***	0.002996 (0.000450)	***	0.002735 (0.000354)	**	0.003191 (0.000557)	***	0.002575 (0.000321)		**0.002480** (0.000276)	0.002486 (0.000308)
	60%	0.004114 (0.000662)	***	0.003691 (0.000465)	***	0.003432 (0.000446)	***	0.003877 (0.000561)	***	0.003073 (0.000333)	**	**0.002847** (0.000337)	0.002907 (0.000341)
MNAR	**30%**	0.002224 (0.000529)	***	0.001610 (0.000318)	**	0.001491 (0.003027)		0.001716 (0.000429)	**	0.001556 (0.000213)	*	**0.001432** (0.000224)	0.001531 (0.000251)
	40%	0.002533 (0.000534)	***	0.001944 (0.000352)	***	0.001658 (0.003526)		0.002022 (0.000435)	***	0.001754 (0.000216)		**0.001662** (0.000209)	0.001742 (0.000234)
	50%	0.002729 (0.000627)	***	0.002063 (0.000438)	*	0.002013 (0.004645)		0.002315 (0.000457)	***	0.001933 (0.000222)		**0.001837** (0.000247)	0.001929 (0.000262)
	60%	0.002949 (0.000655)	***	0.002414 (0.000477)	**	0.002182 (0.004538)		0.002632 (0.000465)	***	0.002143 (0.000267)		**0.002138** (0.000278)	0.002170 (0.000283)

Table 22
CoD for Adult data set.

Missingness mechanism	Missing rate	Mean	T	kNNI	T	MLPI	T	FCMI	T	IARI	T	GFCMI (MLR)	GFCMI (SVR)
MAR	**30%**	0.986540 (0.004222)	***	0.987125 (0.002104)	***	0.987132 (0.002517)	***	0.985625 (0.002230)	***	0.994457 (0.001267)		**0.994525** (0.001243)	0.994384 (0.001125)
	40%	0.985719 (0.004548)	***	0.986790 (0.002746)	***	0.986782 (0.003191)	***	0.984827 (0.002451)	***	0.992532 (0.001383)		**0.992538** (0.001237)	0.992472 (0.001269)
	50%	0.982118 (0.005386)	***	0.985840 (0.003279)	***	0.985502 (0.003589)	***	0.982252 (0.002484)	***	**0.990813** (0.001728)		0.990796 (0.001894)	0.990717 (0.002031)
	60%	0.979073 (0.005412)	***	0.983423 (0.003284)	***	0.983530 (0.003795)	***	0.981234 (0.002710)	***	**0.988730** (0.002028)		0.988562 (0.002034)	0.988429 (0.002047)
MNAR	**30%**	0.985847 (0.004111)	***	0.997180 (0.003059)		0.996775 (0.003070)		0.995898 (0.002912)	**	0.997559 (0.002425)		**0.997723** (0.002217)	0.997604 (0.002134)
	40%	0.984413 (0.004525)	***	0.996162 (0.003265)		0.995638 (0.003242)		0.994484 (0.003022)	***	0.996708 (0.002605)		**0.996911** (0.002397)	0.996739 (0.002369)
	50%	0.983034 (0.005049)	***	0.995275 (0.003503)		0.994546 (0.003521)		0.993123 (0.003445)	***	0.995964 (0.002690)		**0.995971** (0.002668)	0.995917 (0.002708)
	60%	0.981432 (0.005319)	***	0.994129 (0.003576)		0.993303 (0.003751)		0.991541 (0.003514)	***	**0.994940** (0.003032)		0.994915 (0.002938)	0.994886 (0.003149)

Table 23
Classification accuracy for Iris data set.

Missingness mechanism	Imputation method	Classification method				
		C4.5	NB	SVM	kNN	DT
MAR	**Mean**	91.3 ± 2.1***	90.8 ± 2.4***	91.3 ± 2.1***	93.3 ± 2.3***	90.2 ± 2.1***
	kNNI	93.4 ± 1.2***	93.4 ± 1.3***	**94.7** ± 0.9	94.7 ± 1.0**	**94.3** ± 0.8
	MLPI	92.1 ± 1.3***	92.7 ± 1.1***	94.1 ± 0.7*	94.7 ± 1.5	90.6 ± 1.2***
	FCMI	92.0 ± 1.4***	92.2 ± 1.5***	92.8 ± 1.3***	94.3 ± 1.2***	90.8 ± 1.1***
	IARI	92.1 ± 1.1***	92.0 ± 1.0***	93.4 ± 0.8***	92.7 ± 0.9***	93.3 ± 0.7***
	GFCMI	**94.8** ± 1.2	**95.3** ± 0.9	**94.7** ± 1.2	**95.4** ± 1.1	**94.3** ± 0.9
MNAR	**Mean**	94.7 ± 1.5	92.8 ± 1.4***	91.7 ± 1.5***	95.1 ± 1.6	93.4 ± 1.3
	kNNI	94.4 ± 1.2	93.4 ± 1.1	**94.7** ± 1.4	94.0 ± 1.2**	93.5 ± 1.4*
	MLPI	94.4 ± 1.3	92.7 ± 1.2***	94.1 ± 1.5	**95.6** ± 1.3*	**94.1** ± 1.3
	FCMI	94.1 ± 1.1*	93.4 ± 1.3	**94.7** ± 1.2	94.8 ± 1.4	93.3 ± 1.5
	IARI	89.5 ± 1.5***	92.0 ± 1.1***	93.4 ± 1.3***	92.7 ± 1.3***	89.4 ± 1.2***
	GFCMI	**94.8** ± 1.4	**94.0** ± 1.3	**94.7** ± 1.1	94.8 ± 1.2	93.4 ± 1.1

GFCMI and IARI for Iris, Whole, Wine, Haberman, Chess, and Adult data sets in many cases.

(d) The obtained results show significant differences between the performance of FCMI and GFCMI. It suggests that GRG is an appropriate measure to determine the nearness between two instances in the fuzzy clustering process, especially for the imputation process.

(e) The results show that IARI and GFCMI methods offer lower standard deviation values than other methods in general. Therefore, these algorithms perform steadier than other imputation methods.

(f) The experimental results reveal that although SVR type of GFCMI operated better than MLR type in several data sets, and vice versa, they usually provide very similar values for performance metrics. The choice of SVR or MLR model for the prediction step of the proposed method mainly depends on characteristics of the input data set.

(g) Basic imputation methods, such as mean imputation, can produce good and efficient results only for simple imputation problems. For example, when there is a small number of missing values. The main reason for the weakness of these methods is that they do not explore relationships between attributes. For complex imputation problems, they usually perform worse than other methods by a wide margin.

5.3. Experimental evaluation on classification accuracy

One of the goals of data imputation is to improve results of mining algorithms. This section presents results of simulations performed to evaluate the effect of imputed values in classification tasks. To show the influence of various imputation methods on the classification accuracy (CA), GFCMI and other methods were evaluated for different data sets. The classification accuracy refers to the percentage of correctly classified samples. For different data sets, missing values were injected artificially under different missingness mechanisms with a 30% missing rate. Then, five different classification algorithms were applied to the imputed data sets to measure CA. The algorithms include C4.5 (decision tree), Naive Bayes (NB), k Nearest Neighbors (kNN), Decision Table (DT), and dummyTXdummy- Support Vector Machine (SVM). These classifiers are popular in data mining community. The WEKA System (Hall et al., 2009) was used to perform the simulations and all of the parameters were set to their defaults. The 10-fold cross validation resampling procedure was used to assess the accuracy of trained classifiers. That is, each data set was equally partitioned into ten subsets; each subset was used once for testing and the remaining subsets were used for training.

Tables 23–29 show the results of classification accuracy (± standard deviations) based on various imputation methods (rows) and classification algorithms (columns) for different data sets under MAR and MNAR missingness mechanisms. The best results among the imputation methods are presented in bold font. For example, in Iris data set when C4.5 was used to classify the data set under MAR, CA was increased from 91.3% when the mean imputation was used to 94.8% after the imputation by the proposed method. Note that selecting the best classifier for each dataset is not the main point in this section, but we are interested in evaluating the impact of different imputation methods, in the context of classification tasks. Presence of symbols "*", "**", and "***" in front of a method indicates that there is a significant difference between the classification accuracy of that method and the classification accuracy of GFCMI method at 90%, 95%, and 99% confidence levels, respectively.

From Table 23, it is clear that GFCMI method offers the greatest CA for all classification methods for the MAR pattern. Considering the MNAR pattern for this data set, in terms of CA after imputation, GFCMI was superior to other methods when C4.5, Naive Bayes, and SVM classifiers were used, and it had a suboptimal performance when kNN and Decision Table classifiers were employed. It is clear from Table 24 that our method obtained the best accuracies for Naive Bayes and kNN classifiers under both missingness patterns. Surprisingly, for other classification methods (C4.5, SVM, and DT), selecting different imputation approaches has no effect on the accuracy of the classification task and they produced similar results. The results from Table 25 show that GFCMI method yielded the highest CA percentages for all classifiers under the MAR and MNAR missingness mechanisms for Wine data set. Interestingly, for this dataset, simple imputation methods, such as mean and kNNI, have produced satisfactory results which were comparable to those of complex imputation methods.

It is apparent from Table 26 that the best imputation method was different for each classification algorithm in Glass data set. The results showed that even for different missing patterns, the best imputation method was different when we chose the same classification algorithms. In other words, different imputation methods exhibited different behaviors in terms of CA values for each classification algorithm. As we expected, based on the results of Section 5.2, the proposed method was not able to present the best performance for this data set. Table 27 demonstrates that C4.5, NB, and SVM classifiers could achieve better classification accuracies for MAR pattern in the Haberman data set when the proposed method was utilized. However, kNNI and MLPI resulted in higher accuracies for kNN and Decision Table classifiers, respectively. The GFCMI method also achieved the best performance for C4.5, NB, and SVM classifiers under MNAR missingness pattern for this data set. For kNN and Decision Table models, kNNI offered the highest CA values. We can also see a big improvement in CA for SVM classifier under MAR pattern using GFCMI method compared with

Table 24
Classification accuracy for Wholesale Customers data set.

Missingness mechanism	Imputation method	Classification method				
		C4.5	NB	SVM	kNN	DT
MAR	**Mean**	53.1 ± 2.0	55.9 ± 2.2	54.2 ± 1.8	54.2 ± 2.0*	51.6 ± 2.1
	kNNI	53.1 ± 1.5	55.4 ± 1.4***	54.2 ± 1.6	51.5 ± 1.5***	51.6 ± 1.7
	MLPI	53.1 ± 1.6	54.1 ± 1.5***	54.2 ± 1.3	54.1 ± 1.2***	51.6 ± 1.4
	FCMI	53.1 ± 1.5	55.3 ± 2.1**	54.2 ± 1.8	51.6 ± 1.9***	51.6 ± 2.0
	IARI	53.1 ± 1.2	55.6 ± 1.4**	54.2 ± 1.9	53.1 ± 1.3***	51.6 ± 1.5
	GFCMI	53.1 ± 1.2	**56.7** ± 1.3	54.2 ± 1.1	**55.2** ± 1.3	51.6 ± 1.2
MNAR	**Mean**	54.1 ± 1.7	55.7 ± 1.6**	55.2 ± 1.9	55.2 ± 1.4	51.6 ± 1.5
	kNNI	54.1 ± 1.4	56.2 ± 1.2	55.2 ± 1.5	54.3 ± 1.3***	51.6 ± 1.2
	MLPI	54.1 ± 2.1	55.7 ± 1.3**	55.2 ± 1.2	54.3 ± 1.4***	51.6 ± 1.5
	FCMI	54.1 ± 1.5	56.1 ± 1.6	55.2 ± 1.8	54.9 ± 1.7*	51.6 ± 1.3
	IARI	54.1 ± 1.3	56.2 ± 1.5	55.2 ± 1.4	54.3 ± 1.2***	51.6 ± 1.4
	GFCMI	54.1 ± 1.2	**56.8** ± 1.3	55.2 ± 1.5	**55.8** ± 1.3	51.6 ± 1.2

Table 25
Classification accuracy for Wine data set.

Missingness mechanism	Imputation method	Classification method				
		C4.5	NB	SVM	kNN	DT
MAR	**Mean**	91.0 ± 1.4***	96.7 ± 1.7	98.4 ± 1.4	**96.2** ± 1.3	85.9 ± 1.5***
	kNNI	91.0 ± 0.9***	96.7 ± 1.1**	98.4 ± 0.8**	**96.2** ± 0.8	85.9 ± 1.2***
	MLPI	90.5 ± 1.0***	95.6 ± 1.3***	97.3 ± 1.4***	95.7 ± 1.7	87.8 ± 1.6
	FCMI	90.5 ± 1.4***	96.7 ± 1.2*	97.8 ± 1.4***	95.7 ± 1.3	**88.0** ± 1.5
	IARI	89.3 ± 0.7***	96.2 ± 0.6***	97.8 ± 0.8***	95.2 ± 0.5***	87.2 ± 0.8***
	GFCMI	**93.3** ± 0.6	**97.3** ± 0.7	**98.9** ± 0.6	**96.2** ± 0.9	**88.0** ± 1.0
MNAR	**Mean**	93.3 ± 1.5***	95.6 ± 1.8***	**98.9** ± 1.5	95.8 ± 1.6**	86.9 ± 1.4***
	kNNI	94.4 ± 1.1*	96.7 ± 1.2	98.4 ± 1.0*	96.3 ± 1.1*	91.1 ± 1.4***
	MLPI	93.9 ± 1.3***	96.7 ± 1.5	98.4 ± 1.6	95.8 ± 1.8**	88.2 ± 1.6***
	FCMI	94.4 ± 1.5	96.1 ± 1.4***	98.4 ± 1.5	95.8 ± 1.2***	91.2 ± 1.3***
	IARI	94.4 ± 1.1*	96.6 ± 0.8**	98.4 ± 0.9**	96.3 ± 0.7**	91.6 ± 0.8***
	GFCMI	**95.0** ± 1.0	**97.2** ± 0.8	**98.9** ± 0.6	**96.8** ± 0.7	**93.6** ± 1.0

Table 26
Classification accuracy for Glass data set.

Missingness mechanism	Imputation method	Classification method				
		C4.5	NB	SVM	kNN	DT
MAR	**Mean**	65.1 ± 2.8	47.7 ± 2.7*	54.0 ± 3.0	**63.4** ± 2.4	67.1 ± 2.6***
	kNNI	**71.7** ± 2.3***	**49.7** ± 2.1	54.2 ± 2.5	61.4 ± 2.0**	68.2 ± 2.2***
	MLPI	68.2 ± 2.0***	49.2 ± 2.1	53.0 ± 2.3	62.9 ± 2.4	68.7 ± 2.3**
	FCMI	65.3 ± 2.7	48.9 ± 2.6	**63.4** ± 2.1*	53.6 ± 2.3***	67.9 ± 1.9***
	IARI	64.2 ± 1.8*	48.1 ± 1.9	55.0 ± 1.8***	63.2 ± 1.7	68.2 ± 1.6***
	GFCMI	65.1 ± 1.4	49.0 ± 1.7	54.2 ± 1.8	62.9 ± 1.6	**70.2** ± 1.7
MNAR	**Mean**	**68.5** ± 2.4	**48.6** ± 3.5	53.1 ± 3.2**	**67.4** ± 2.6	67.1 ± 2.7**
	kNNI	67.2 ± 2.7	47.7 ± 2.5	53.7 ± 2.3*	66.7 ± 2.1	68.0 ± 2.1
	MLPI	67.9 ± 2.1	46.2 ± 1.9**	54.0 ± 2.0	66.3 ± 2.3	69.1 ± 1.8
	FCMI	66.4 ± 2.0**	44.6 ± 1.7***	53.5 ± 1.7**	66.9 ± 1.9	67.2 ± 1.7**
	IARI	66.6 ± 1.5**	47.9 ± 1.6	**56.1** ± 1.4***	67.1 ± 1.2	**70.2** ± 1.3***
	GFCMI	67.6 ± 1.4	47.4 ± 1.6	54.8 ± 1.3	67.1 ± 1.2	68.5 ± 1.4

Table 27
Classification accuracy for Haberman data set.

Missingness mechanism	Imputation method	Classification method				
		C4.5	NB	SVM	kNN	DT
MAR	**Mean**	68.4 ± 1.5***	73.8 ± 1.7	54.1 ± 1.5***	66.1 ± 1.4	67.6 ± 1.2***
	kNNI	68.2 ± 1.4***	74.4 ± 1.3	54.1 ± 1.6***	**66.9** ± 1.2	69.3 ± 1.6
	MLPI	72.2 ± 1.5*	73.2 ± 1.4	54.1 ± 1.7***	62.4 ± 1.5***	**72.2** ± 1.6***
	FCMI	68.4 ± 1.5***	73.8 ± 1.7	54.1 ± 1.7***	66.1 ± 1.8	67.6 ± 1.9***
	IARI	64.8 ± 1.7***	72.1 ± 1.6***	54.1 ± 1.7***	65.5 ± 1.5**	54.1 ± 1.6***
	GFCMI	**73.0** ± 1.4	**73.9** ± 1.3	**67.4** ± 1.5	66.5 ± 1.6	69.3 ± 1.3
MNAR	**Mean**	66.1 ± 2.2***	54.1 ± 2.1***	62.9 ± 2.4***	63.0 ± 2.1	58.3 ± 2.0***
	kNNI	73.2 ± 2.0	**74.3** ± 1.9	62.9 ± 1.8***	**66.1** ± 1.7***	**75.0** ± 2.1
	MLPI	62.9 ± 1.6***	**74.3** ± 1.4	**67.4** ± 1.5	61.0 ± 1.6***	53.8 ± 1.7***
	FCMI	73.2 ± 1.7	**74.3** ± 1.7	**67.4** ± 1.8	62.3 ± 1.7*	63.0 ± 1.5***
	IARI	73.4 ± 1.5	**74.3** ± 1.3	**67.4** ± 1.7	65.2 ± 1.8***	74.4 ± 1.6
	GFCMI	**73.8** ± 1.2	**74.3** ± 1.4	**67.4** ± 1.5	63.3 ± 1.7	74.4 ± 1.6

Table 28
Classification accuracy for Chess data set.

Missingness mechanism	Imputation method	Classification method				
		C4.5	NB	SVM	kNN	DT
MAR	**Mean**	$95.6 \pm 1.0^{***}$	$82.9 \pm 1.6^{***}$	$92.6 \pm 1.3^{***}$	$94.4 \pm 1.5^{***}$	$95.2 \pm 1.5^{***}$
	kNNI	$98.3 \pm 0.7^{***}$	$80.9 \pm 1.3^{***}$	$95.5 \pm 1.2^{***}$	$\mathbf{97.2} \pm 1.0^{***}$	96.2 ± 1.4
	MLPI	$95.6 \pm 1.9^{***}$	$82.9 \pm 1.3^{***}$	$93.4 \pm 1.4^{***}$	96.7 ± 1.7	$94.6 \pm 1.7^{***}$
	FCMI	$96.6 \pm 1.8^{***}$	$81.3 \pm 2.1^{***}$	$95.6 \pm 1.2^{***}$	96.4 ± 2.1	96.5 ± 1.9
	IARI	$98.3 \pm 0.6^{***}$	$78.2 \pm 1.5^{***}$	$95.3 \pm 1.0^{***}$	$97.0 \pm 0.8^{**}$	$\mathbf{96.6} \pm 0.8$
	GFCMI	$\mathbf{99.3} \pm 0.4$	$\mathbf{90.8} \pm 1.1$	$\mathbf{97.2} \pm 0.9$	96.4 ± 0.7	$\mathbf{96.6} \pm 1.0$
MNAR	**Mean**	$85.0 \pm 1.9^{***}$	$80.0 \pm 2.1^{***}$	$88.1 \pm 1.8^{***}$	$92.3 \pm 1.7^{***}$	$85.0 \pm 1.8^{***}$
	kNNI	$88.4 \pm 1.8^{***}$	87.9 ± 1.6	$90.5 \pm 1.4^{**}$	$94.1 \pm 1.3^{**}$	$88.1 \pm 1.6^{***}$
	MLPI	$86.5 \pm 1.7^{***}$	$82.3 \pm 2.6^{***}$	$88.3 \pm 1.5^{***}$	$93.7 \pm 1.6^{***}$	$86.4 \pm 2.3^{***}$
	FCMI	$89.1 \pm 1.7^{**}$	$85.0 \pm 1.8^{***}$	$90.0 \pm 2.1^{**}$	$92.8 \pm 1.5^{***}$	$88.0 \pm 1.9^{***}$
	IARI	$88.9 \pm 1.5^{***}$	87.6 ± 1.4	91.1 ± 1.3	$\mathbf{95.3} \pm 1.4$	$\mathbf{89.7} \pm 1.5$
	GFCMI	$\mathbf{90.4} \pm 1.6$	$\mathbf{88.3} \pm 1.7$	$\mathbf{91.5} \pm 1.5$	95.1 ± 1.6	$\mathbf{89.7} \pm 1.6$

Table 29
Classification accuracy for Adult data set.

Missingness mechanism	Imputation method	Classification method				
		C4.5	NB	SVM	kNN	DT
MAR	**Mean**	$56.4 \pm 3.1^{***}$	$55.3 \pm 2.8^{***}$	$55.1 \pm 3.2^{**}$	62.1 ± 2.9	$53.3 \pm 3.1^{***}$
	kNNI	$58.6 \pm 2.7^{***}$	$58.6 \pm 2.6^{**}$	57.3 ± 3.0	62.3 ± 2.4	$56.4 \pm 2.7^{***}$
	MLPI	$59.7 \pm 2.4^{***}$	$58.3 \pm 2.6^{**}$	$\mathbf{57.8} \pm 2.5$	$\mathbf{62.5} \pm 2.1$	$56.7 \pm 2.3^{***}$
	FCMI	$56.4 \pm 2.6^{***}$	60.4 ± 2.4	$55.4 \pm 2.5^{**}$	62.4 ± 2.7	$55.2 \pm 2.9^{***}$
	IARI	62.6 ± 2.3	$\mathbf{61.8} \pm 2.4^{*}$	56.4 ± 2.7	$\mathbf{62.5} \pm 2.6$	$60.3 \pm 2.2^{***}$
	GFCMI	$\mathbf{63.5} \pm 2.2$	60.4 ± 2.4	57.2 ± 2.5	$\mathbf{62.5} \pm 2.0$	$\mathbf{62.6} \pm 1.9$
MNAR	**Mean**	$55.3 \pm 3.0^{***}$	$53.7 \pm 3.3^{***}$	$55.7 \pm 3.1^{***}$	62.1 ± 2.9	$54.0 \pm 2.8^{***}$
	kNNI	$\mathbf{67.1} \pm 2.7$	$63.0 \pm 2.5^{***}$	$57.8 \pm 2.8^{***}$	62.6 ± 2.3	$56.7 \pm 2.4^{***}$
	MLPI	$\mathbf{67.1} \pm 2.8$	$63.1 \pm 2.6^{***}$	$58.7 \pm 2.7^{***}$	$\mathbf{62.7} \pm 2.3$	$56.9 \pm 2.4^{***}$
	FCMI	$60.9 \pm 2.8^{***}$	$63.0 \pm 2.3^{***}$	$56.5 \pm 2.7^{***}$	62.3 ± 2.5	$55.4 \pm 2.7^{***}$
	IARI	$64.3 \pm 2.0^{***}$	$64.5 \pm 1.7^{***}$	$\mathbf{65.7} \pm 1.9$	62.3 ± 1.8	62.6 ± 1.8
	GFCMI	$\mathbf{67.1} \pm 1.7$	$\mathbf{67.2} \pm 1.9$	$\mathbf{65.7} \pm 2.1$	$\mathbf{62.7} \pm 1.9$	$\mathbf{63.1} \pm 1.7$

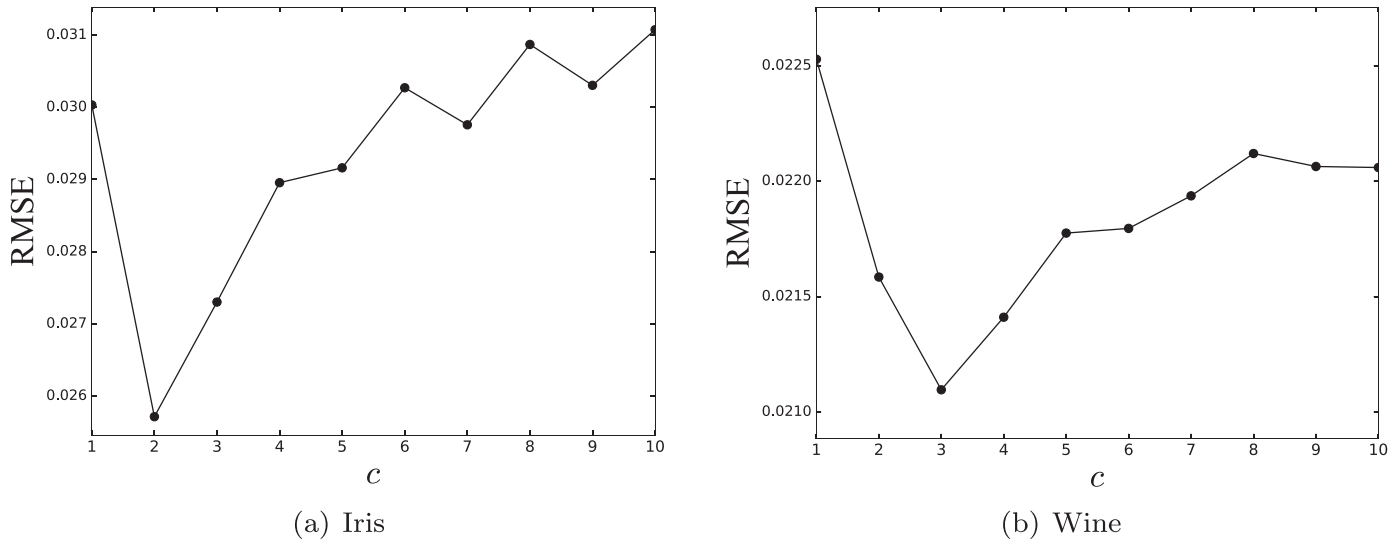

Fig. 2. The impact of parameter c on RMSE for data sets (a) Iris and (b) Wine.

other methods. Table 28 shows that GFCMI would appear to the best estimators of the missing values when C4.5, NB, SVM, and DT classifiers were applied to the Chess data set. However, using kNNI led to higher classification accuracies when the classification method was k nearest neighbor for Chess data set. From Table 29, it can be observed that when the missing data were injected to the Adult data set using the MNAR pattern, GFCMI obtained better classification accuracies than others, regardless of the type of classifier. However, for the MAR pattern, this is the case only for C4.5, kNNI, and DT classification methods. For NB and SVM classifiers, imputation by IARI and MLPI led to the highest accuracy, respectively. In summary, experimental results suggest that the use of GFCMI approach can help to enhance classification accuracies of different popular classifiers for incomplete data sets in many cases.

5.4. Impact of parameters c and θ on the performance

This section investigates impact of different input parameters on the performance of proposed method. GFCMI algorithm has two parameters c and θ that must be set. The parameter c specifies the number of fuzzy clusters in the grey based fuzzy c-means. The parameter θ defines a minimum mutual information value which

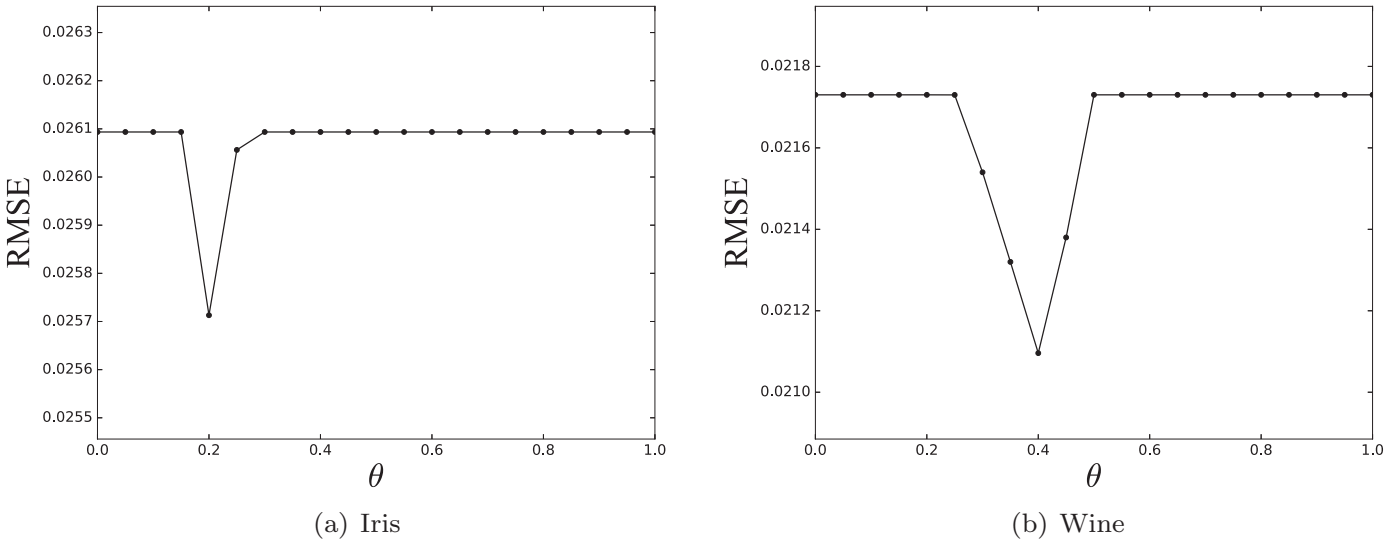

Fig. 3. The impact of parameter θ on RMSE for data sets (a) Iris and (b) Wine.

determines whether a feature will be involved in the imputation process or not.

To evaluate the effect of parameter c, its value was varied from 1 to 10 with intervals 1 at a fixed θ value and the obtained RMSE value was recorded for each c value. Fig. 2 illustrates the RMSE value for different c values on Iris and Wine data sets with 30% of missing values under MAR mechanism. For the remaining data sets and other evaluation criterion, results are not shown due to space limitations. It can be seen that the error value decreases until it reaches the optimal value and then it has an upward trend. The best RMSE value for Iris and Wine data sets appeared at $c = 2$ and $c = 3$, respectively. The optimal number of clusters for a data set mainly depends on the size of that data set. For example, the results shows that if the value of c is too large for a small data set, it may lead to empty clusters. Thus, c is an important parameter of the GFCMI algorithm.

To examine the impact of parameter θ, the value of c was kept at a fixed level and the value of θ was varied from 0.0 to 1.0 with an increasing step of 0.01. Then, RMSE value for each θ value was recorded. Fig. 3 represents the RMSE value for different θ values on Iris data set with 30% missing rate. Again, for the sake of space, we do not report the other cases. As the figure shows, $\theta = 0.2$ provides the best result for Iris data set. By comparison, the best θ value for Wine data is 0.4. Thus, for both data sets, the lowest RMSE value is obtained neither in $\theta = 0$ nor $\theta = 1$. This means that the mutual information based feature selection was able to reduce the prediction error. Note that, when no feature satisfies the minimum condition, all features will be involved, as explained in Section 3.2.3.

6. Conclusions

The imputation of missing data is a very important step in the preprocessing task. In this paper, we have proposed a new missing value imputation method called GFCMI which makes use of a novel fuzzy c-means clustering algorithm (GFCM) and mutual information. The main idea behind the method is to replace the grey relational analysis in the fuzzy clustering algorithm because of its superiority over the Euclidean distance in the domain of partially unknown information. Moreover, the mutual information was employed to select only highly related features in order to provide better estimations.

The proposed technique was compared with five other existing imputation approaches. The empirical tests on seven different data sets demonstrated that, in most cases, the proposed method was superior to the other imputation techniques in terms of RMSE, MAE, and CoD. In addition, we examined the impact of imputation on the accuracy of different classification approaches. The results showed that GFCMI method could lead to high accuracies according to most classification algorithms. Finally, the effect of different input parameters on the final performance of GFCMI was investigated.

There are some deficiencies in the proposed method. Firstly, training the regression models is an influential and challenging issue. In this paper, we have set the same parameters for all regression models applied on the clusters. However, it is clearly better to configure parameters of each regression model for each cluster separately, based on the instances of that cluster. Therefore, future work could include automating the selection of optimal parameters for each cluster. Secondly, the current version of the proposed method cannot handle categorical features directly. We perform an additional preprocessing step to transform categorical data into numerical data. Therefore, further researches are required to extend GFCMI for discrete domains. Thirdly, GFCMI requires more computing resources than other methods. However, in order to reduce the computational load, GFCMI can be parallelized by distributing the imputation process within each cluster on a node. In summary, our future work involves evaluating the proposed method on the other data sets, parallelizing the imputation process, and automating the selection of optimal parameters.

References

Ankaiah, N., & Ravi, V. (2011). A novel soft computing hybrid for data imputation. In *Proceedings of the 7th international conference on data mining (DMIN)*.

Aydilek, I. B., & Arslan, A. (2013). A hybrid method for imputation of missing values using optimized fuzzy c-means with support vector regression and a genetic algorithm. *Information Sciences, 233*(1), 25–35. doi:10.1016/j.ins.2013.01.021.

Ayuyev, V. V., Jupin, J., Harris, P. W., & Obradovic, Z. (2009). In T. B. Pedersen, M. K. Mohania, & A. M. Tjoa (Eds.), *Dynamic clustering-based estimation of missing values in mixed type data* (pp. 366–377)). Berlin, Heidelberg: Springer Berlin Heidelberg.

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.

Batista, G. E., Monard, M. C., et al. (2002). A study of k-Nearest Neighbour as an imputation method. In *HIS: 87* (p. 48).

Beretta, L., & Santaniello, A. (2016). Nearest neighbor imputation algorithms: A critical evaluation. *BMC Medical Informatics and Decision Making, 16*(3), 74.

Bezdek, J. C. (2013). *Pattern recognition with fuzzy objective function algorithms*. Springer Science and Business Media.

Bezdek, J. C., Ehrlich, R., & Full, W. (1984). FCM: The fuzzy c-means clustering algorithm. *Computers and Geosciences, 10*(2–3), 191–203.

Bose, S., Das, C., Dutta, S., & Chattopadhyay, S. (2012). A novel interpolation based missing value estimation method to predict missing values in microarray gene expression data. In *Communications, devices and intelligent systems (CODIS), 2012 international conference on* (pp. 318–321). IEEE.

Breiman, L. (2001). Random forests. *Machine Learning, 45*(1), 5–32.

Bu, F., Chen, Z., Zhang, Q., & Yang, L. T. (2016). Incomplete high-dimensional data imputation algorithm using feature selection and clustering analysis on cloud. *The Journal of Supercomputing, 72*(8), 2977–2990.

Carmona, C. J., Luengo, J., González, P., & Del Jesus, M. J. (2012). An analysis on the use of pre-processing methods in evolutionary fuzzy systems for subgroup discovery. *Expert Systems with Applications, 39*(13), 11404–11412.

Chang, F., Guo, C.-Y., Lin, X.-R., & Lu, C.-J. (2010). Tree decomposition for large-scale SVM problems. *Journal of Machine Learning Research, 11*(Oct), 2935–2972.

Deb, R., & Liew, A. W.-C. (2016). Missing value imputation for the analysis of incomplete traffic accident data. *Information Sciences, 339*, 274–289.

Di Nuovo, A. G. (2011). Missing data analysis with fuzzy c-means: A study of its application in a psychological scenario. *Expert Systems with Applications, 38*(6), 6793–6797.

Di Zio, M., Scanu, M., Coppola, L., Luzi, O., & Ponti, A. (2004). Bayesian networks for imputation. *Journal of the Royal Statistical Society: Series A (Statistics in Society), 167*(2), 309–322.

Doove, L. L., Buuren, S. V., & Dusseldorp, E. (2014). Recursive partitioning for missing data imputation in the presence of interaction effects. *Computational Statistics and Data Analysis, 72*, 92–104. doi:10.1016/j.csda.2013.10.025.

Duan, L., Yue, K., Qian, W., & Liu, W. (2013). Cleaning missing data based on the Bayesian network. In *International conference on web-age information management* (pp. 348–359). Springer.

Evans, D. (2008). A computationally efficient estimator for mutual information. In *Proceedings of the royal society of London a: Mathematical, physical and engineering sciences: vol. 464* (pp. 1203–1215). The Royal Society.

Folch-Fortuny, A., Villaverde, A. F., Ferrer, A., & Banga, J. R. (2015). Enabling network inference methods to handle missing data and outliers. *BMC Bioinformatics, 16*(1), 283.

Fujikawa, Y., & Ho, T. (2002). Cluster-based algorithms for dealing with missing values. *Advances in Knowledge Discovery and Data Mining*, 549–554.

Gan, X., Liew, A. W.-C., & Yan, H. (2006). Microarray missing data imputation based on a set theoretic framework and biological knowledge. *Nucleic Acids Research, 34*(5), 1608–1619.

García, S., Luengo, J., & Herrera, F. (2015). *Data preprocessing in data mining*. Springer.

García-Laencina, P. J., Sancho-Gómez, J.-L., Figueiras-Vidal, A. R., & Verleysen, M. (2009). K nearest neighbours with mutual information for simultaneous classification and missing data imputation. *Neurocomputing, 72*(7), 1483–1493.

Garciarena, U., & Santana, R. (2017). An extensive analysis of the interaction between missing data types, imputation methods, and supervised classifiers. *Expert Systems with Applications, 89*, 52–65.

García-Laencina, P. J., Sancho-Gómez, J.-L., & Figueiras-Vidal, A. R. (2013). Classifying patterns with missing values using Multi-Task Learning perceptrons. *Expert Systems with Applications, 40*(4), 1333–1341. doi:10.1016/j.eswa.2012.08.057.

Gautam, C., & Ravi, V. (2015). Data imputation via evolutionary computation, clustering and a neural network. *Neurocomputing, 156*, 134–142.

Gelman, A., & Hill, J. (2006). *Data analysis using regression and multilevel/hierarchical models*. Cambridge University Press.

Guessoum, S., Laskri, M. T., & Lieber, J. (2014). Respidiag: A case-based reasoning system for the diagnosis of chronic obstructive pulmonary disease. *Expert Systems with Applications, 41*(2), 267–273.

Gupta, A., & Lam, M. S. (1996). Estimating missing values using neural networks. *Journal of the Operational Research Society, 47*(2), 229–238.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: An update. *ACM SIGKDD Explorations Newsletter, 11*(1), 10–18. doi:10.1145/1656274.1656278.

Han, J., Kamber, M., & Pei, J. (2011). *Data mining: Concepts and techniques* (3rd). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc..

Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics, 12*(1), 55–67.

Hruschka, E. R., Jr, Hruschka, E. R., & Ebecken, N. F. F. (2007). Bayesian networks for imputation in classification problems. *Journal of Intelligent Information Systems, 29*(3), 231–252.

Huang, C.-C., & Lee, H.-M. (2004a). A grey-based nearest neighbor approach for missing attribute value prediction. *Applied Intelligence, 20*(3), 239–252.

Huang, C.-C., & Lee, H.-M. (2004b). A grey-based nearest neighbor approach for missing attribute value prediction. *Applied Intelligence, 20*(3), 239–252.

Ishay, R. B., & Herman, M. (2015). A novel algorithm for the integration of the imputation of missing values and clustering. In *International workshop on machine learning and data mining in pattern recognition* (pp. 115–129). Springer.

Jiang, C., & Yang, Z. (2015). In D.-S. Huang, & K. Han (Eds.), *CKNNI: An improved KNN-based missing value handling technique* (pp. 441–452)). Cham: Springer International Publishing.

Ju-Long, D. (1982). Control problems of grey systems. *Systems and Control Letters, 1*(5), 288–294.

Julong, D. (1989). Introduction to grey system theory. *The Journal of Grey System, 1*(1), 1–24.

Junninen, H., Niska, H., Tuppurainen, K., Ruuskanen, J., & Kolehmainen, M. (2004). Methods for imputation of missing values in air quality data sets. *Atmospheric Environment, 38*(18), 2895–2907.

Kolen, J. F., & Hutcheson, T. (2002). Reducing the time complexity of the fuzzy c-means algorithm. *IEEE Transactions on Fuzzy Systems, 10*(2), 263–267.

Krishnamoorthy, R., Kumar, S. S., & Neelagund, B. (2014). A new approach for data cleaning process. In *Recent advances and innovations in engineering (ICRAIE), 2014* (pp. 1–5). IEEE.

Kwon, O., & Sim, J. M. (2013). Effects of data set features on the performances of classification algorithms. *Expert Systems with Applications, 40*(5), 1847–1857.

Li, D., Deogun, J., Spaulding, W., & Shuart, B. (2004). In S. Tsumoto, R. Słowiński, J. Komorowski, & J. W. Grzymała-Busse (Eds.), *Towards missing data imputation: A study of fuzzy k-means clustering method* (pp. 573–579)). Berlin, Heidelberg: Springer Berlin Heidelberg.

Li, D., Gu, H., & Zhang, L. (2010). A fuzzy c-means clustering algorithm based on nearest-neighbor intervals for incomplete data. *Expert Systems with Applications, 37*(10), 6942–6947.

Lichman, M. (2013). UCI machine learning repository. http://archive.ics.uci.edu/ml.

Little, R. J. A., & Rubin, D. B. (2002). *Statistical analysis with missing data*. John Wiley & Sons.

Lobato, F., Sales, C., Araujo, I., Tadaiesky, V., Dias, L., Ramos, L., & Santana, A. (2015). Multi-objective genetic algorithm for missing data imputation. *Pattern Recognition Letters, 68*, 126–131.

Loh, W. P., & H'ng, C. W. (2014). Data treatment effects on classification accuracies of bipedal running and walking motions. In *Recent advances on soft computing and data mining* (pp. 477–485). Springer.

Lv, Z., Zhao, J., Liu, Y., & Wang, W. (2016). Data imputation for gas flow data in steel industry based on non-equal-length granules correlation coefficient. *Information Sciences, 367*, 311–323.

Mohammed, E. A., Naugler, C. T., & Far, B. H. (2016). Breast tumor classification using a new OWA operator. *Expert Systems with Applications, 61*, 302–313.

Nelwamondo, F. V., Golding, D., & Marwala, T. (2013). A dynamic programming approach to missing data estimation using neural networks. *Information Sciences, 237*, 49–58.

Nishanth, K. J., Ravi, V., Ankaiah, N., & Bose, I. (2012). Soft computing based imputation and hybrid data and text mining: The case of predicting the severity of phishing alerts. *Expert Systems with Applications, 39*(12), 10583–10589.

Pan, L., & Li, J. (2010). K-nearest neighbor based missing data estimation algorithm in wireless sensor networks. *Wireless Sensor Network, 2*(2), 115.

Pan, R., Yang, T., Cao, J., Lu, K., & Zhang, Z. (2015). Missing data imputation by K nearest neighbours based on grey relational structure and mutual information. *Applied Intelligence, 43*(3), 614–632.

Patil, B. M., Joshi, R. C., & Toshniwal, D. (2010). In S. Ranka, A. Banerjee, K. K. Biswas, S. Dua, P. Mishra, R. Moona, S.-H. Poon, & C.-L. Wang (Eds.), *Missing value imputation based on k-mean clustering with weighted distance* (pp. 600–609)). Berlin, Heidelberg: Springer Berlin Heidelberg.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research, 12*, 2825–2830.

Pinzon-Morales, R.-D., Baquero-Duarte, K.-A., Orozco-Gutierrez, A.-A., & Grisales–Palacio, V.-H. (2011). In H. R. Arabnia, & Q.-N. Tran (Eds.), *Pattern recognition of surface EMG biological signals by means of hilbert spectrum and fuzzy clustering* (pp. 201–209)). New York, NY: Springer New York.

Purwar, A., & Singh, S. K. (2015). Hybrid prediction model with missing value imputation for medical data. *Expert Systems with Applications, 42*(13), 5621–5631.

Rahman, M. G., & Islam, M. Z. (2013). KDMI: a novel method for missing values imputation using two levels of horizontal partitioning in a data set. In *International conference on advanced data mining and applications* (pp. 250–263). Springer.

Rahman, M. G., & Islam, M. Z. (2016). Missing value imputation using a fuzzy clustering-based EM approach. *Knowledge and Information Systems, 46*(2), 389–422.

Raja, P. S., & Thangavel, K. (2016). Soft clustering based missing value imputation. In *Digital connectivity–social impact* (pp. 119–133). Springer.

Rubin, D. B. (2004). *Multiple imputation for nonresponse in surveys*: vol. 81. John Wiley & Sons.

Samad, T., & Harp, S. A. (1992). Self–organization with partial data. *Network: Computation in Neural Systems, 3*(2), 205–212.

Saravanan, P., & Sailakshmi, P. (2015). Missing value imputation using fuzzy possibilistic C means optimized with support vector regression and genetic algorithm. *Journal of Theoretical and Applied Information Technology, 72*(1).

Sharpe, P. K., & Solly, R. J. (1995). Dealing with missing values in neural network-based diagnostic systems. *Neural Computing and Applications, 3*(2), 73–77.

Silva-Ramírez, E.-L., Pino-Mejías, R., López-Coello, M., & Cubiles-de-la Vega, M.-D. (2011). Missing value imputation on missing completely at random data using multilayer perceptrons. *Neural Networks, 24*(1), 121–129.

Silva-Ramírez, E.-L., Pino-Mejías, R., & López-Coello, M. (2015). Single imputation with multilayer perceptron and multiple imputation combining multilayer perceptron and k-nearest neighbours for monotone patterns. *Applied Soft Computing, 29*, 65–74. doi:10.1016/j.asoc.2014.09.052.

Sim, J., Kwon, O., & Lee, K. C. (2016). Adaptive pairing of classifier and imputation methods based on the characteristics of missing values in data sets. *Expert Systems with Applications, 46*, 485–493. doi:10.1016/j.eswa.2015.11.004.

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*(12), 1–14.

Singh, N., Javeed, A., Chhabra, S., & Kumar, P. (2015). In N. R. Shetty, N. H. Prasad, & N. Nalini (Eds.), *Missing value imputation with unsupervised Kohonen Self Organizing Map* (pp. 61–76)). New Delhi: Springer India.

van Stein, B., & Kowalczyk, W. (2016). In J. P. Carvalho, M.-J. Lesot, U. Kaymak, S. Vieira, B. Bouchon-Meunier, & R. R. Yager (Eds.), *An incremental algorithm for*

repairing training sets with missing values (pp. 175–186)). Cham: Springer International Publishing.

Szczepaniak, P. S., & Lisboa, P. J. G. (2012). *Fuzzy systems in medicine*: vol. 41. Physica.

Tian, J., Yu, B., Yu, D., & Ma, S. (2014). Missing data analyses: A hybrid multiple imputation algorithm using gray system theory and entropy based on clustering. *Applied Intelligence, 40*(2), 376–388. doi:10.1007/s10489-013-0469-x.

Tran, C. T., Zhang, M., & Andreae, P. (2016). A genetic programming-based imputation method for classification with missing data. In *European conference on genetic programming* (pp. 149–163). Springer.

Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., et al. (2001). Missing value estimation methods for DNA microarrays. *Bioinformatics, 17*(6), 520–525.

Tsai, C. F., & Chang, F. Y. (2016). Combining instance selection for better missing value imputation. *Journal of Systems and Software, 122*(1), 63–71. doi:10.1016/j.jss.2016.08.093.

Tutz, G., & Ramzan, S. (2015). Improved methods for the imputation of missing data by nearest neighbor methods. *Computational Statistics and Data Analysis, 90*, 84–99.

Van Hulse, J., & Khoshgoftaar, T. M. (2014). Incomplete-case nearest neighbor imputation in software measurement data. *Information Sciences, 259*, 596–610.

Wang, G., Yeung, D.-Y., & Lochovsky, F. H. (2008). A new solution path algorithm in support vector regression. *IEEE Transactions on Neural Networks, 19*(10), 1753–1767.

Wang, H., & Wang, S. (2010). Mining incomplete survey data through classification. *Knowledge and information systems, 24*(2), 221–233.

Wu, C.-H., Wun, C.-H., & Chou, H.-J. (2004). Using association rules for completing missing data. In *Hybrid intelligent systems, 2004. HIS'04. Fourth international conference on* (pp. 236–241). IEEE.

Zhang, C., Zhu, X., Zhang, J., Qin, Y., & Zhang, S. (2007). In Z.-H. Zhou, H. Li, & Q. Yang (Eds.), *GBKII: An imputation method for missing values* (pp. 1080–1087)). Berlin, Heidelberg: Springer Berlin Heidelberg.

Zhang, S. (2011). Shell-neighbor method and its application in missing data imputation. *Applied Intelligence, 35*(1), 123–133.

Zhang, S. (2012). Nearest neighbor selection for iteratively kNN imputation. *Journal of Systems and Software, 85*(11), 2541–2552.

Zhang, S., Zhang, J., Zhu, X., Qin, Y., & Zhang, C. (2008). Missing value imputation based on data clustering. In *Transactions on computational science I* (pp. 128–138). Springer.