Advanced methods for missing values imputation based on similarity learning

Khaled M. Fouad [Equal first author, 1, 2] , **Mahmoud M. Ismail** [Corresp., Equal first author, 1] , **Ahmad Taher Azar** [1, 3] , **Mona M. Arafa** [1]

[1] Faculty of Computers and Artificial Intelligence, Benha University, Benha, Qaliobia, Egypt

[2] Faculty of Information Technology and Computer Science, Nile University, El Shikh Zaid, Giza, Egypt

[3] College of Computer & Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia

Corresponding Author: Mahmoud M. Ismail
Email address: mahmoud.ismael@fci.bu.edu.eg

The real-world data analysis and processing using data mining techniques often are facing observations that contain missing values. The main challenge of mining datasets is the existence of missing values. The missing values in a dataset should be imputed using the imputation method to improve the data mining methods' accuracy and performance. There are existing techniques that use k-nearest neighbors algorithm for imputing the missing values but determining the appropriate k value can be a challenging task. There are other existing imputation techniques that are based on hard clustering algorithms. When records are not well-separated, as in the case of missing data, hard clustering provides a poor description tool in many cases. In general, the imputation depending on similar records is more accurate than the imputation depending on the entire dataset's records. Improving the similarity among records can result in improving the imputation performance. This paper proposes two numerical missing data imputation methods. A hybrid missing data imputation method is initially proposed, called KI, that incorporates k-nearest neighbors and iterative imputation algorithms. The best set of nearest neighbors for each missing record is discovered through the records similarity by using the k-nearest neighbors algorithm (kNN). To improve the similarity, a suitable k value is estimated automatically for the kNN. The iterative imputation method is then used to impute the missing values of the incomplete records by using the global correlation structure among the selected records. An enhanced hybrid missing data imputation method is then proposed, called FCKI, which is an extension of KI. It integrates fuzzy c-means, k-nearest neighbors, and iterative imputation algorithms to impute the missing data in a dataset. Fuzzy c-means algorithm is selected because the records can belong to multiple clusters at the same time. This can lead to further improvement for similarity. FCKI searches a cluster, instead of the whole dataset, to find the best k-nearest neighbors. It applies two levels of similarity to achieve a higher imputation accuracy. The performance of the proposed imputation

techniques is assessed by using fifteen datasets with variant missing ratios for three types of missing data; MCAR, MAR, MNAR. These different missing data types are generated in this work. The datasets with different sizes are used in this paper to validate the model. Therefore, proposed imputation techniques are compared with other missing data imputation methods by means of three measures; the root mean square error (RMSE), the normalized root mean square error (NRMSE), and the mean absolute error (MAE). The results show that the proposed methods achieve better imputation accuracy and require significantly less time than other missing data imputation methods.

1   # Advanced methods for missing values imputation
2   # based on similarity learning

3   Khaled M. Fouad [Equal first author, 1, 2], Mahmoud M. Ismail [Corresp., Equal first author, 1], Ahmad Taher Azar
4   [1, 3], Mona M. Arafa [1]

5   [1] Faculty of Computers and Artificial Intelligence, Benha University, Benha, Qaliobia, Egypt

6   [2] Faculty of Information Technology and Computer Science, Nile University, El Shikh Zaid,
7   Giza, Egypt

8   [3] College of Computer & Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia

9   Corresponding Author:

10 Mahmoud M. Ismail [1]

11 Benha, Qaliobia, 13511, Egypt

12 Email address: mahmoud.ismael@fci.bu.edu.eg

13 ## Abstract
14 The real-world data analysis and processing using data mining techniques often are facing
15 observations that contain missing values. The main challenge of mining datasets is the existence
16 of missing values. The missing values in a dataset should be imputed using the imputation
17 method to improve the data mining methods' accuracy and performance. There are existing
18 techniques that use k-nearest neighbors algorithm for imputing the missing values but
19 determining the appropriate k value can be a challenging task. There are other existing
20 imputation techniques that are based on hard clustering algorithms. When records are not well-
21 separated, as in the case of missing data, hard clustering provides a poor description tool in many
22 cases. In general, the imputation depending on similar records is more accurate than the
23 imputation depending on the entire dataset's records. Improving the similarity among records can
24 result in improving the imputation performance. This paper proposes two numerical missing data
25 imputation methods. A hybrid missing data imputation method is initially proposed, called KI,
26 that incorporates k-nearest neighbors and iterative imputation algorithms. The best set of nearest
27 neighbors for each missing record is discovered through the records similarity by using the k-
28 nearest neighbors algorithm (kNN). To improve the similarity, a suitable k value is estimated
29 automatically for the kNN. The iterative imputation method is then used to impute the missing
30 values of the incomplete records by using the global correlation structure among the selected
31 records. An enhanced hybrid missing data imputation method is then proposed, called FCKI,
32 which is an extension of KI. It integrates fuzzy c-means, k-nearest neighbors, and iterative
33 imputation algorithms to impute the missing data in a dataset. Fuzzy c-means algorithm is
34 selected because the records can belong to multiple clusters at the same time. This can lead to
35 further improvement for similarity. FCKI searches a cluster, instead of the whole dataset, to find
36 the best k-nearest neighbors. It applies two levels of similarity to achieve a higher imputation

37 accuracy. The performance of the proposed imputation techniques is assessed by using fifteen
38 datasets with variant missing ratios for three types of missing data; MCAR, MAR, MNAR.
39 These different missing data types are generated in this work. The datasets with different sizes
40 are used in this paper to validate the model. Therefore, proposed imputation techniques are
41 compared with other missing data imputation methods by means of three measures; the root
42 mean square error (RMSE), the normalized root mean square error (NRMSE), and the mean
43 absolute error (MAE). The results show that the proposed methods achieve better imputation
44 accuracy and require significantly less time than other missing data imputation methods.

45- Introduction

46   Organizations today depend heavily on data gathering, storing, and processing for
47 different decision-making processes (Müller, Naumann & Freytag, 2003; Razavi-Far, Zio &
48 Palade, 2014). Data is gathered in several different ways, such as documents, surveys, sensors,
49 interviews, and observations (Chapman & Speers, 2005; Rahman & Islam, 2011). According to
50 various causes, such as human error, confusion, misunderstanding, equipment faults,
51 measurement error, noise generation during transformation, and non-response, data may be lost
52 or interrupted (Rahman & Islam, 2013a). If the data gathered is not completed, issues may occur
53 in the decision-making process. An incomplete dataset may also affect data mining models'
54 performance, resulting in a lack of computing process efficiency and an invalid and inefficient
55 outcome due to dataset gaps (Salleh & Samat, 2017). The main challenge of mining datasets is
56 the existence of missing values (Poolsawad et al., 2012). Extracting valuable information and
57 knowledge from incomplete datasets is difficult (Houari et al., 2014). The preprocessing
58 approach would play a significant role in the process of data mining (Fouad, El Shishtawy &
59 Altae, 2018). Therefore, it is essential to clean the dataset to ensure the high-quality mining (Han
60 & Kamber, 2013; Sree Dhevi, 2014).

61   The dataset is imputed from missing values using the imputation method, which is one of
62 the effective approaches to solve that issue and improve the accuracy and performance of the
63 data mining techniques. The accurate estimation of missing values plays a vital role in ensuring a
64 high level of data quality in many areas, such as healthcare (Azimi et al., 2019) and traffic
65 monitoring (Li et al., 2020). Identifying patterns of missing data is a crucial factor when
66 developing strategies for tackling incomplete data. In particular, the type of missing data can
67 significantly affect the accuracy of data mining techniques. The missing data can be divided into
68 three categories (Soley-bori, 2013; Salgado et al., 2016; Garciarena & Santana, 2017). The first
69 category is missing completely at random (MCAR). In MCAR, when missing in the dataset
70 occurs entirely at random, no specific pattern can be determined. For example, some patients
71 may have lost laboratory values due to incorrect handling of several lab results. The second
72 category is missing at random (MAR). In MAR, a specific pattern can be determined. The
73 possibility that a particular variable's value is missed or not for any observation depends on the
74 values of other variables; therefore, a common factor can be found in all observations that have
75 missing values. For example, a depression-examination registry may encounter data representing
76 MAR if males are less likely to complete a questionnaire about the severity of depression than
77 females. Therefore, the likelihood of completion of the questionnaire is related to their gender

78   (that is fully observed). The third category is missing not at random (MNAR). In MNAR,
79   missing data is neither MCAR nor MAR. In this case, the data that cause others to be missing are
80   unobserved. Following the previous example, the depression-examination registry may
81   encounter data representing MNAR if people with extreme depression are more likely to refuse
82   to complete the questionnaire about the severity of depression.

83   Varieties of methods that impute the missing values have been presented (Zhang et al.,
84   2006). In general, imputation performance depends heavily on choosing an acceptable
85   imputation method to impute missing data (Zhang et al., 2006). The performance of each
86   imputation method can be varied dependent on the types of missing data and datasets. In the
87   imputation of the missing data, the current approaches usually use the similarities of missing
88   rows with the other rows in the dataset and the correlations of the features (Sefidian &
89   Daneshpour, 2020). Therefore, the missing data imputation methods can be partitioned into two
90   main categories, which are global missing data imputation methods and local missing data
91   imputation methods (Cheng, Law & Siu, 2012; Feng et al., 2015). The global missing data
92   imputation includes the strategies that use the whole dataset's global correlation structure to
93   impute missing values found in the dataset. Several current imputation methods, such as iterative
94   imputation (Little & Rubin, 2002; Buuren & Groothuis-Oudshoorn, 2011; Pedregosa et al., 2011)
95   and expectation-maximization imputation (EMI) (Schneider, 2001; Junninen et al., 2004), are
96   considered in this category that was described in (Rahman & Islam, 2013b). The iterative
97   imputation is a multivariate imputation that uses the whole set of available features to predict
98   missing values. It is a sophisticated method that models each feature, which has missing values
99   as a function of other features in a round-robin fashion (Little & Rubin, 2002; Buuren &
100 Groothuis-Oudshoorn, 2011; Pedregosa et al., 2011). The local missing data imputation includes
101 the strategies that use only the records similar to the missing record to impute missing values
102 such as the k-nearest neighbor imputation (kNNI) (Batista & Monard, 2003). kNNI is an
103 effective method to impute missing values. However, it is expensive for a large dataset because it
104 is required to search within the entire dataset to find the most similar records. Moreover, k value
105 is determined by the user so determining the appropriate k value can be a challenging task
106 (Batista & Monard, 2003; Rahman & Islam, 2013b; Liu et al., 2015). Recently, a number of
107 methods based on k-means clustering algorithm have been proposed to solve the problem of
108 missing data imputation (Patil, Joshi & Toshniwal, 2010; Jiang & Yang, 2015). The basic idea
109 behind these techniques is to estimate a missing value in a record based on the cluster
110 information in which the missing record is located. But in the hard clustering, such as the k-
111 means algorithm, a record belongs to only one cluster. When records are not well-separated, as in
112 the case of missing data, hard clustering provides a poor description tool in many cases. Besides,
113 if the initial points are not chosen properly, the k-means algorithm may become stuck in a local
114 minimum state (Sefidian & Daneshpour, 2019). There are group of missing data imputation
115 techniques that make utilization of a decision tree for the horizontal partitioning such as DMI
116 (Rahman & Islam, 2011) and KDMI (Rahman & Islam, 2013b). This is computationally
117 expensive (Razavi-Far et al., 2020). Moreover, a decision tree algorithm may generate some
118 heterogeneous leaves, where the records in each heterogeneous leaf are not always very similar.
119 This may lead to poor imputation accuracy (Rahman & Islam, 2014). In general, the imputation
120 depending on similar records is more accurate than the imputation depending on the entire

121   dataset's records (Young, Weckman & Holland, 2011). Improving the similarity among records
122   can result in improving the imputation performance. Another issue of interest is finding a
123   strategy that can impute the missing values in large-scale datasets (Fouad & Elbably, 2020). A
124   missing data imputation method should be time-efficient, which means that it should not rely on
125   the entire dataset for imputing missing records (Razavi-Far et al., 2020).

126   This paper initially proposes a hybrid missing data imputation method, called KI, that
127   consolidates k-nearest neighbors and iterative imputation algorithms to impute the missing
128   values in a dataset. The best set of nearest neighbors for each missing record is discovered based
129   on the similarity of records by the kNN algorithm. To improve the similarity, a suitable k value
130   is estimated automatically for the kNN without any user input. The iterative imputation method
131   is then used to impute the missing values of the incomplete records by using the global
132   correlation structure among the selected records. This technique applies only one level of
133   similarity. It can improve missing data imputation accuracy. However, it is expensive for a large
134   dataset because it is required to search within the entire dataset to find the most similar records
135   for each missing record.

136   This paper then proposes another enhanced hybrid missing data imputation method,
137   called FCKI, that is an extension of KI. It integrates fuzzy c-means, k-nearest neighbors, and
138   iterative imputation algorithms for imputing the missing data for a dataset. This technique
139   focuses on improving time efficiency for the proposed missing data imputation algorithm as well
140   as missing data imputation accuracy. It performs fuzzy c-means clustering for the dataset to
141   divide the dataset records into a c fuzzy clusters where the records in the same cluster are similar.
142   Then, it imputes each cluster separately using KI algorithm through two phases. In the first
143   phase, the best set of nearest neighbors for each missing record is discovered based on the
144   similarity of records by the kNN algorithm. Phase two focuses on exploiting the iterative
145   imputation method to impute the missing values of the incomplete records by using the global
146   correlation structure among the selected records. FCKI differs from KI, in that FCKI applies two
147   levels of similarity to achieve a higher imputation accuracy before imputing the missing values
148   through the iterative imputation. For the first level of similarity, fuzzy c-means clustering is
149   selected. The similarities of all records belonging to the cluster are higher than the similarities of
150   all the whole dataset records. Fuzzy c-means algorithm is selected because the records can
151   belong to multiple clusters at the same time. This can lead to further improvement for similarity.
152   For the second level of similarity, kNN is selected. It finds the best k records that are the utmost
153   similar to the missing record by using the Euclidean distance measure. FCKI, similar to KI finds
154   the best k value for the kNN automatically. FCKI searches a cluster, instead of the whole dataset,
155   to find the best k-nearest neighbors. The technique has the advantage of imputing missing values
156   based on the similarity of the set of records instead of the whole dataset. Iterative imputation is
157   then applied to discover the global correlation among the selected records similar to each other to
158   impute missing values. FCKI can improve time efficiency because It does not have many
159   iterations for imputing missing values in the dataset. It also focuses on both the similarity of data
160   records and the correlation among the features. Therefore, it can enhance imputation efficiency

161   and effectiveness where the most efficient imputation method should impute incomplete dataset

162   with the least amount of time, and the most effective imputation method should achieve the

163   highest imputation accuracy. Effectiveness of the imputation can be checked using different

164   imputation performance measures.

165   The scalability of the clustering algorithm is becoming increasingly important in today's

166   data mining applications due to the growing size of datasets (Ganti, Gehrke & Ramakrishnan,

167   1999). It's natural for a single processor machine to be unable to store the entire dataset in main

168   memory for processing, and frequent disk access results in a performance bottleneck. Scalable

169   and high-performance solutions can now be easily accomplished by implementing parallel

170   clustering algorithms, especially after the recent development of affordable parallel computing

171   platforms. Kwok et al. (2002) proposed the parallel fuzzy c-means (PFCM) algorithm for

172   clustering large datasets. parallel computers of the Single Program Multiple Data model type

173   with the Message Passing Interface are used to run this algorithm (Kwok et al., 2002). PFCM

174   algorithm can be used for FCKI algorithm instead of traditional FCM for clustering large

175   datasets.

176   It can be said that the proposed imputation methods, KI and FCKI, follow a hot deck

177   approach. Hot deck imputation includes using observed values from a respondent (the donor)

178   who is close to the non-respondent (the recipient) to fill in missing values for one or more

179   variables for a non-respondent (Andridge & Little, 2010).

180   The proposed imputation methods, KI and FCKI, consider datasets with missing values in

181   multiple numerical features. They are assessed by using fifteen known datasets with variant

182   missing ratios ranging from 1% to 20% of total attribute values for each type of missing data;

183   MCAR, MAR, MNAR. These different missing data types are generated in this work. The

184   datasets with different sizes are used in this paper to validate the model. These sizes are

185   gradually increased from small to large-scale. Small data is data in a volume and format that

186   makes it accessible, informative, and actionable. Large-scale data can be associated with data

187   that grows to a huge size over time. The proposed techniques are compared to different ten

188   missing data imputation methods, which are mean imputation (Ravi & Krishna, 2014), kNNI

189   (Batista & Monard, 2003; Rahman & Islam, 2013b; Liu et al., 2015), SoftImpute (Mazumder,

190   Hastie & Tibshirani, 2010), SVDImpute (Troyanskaya et al., 2001), traditional iterative

191   imputation (Little & Rubin, 2002; Buuren & Groothuis-Oudshoorn, 2011; Pedregosa et al.,

192   2011), EMI (Schneider, 2001; Junninen et al., 2004), DMI (Rahman & Islam, 2011), KDMI

193   (Rahman & Islam, 2013b), KEMI (Razavi-Far et al., 2020), and KEMI+ (Razavi-Far et al., 2020).

194   The evaluation of the proposed missing data imputation methods is performed by using three

195   imputation performance measures. These measures are the root mean square error (RMSE),

196   the normalized root mean square error (NRMSE), and the mean absolute error (MAE).

197   The structure of the paper is as follows. Section 2 provides some related works on

198   missing data imputation. Section 3 gives a formal presentation of the algorithms used to generate

199 different missing data types. The proposed missing data imputation methods, KI and FCKI, are
200 formally presented in Section 4. Section 5 presents results and discussion and compares KI and
201 FCKI with other missing data imputation methods. Section 6 gives the conclusions and future
202 work.

203 **2- Related work**

204 Recently, the imputation of missing values has attracted more and more attention from
205 researchers. There are two primary techniques are used to impute missing data. The first
206 technique aims at ignoring records that contain missing values. Although it is simple, it is an
207 inefficient method, particularly through high missing rate datasets. The second primary
208 technique is estimating the missing values, so-called missing data imputation (Bethlehem, 2009).
209 Several missing data imputation methods have been proposed, and they can show significant
210 variations in terms of complexity and quality of the imputation. This section presents several
211 missing data imputation methods.

212 Mean imputation is the most basic method used by previous researchers. It replaces the
213 missing value with the mean of non-missing values for the attribute. If there are several missing
214 values in an attribute, they all will be replaced by the same value because mean imputation
215 generates only one imputed value. It does not preserve the correlation among the features. As
216 most research studies are concerned with the relationships among features, mean imputation is
217 not a good solution (Ravi & Krishna, 2014).

218 Hot deck imputation includes using observed values from a respondent (the donor) who
219 is close to the non-respondent (the recipient) to fill in missing values for one or more variables
220 for a non-respondent. The donor is chosen at random from a group of possible donors in some
221 versions. These techniques are named random hot deck techniques. In other variants, a single
222 donor is defined, and values are estimated from that case, typically the "nearest neighbor". Since
223 no randomness is involved in the selection of the donor, these methods are known as
224 deterministic hot deck methods (Andridge & Little, 2010). One of the most common
225 deterministic hot deck imputation methods is sequential nearest neighbor hot deck imputation.
226 Traditional hot deck imputation is another name for this form. The first step in this method is to
227 define imputation classes using some auxiliary variables. Second, a single value, such as the
228 class mean or any pre-specified value, is assigned as a starting point for each imputation class.
229 The data file's records are then treated in sequence. If a record has a response for the target
230 variable, that value takes the place of previously stored value for its imputation class. If a record
231 has a missing value for a target attribute, the value currently stored for its imputation class is
232 assigned to it. The major drawback of this method has is that it is more likely to leads to multiple
233 use of donors, a feature which leads to a loss of precision in estimation (Hu & Salvucci, 2001).

234 k-nearest neighbor imputation (kNNI) is an effective method to impute missing values. It
235 first identifies k-nearest neighbors, which are the most similar to the missing record among all
236 records within the dataset by using the euclidean distance (k is determined by the user). kNNI
237 uses the mean value of the feature, which has the missing value within the selected nearest

238 neighbors. KNNI imputation accuracy is better than mean imputation accuracy which calculates
239 the mean from the entire dataset instead of the k-nearest neighbors of missing record within the
240 dataset. kNNI is an efficient and unpretentious method. However, it is expensive for a large
241 dataset because it is required to search within the entire dataset to find the most similar records.
242 Moreover, determining the appropriate k value can be a challenging task (Batista & Monard,
243 2003; Rahman & Islam, 2013b; Liu et al., 2015).

244   SVDimpute algorithm uses a low-rank SVD approximation to impute the missing values
245 in an incomplete dataset. It firstly initializes all missing values to the column means. Until
246 convergence, it computes a rank-k approximation to the completed matrix. It replaces the
247 previously missing values with corresponding values from the rank-k approximation obtained in
248 the previous step. (Troyanskaya et al., 2001).

249   SoftImpute is used for solving the problem of matrix completion. It fits a low-rank matrix
250 approximation to the matrix, which contains the missing values via nuclear-norm regularization.
251 It makes use of soft-thresholded SVD to impute missing values. (Mazumder, Hastie &
252 Tibshirani, 2010).

253   Iterative imputation is a multivariate imputation that uses the whole set of available
254 features to predict missing values. A sophisticated method that considers each feature that has
255 missing values depends on other features, and this estimation is used for imputation. It does so in
256 an iterated round-robin fashion: at each iteration, a feature with missing values is specified as
257 output y, and the other features are handled as inputs x. a regressor is fit on (X, y) for known y.
258 the regressor is then used to estimate the missing values of y. This is iteratively executed for each
259 feature and is repeated for several imputation rounds. The final imputation round results are
260 returned (Little & Rubin, 2002; Buuren & Groothuis-Oudshoorn, 2011; Pedregosa et al., 2011).

261   Expectation-Maximization Imputation (EMI) algorithm uses the mean and covariance
262 matrix of the dataset for imputing the missing numerical values of the incomplete dataset. It
263 firstly computes the derived matrix that includes the mean and the covariance values of the
264 dataset, which contains missing values. It then imputes the missing values through the mean and
265 covariance matrix (Schneider, 2001; Junninen et al., 2004). The main downside of this approach
266 is that it uses information from the entire dataset to impute the missing value, so it is only
267 appropriate for the datasets which have strong correlations between the attributes (Deb & Liew,
268 2016).

269   There are group of techniques that use a decision tree for imputing the missing data such
270 as DMI (Rahman & Islam, 2011) and KDMI (Rahman & Islam, 2013b). DMI is an imputation
271 technique that is based on the decision tree to impute the missing data. DMI consolidates the
272 decision tree and the expectation-maximization algorithm (EM). Therefore, DMI split the dataset
273 into two sub-datasets. The first sub dataset contains records with no missing values, and the
274 second one contains records with missing values. It creates a collection of decision trees on the
275 first sub dataset considering the features which have missing values in the second sub dataset as

276   the class attributes. It assigns each missing record of the second sub dataset to the leaf where it
277   exists for the tree that considers the attribute, which has a missing value for the record, as the
278   class attribute. It lastly uses the EMI algorithm (Schneider, 2001; Junninen et al., 2004) for
279   imputing the missing values among the records within each leaf (Rahman & Islam, 2011). KDMI
280   is an enhanced version of the DMI. It has two levels of partitioning for a dataset. The first phase
281   is similar to the DMI algorithm, where it horizontally splits the data into a collection of portions
282   generated by decision tree leaves. The second phase uses the kNN algorithm to identify a set of
283   nearest neighbors, which are extremely similar to the missing record through all records that
284   exist in the decision tree leaf where the missing record is found in. It eventually uses the EMI
285   algorithm for imputing the missing values using the selected nearest neighbors (Rahman &
286   Islam, 2013b). DMI and kDMI make utilization of a decision tree for the horizontal partitioning.
287   This is computationally expensive. They build a decision tree for each feature having missing
288   values in the dataset, considering this feature as a class attribute. If there are a large number of
289   features having missing values, DMI and KDMI will build a large number of trees, even if the
290   dataset contains a small number of records (Razavi-Far et al., 2020). A decision tree algorithm
291   may generate some heterogeneous leaves, where the records in each heterogeneous leaf are not
292   always very similar. This may lead to poor imputation accuracy (Rahman & Islam, 2014). DMI
293   and KDMI divide the entire dataset into complete and incomplete records and then build trees
294   from complete records only, so if the dataset does not contain any complete record, the
295   algorithms will not be able to build any tree and will not be able to impute any incomplete
296   records. Therefore, there must be a minimum number of complete records for the imputation
297   process to be performed correctly. Another issue is that it is unclear how the imputation is
298   handled if the missing record falls into more than one leaf, which may happen if a record has
299   several missing values (Deb & Liew, 2016).

300   A number of methods based on k-means clustering algorithm have been proposed to
301   solve the problem of missing data imputation (Patil, Joshi & Toshniwal, 2010; Jiang & Yang,
302   2015). Patil, Joshi & Toshniwal (2010) proposed an efficient missing value imputation method
303   based on k-means clustering with weighted distance. They use the user-specified value k to
304   divide the dataset into clusters. Then determine the complete neighbor which is similar to the
305   missing instance. The missing value is then estimated by taking the average of the centroid value
306   and the neighbor's centroidal distance. Jiang & Yang (2015) proposed an improved KNN based
307   algorithm, called class-based k-clusters nearest neighbor imputation (CKNNI). The k-means
308   cluster algorithm and traditional kNN algorithm are used for imputing missing data. First,
309   CKNNI uses the k-means algorithm to cluster instances into clusters. Then, the nearest neighbor
310   from the collection of centroids in clusters is selected using kNN algorithm, and with the ones
311   from corresponding attributes in a chosen neighbor, missing values are imputed. The limitation
312   of using hard clustering, such as the k-means algorithm, is that a record belongs to only one
313   cluster. When records are not well-separated, as in the case of missing data, hard clustering
314   provides a poor description tool in many cases. Besides, if the initial points are not chosen

315   properly, the original k-means algorithm may become stuck in a local minimum state (Sefidian
316   & Daneshpour, 2019).

317   Razavi-Far et al. (2020) proposed two methods for missing data imputation, named
318   KEMI and KEMI+. KEMI tackles missing values that uses the k-nearest neighbors algorithm and
319   expectation-maximization imputation (EMI) algorithm to tackle missing values in an incomplete
320   dataset. It firstly uses the kNN algorithm to identify the best set of nearest neighbors, which are
321   the most similar to the missing record among all records that don't contain any missing value.
322   Then, EMI is used to impute the missing values using the selected nearest neighbors (Razavi-Far
323   et al., 2020). KEMI+ is an enhanced version of the KEMI. It is used to identify the collection of
324   superior k values that leads to the minimum imputation error. It uses EMI to derive the missing
325   values using a collection of top nearest neighbors. It eventually provides a collection of superior
326   estimations into the dempster–shafer fusion to fuse these estimations and return final estimation
327   (Razavi-Far et al., 2020). KEMI and KEMI+ divide the entire dataset into complete and
328   incomplete records and then impute the missing values using only complete records, so if the
329   dataset does not contain any complete record, the algorithms will stop and will not be able to
330   impute any incomplete record. Therefore, there must be a minimum number of complete records
331   for the imputation process to be performed correctly. Moreover, KEMI and KEMI+ are
332   expensive because they search within all complete records of the dataset to find the most similar
333   records for each missing record.

3- Algorithms used to generate missing data

335   As mentioned earlier, missing data has various types. The original datasets that are used
336   in this paper are not incomplete datasets. The specific types of missing values are generated in
337   this section based on three described algorithms. These algorithms are used to generate missing
338   values in a predefined ratio for datasets to simulate the various missing data types; MCAR,
339   MAR, MNAR. These algorithms are derived from Garciarena & Santana (2017). For each of the
340   three types of missing data mentioned in this paper, one method has been implemented. For the
341   experimentation part, we decided to add missing data with different missing ratios ranging from
342   1% to 20% of total attribute values. These are reasonable values for the amount of missing data
343   in real datasets. These algorithms randomly generate missing data to achieve the delivered
344   datasets that are nearly real. The dataset positions to be modified are selected randomly
345   according to the specified conditions for each missing data type. It aims to generate instances of
346   the dataset with variant missing values for each execution of the algorithm.

347   3.1. MCAR

348   The algorithm used to generate MCAR is straightforward, where it first generates two
349   random numbers and used them as coordinates in the dataset. Then, it changes the value
350   indicated by these coordinates to "NaN." This process is iterated until the predefined missing
351   ratio is reached. The pseudo-code is presented in Algorithm 1.

Algorithm 1: MCAR generating algorithm

Input:

D: A complete dataset

MDR: Missing data ratio

Output: Incomplete dataset with MDR % missing ratio of total attribute values of D

Definitions:

N is the total number of records in D

M is the total number of attributes in D

X is the index of a random record

Y is the index of a random attribute

R is the current missing ratio of total attribute values of D

Begin

    while true **do**

        X ← Random([0,N]) /* Generate random record index between 0 and N */

        Y ← Random([0,M]) /* Generate random attribute index between 0 and M */

        if D[X,Y] != NaN **then**

            D[X,Y] = NaN

            R ← GetCurrentMDRatio(D) /* Return the current missing ratio of total attribute

values of D */

            If R ≥ MDR **then**

                break

            end

        end

    end

    Return incomplete dataset D

end

---

352   3.2. MAR

353   The algorithm used to generate MAR is less straightforward than the previous one, where

354   it needs to determine which attribute is the causative of the missing data. Garciarena & Santana

355   (2017) presumes a single attribute as causative, but causative attributes of MAR can be multiple.

356   Therefore, the algorithm first selects the causative attribute randomly and reproduces its values

357   to a new vector (aux). The next step is to select randomly the attributes that will lose their values

358   (MDAttributes). These attributes are called dependent because causative attribute tends to cause

359   their values to be lost. It then determines which records will lose their values for the dependent

360   attributes (MDRecords) by selecting the minimum values for the causative attribute (aux) and

361   making them incompetent by allocating them a great number. The process of choosing records is

362   repeated until the predefined missing ratio is reached. Finally, it uses the nested loop to generate

363   the missing data using the selected attributes and the selected records. The pseudo-code is

364   presented in Algorithm 2.

Algorithm 2: MAR generating algorithm

Input:

D: A complete dataset

MDR: Missing data ratio

NA: Attributes number losing their values

Output: Incomplete dataset with MDR % missing ratio of total attribute values of D

Definitions:

N is the total number of records in D

M is the total number of attributes in D

V is the total number of dataset values

Causative is the attribute that causes data loss in other attributes

MV is the total number of missing values

X is the index of a random record

Y is the index of a random attribute

R is the current missing ratio of total attribute values of D

MDAttributes is the list of attributes that lose their values

MDRecords is the list of records that lose their values

Aux is the vector of causative attribute values

MinIndex is the index of the minimum value in Aux

Begin

    V = N * M /* Calculate the total number of dataset values */

    Causative ← Random([0,M]) /* Generate random attribute index between 0 and M */

    MDAttributes = []

    for i = 1, . . . , NA **do**

        do

            Y ← Random([0,M]) /* Generate random attribute index between 0 and M */

        while Y in MDAttributes

        MDAttributes.Append(Y) /* Append Y to MDAttributes */

    end

    MDRecords = []

    MV = 0

    Aux = D[:,Causative] /* Return the vector of causative attribute values */

    while true **do**

        MinIndex ← FindMinIndex(Aux) /* Find the index of the minimum value in Aux */

        MDRecords.Append(MinIndex) /* Append MinIndex to MDRecords */

        Aux[MinIndex] = MaxInt /* MaxInt is the maximum integer that can be interpreted to a programming language, and it is utilized to deliver this index ineligible */

        MV += NA /* Missing values are incremented by the number of attributes which lose their values */

        R = (MV / V)*100 /* Calculate the current missing ratio of total attribute values of D */

> **If** R ≥ MDR **then**
> > **break**
>
> **end**
>
> **end**
>
> **foreach** record i ∈ MDRecords **do**
>
> > **foreach** attribute j ∈ MDAttributes **do**
> >
> > > D[i,j] = NaN
> >
> > **end**
>
> **end**
>
> Return incomplete dataset D
>
> **end**

3.3. MNAR

365

366   The algorithm used to produce MNAR is quite similar to the one used for producing
367 MAR. In MNAR, the causative attribute is unknown. Therefore, the records that will lose their
368 values will be selected randomly. The pseudo-code is presented in Algorithm 3.

Algorithm 3: MNAR generating algorithm

Input:

D: A complete dataset

MDR: Missing data ratio

NA: Attributes number losing their values

Output: Incomplete dataset with MDR % missing ratio of total attribute values of D

Definitions:

N is the total number of records in D

M is the total number of attributes in D

V is the total number of dataset values

MV is the total number of missing values

X is the index of a random record

Y is the index of a random attribute

R is the current missing ratio of total attribute values of D

MDAttributes is the list of attributes that lose their values

MDRecords is the list of records that lose their values

Begin

> V = N * M /* Calculate the total number of dataset values */
>
> MDAttributes = []
>
> **for** i = 1, . . . , NA **do**
>
> > **do**
> >
> > > Y ← Random([0,M]) /* Generate random attribute index between 0 and M */
> >
> > **while** Y in MDAttributes
> >
> > MDAttributes.Append(Y) /* Append Y to MDAttributes */

```
        end
        MDRecords = []
        MV = 0
        while true do
            X ← Random([0,N])  /* Generate random record index between 0 and N */
            if X not in MDRecords  then
                MDRecords.Append(X)  /* Append X to MDRecords */
                MV += NA  /* Missing values are incremented by the number of attributes which
lose their values */
                R = (MV / V)*100  /* Calculate the current missing ratio of total attribute values of D
*/
                If R ≥ MDR then
                    break
                end
            end
        end
        foreach record i ∈ MDRecords do
            foreach attribute j ∈ MDAttributes do
                D[i,j] = NaN
            end
        end
        Return incomplete dataset D
    end
```

369

34- Proposed missing data imputation methods

371   This paper initially proposes a hybrid missing data imputation method, called KI. It
372 consolidates k-nearest neighbors and iterative imputation algorithms to impute the missing
373 values in a dataset. The general scheme for KI is visualized in Figure 1. The best set of nearest
374 neighbors for each missing record is discovered based on similarity of records using the kNN
375 algorithm. The iterative imputation method is then used to impute the missing values of the
376 incomplete records by using the global correlation structure among the selected records. This
377 technique applies only one level of similarity. It can improve missing data imputation accuracy.
378 However, it is expensive for a large dataset because it is required to search within the entire
379 dataset to find the most similar records for each missing record.

380   kNN algorithm is selected because it is commonly one of the utmost machine learning
381 methods to search for local similarities. It finds the best k records that are the utmost similar to
382 the missing record based on the Euclidean distance measure (Santos et al., 2020).

383   In KI, an appropriate k value is estimated automatically for the kNN. Initially, the
384   algorithm creates a random missing value r_{iz} for a record R_i that includes an actual missing value
385   r_{ij}. For each possible k value (ranging from 2 to N where N is the number of records), kNN finds
386   the k-most similar records to R_i. The algorithm then imputes the missing value of r_{iz} using the
387   mean of the zth attribute over all records related to k-nearest neighbors of R_i. The algorithm then
388   calculates the root mean squared error (RMSE) using the imputed value and the actual value of
389   r_{iz}. Note that RMSE values for all sets of k-nearest neighbors of R_i for the same r_{iz} are computed.
390   The best k value is extracted from the set of k-nearest neighbors that deliver the minimal value of
391   RMSE (Rahman & Islam, 2013b).

392   The pseudo-code of KI is presented in Algorithm 4. The definitions of notations used in
393   Algorithm 4 are presented in Table 1. The main steps of the KI algorithm are explained in the
394   following:

395   Step 1 – Create a subset D_{mis} that contains a set of records in D which have missing
396   values.

397   Step 2 – Select an incomplete record R_i from D_{mis}.

398   Step 3 – Find a set of attributes A_{mis} which contain missing values in R_i.

399   Step 4 – Create a pool P that contains the set of records in D which haven't missing
400   values in corresponding attributes A_{mis}.

401   Step 5 – Add the incomplete record R_i into the pool $P = P \cup R_i$.

402   Step 6 – Copy the pool P into Temp.

403   Step 7 – Find a random index z of an attribute within R_i for which the value of r_{iz} is
404   available.

405   Step 8 – Preserve the value $r_{iz} \in R_i$ into a variable of actual value (AV).

406   Step 9 – Create missing value into r_{iz} within Temp.

407   Step 10 – For each k value (ranging from 2 to $N_p - 1$ where N_p represents the number of
408   records in P), find the k-nearest neighbors d_k of R_i within Temp by using kNN algorithm. Then
409   impute r_{iz} using the kNNI algorithm and the k-nearest neighbors d_k. Based on the imputed value
410   $r_{iz}{'}$ and the actual value AV, calculate the $RMSE_k$. This has been iteratively performed $N_p - 1$
411   times, so $N_p - 1$ estimations and RMSE values are generated.

412   Step 11 – Sort ascendingly the obtained N_p - 1 RMSE values and find minimum RMSE
413   value.

414   Step 12 – determines the k value that produces the minimal RMSE value.

415   Step 13 – Find the k-nearest neighbors d_k of R_i within P using kNN algorithm employing
416   the best k value found in step 12 and create a subset S made up with d_k and R_i.

417   Step 14 – Feed the subset S to Iterative Imputer. Iterative Imputer imputes the missing
418   values in R_i and returns the imputed record R_i'.

419   Step 15 – Remove incomplete record R_i from D.

420   Step 16 – Add imputed record R_i' into D.

421   Step 17 – Remove incomplete record R_i from D_{mis}.

422   Step 18 – Return iteratively to step 2 as long as $D_{mis} \neq \oslash$ for imputing the rest of the
423   incomplete records in D_{mis}.

Algorithm 4: KI

Input: A dataset D_{N*M} with missing values
Output: A dataset D´ with all missing values imputed
Begin
    $D_{mis} \leftarrow$ Find a set of records in D which contain missing values
    while $D_{mis} \neq \o$ **do**
        $R_i \leftarrow$ Select an incomplete record from D_{mis}
        $A_{mis} = \{\}$
        for j = 1, . . . , M **do**
            if r_{ij} = NaN **then**
                $A_{mis} = A_{mis}$ U A_j /* Find a set of attributes which contain missing values in R_i */
            end
        end
        $P \leftarrow$ Find a set of records in D which haven't missing values in corresponding attributes
A_{mis}
        P = P U R_i /* Add the incomplete record R_i into the pool P */
        Temp = P /* Copy the pool P into Temp */
        $z \leftarrow$ Find a random index of the available value in R_i /* Find a random index z of an
attribute for which the value of r_{iz} is available */
        AV = r_{iz} /* Preserve the value $r_{iz} \in R_i$ into a variable of actual value (AV) */
        r_{iz} = NaN /* Create missing value into r_{iz} within Temp */
        for k = 2, . . . , N_p − 1 **do**
            $d_k \leftarrow$ FindKNNRecords(R_i, Temp, k); /* Find the k-nearest neighbors of R_i within
Temp by using kNN algorithm */
            $r_{iz}' \leftarrow$ kNNI(R_i, d_k, k) /* Impute r_{iz} using kNNI algorithm and the k-nearest
neighbors d_k */
            $RMSE_k \leftarrow$ CalculateRMSE(AV, r_{iz}'); /* Compute RMSE value between the existing
value AV and the imputed value r_{iz}' */

           end

           $k \leftarrow \text{argmin}\{RMSE_k\}$ /* Find the minimum RMSE and return corresponding k */

           $d_k \leftarrow \text{FindKNNRecords}(R_i, P, k);$ /* Find the k-nearest neighbors of R_i within P by using kNN algorithm */

           $S = d_k \cup R_i$ /* Create a subset made up with d_k and R_i */

           $R_i' \leftarrow \text{IterativeImputer}(S)$ /* Impute the missing values in R_i using Iterative Imputation and return R_i' */

           $D = D \setminus R_i$ and $D_{mis} = D_{mis} \setminus R_i$ /* Remove incomplete record R_i from D and D_{mis} */

           $D = D \cup R_i'$ /* Add imputed record R_i' into D */

      End

      $D' = D$

      Return complete dataset D'

    End

424

425   This paper then proposes another enhanced hybrid missing data imputation method,
426 called FCKI, that is an extension of KI. It integrates Fuzzy c-Means, k-nearest neighbors, and
427 iterative imputation algorithms for imputing the missing data for a dataset. This technique
428 focuses on improving time efficiency for the proposed missing data imputation algorithm as well
429 as missing data imputation accuracy. The general scheme for FCKI is illustrated in Figure 2.
430 FCKI performs fuzzy c-means clustering for the dataset to divide records of the dataset into c
431 fuzzy clusters where the records in the same cluster are more similar to each other. Then, it
432 imputes each cluster separately using KI algorithm through two phases. In the first phase, the
433 best set of nearest neighbors for each missing record is discovered based on records similarity by
434 the kNN algorithm. Phase two focuses on exploiting the iterative imputation method to impute
435 the missing values of the incomplete records by using the global correlation structure among the
436 selected records. Therefore, the technique applies two levels of similarity.

437   For the first level of similarity, fuzzy c-means clustering is selected. The clustering
438 approaches are often unsupervised strategies that can be utilized to break down data into sub-
439 groups or clusters using the similarities through records (Pinzon-Morales et al., 2011). They are
440 divided into two categories, named hard (crisp) clustering and fuzzy (soft) clustering. In the case
441 of crisp clustering, like the k-means algorithm, a record R_i assigned to one and only one cluster
442 to which R_i is the utmost similar (Rahman & Islam, 2016). In the case of fuzzy clustering
443 techniques, records on the frontiers between multiple clusters are not compelled to assign to one
444 of the clusters completely. The records can belong to multiple clusters at the same time (Sefidian
445 & Daneshpour, 2019). Each record has a membership degree between 0 and 1, showing its
446 partial membership.

447   The main k-means is used for imputing the missing data, but fuzzy clustering has many
448 advantages because it is more realistic contrariwise hard clustering in many situations. when
449 records are not well-separated, as is the case for missing data problems, it provides a better

450   description tool (Sefidian & Daneshpour, 2019). Besides, if the initial points are not chosen
451   properly, the original k-means algorithm may be stuck in the local minimum state. Continuous
452   membership values in the fuzzy clustering, on the other hand, provide the resulting algorithms
453   minimal sensitivity to achieve stuck at a local minimum (Li et al., 2004; García, Luengo &
454   Herrera, 2015).

455   The most popular soft clustering technique is the fuzzy c-Means (FCM) algorithm
456   (Bezdek, Ehrlich & Full, 1984; Sefidian & Daneshpour, 2019). The algorithm of the FCM
457   partitions a set of input data $\{R_1, R_2, \ldots, R_n\}$ into c fuzzy clusters $\{C_1, C_2, \ldots, C_c\}$ by
458   minimizing the following objective function, which is based on distance:

$$\sum_{i=1}^{n}\sum_{k=1}^{c}(\delta_{ik})^{m'}\|R_i - c_k\|^2 \qquad\qquad (1)$$

459    $R_i = [R^1, R^2, \ldots, R^m]^T$ represents an input record, and R^j refers to the value of jth attribute for R_i.
460   c_k denotes the kth cluster prototype (centroid). $m' \in (1, \infty)$ is a fuzzification parameter that
461   specifies how much the clusters can overlap. $\|.\|$ denotes the euclidean norm which is used to
462   measure the similarity of data record R_i to the center vector c_k. δ_{ik} is the likelihood value that
463   expresses the degree to which R_i belongs to the kth cluster (C_k), $\forall i, k: \delta_{ik} \in [0, 1]$. The increased
464   value of δ_{ik} expresses the increased association between R_i and C_k. The total association of R_i
465   with c clusters is equal to 1.

466   Based on fuzzy set theory, incomplete records may belong to multiple clusters at the
467   same time. A record R_i with a missing value is considered to have a membership degree (fuzzy
468   association) with each cluster. The cluster with which the record R_i has a higher membership
469   degree has a greater effect on the imputation than the cluster with the lower membership degree.
470   So when a missing record R_i belongs to two or more clusters, FCKI get imputed record R_i' only
471   from the cluster which has highest membership degree.

472   In cluster analysis, the elbow approach is a heuristic utilized in deciding the number of
473   clusters in a dataset. The approach comprises plotting the demonstrated difference as a function
474   of the number of clusters and picking the curve elbow as the number of clusters to be used
475   (Kodinariya & Makwana, 2013).

476   For the second level of similarity, kNN is selected. It finds the best k records that are the
477   utmost similar to the missing record by using the Euclidean distance measure. FCKI, similar to
478   KI finds the best k value for the kNN automatically.

479   The proposed imputation method has the advantage of tackling missing values based on
480   the similarity of the set of records instead of the whole dataset. Iterative imputation is applied to
481   discover the global correlation among the selected records that are similar to each other to
482   impute missing values. It does not have many iterations to impute missing values of the dataset,
483   which is a significant factor for improving time efficiency. It focuses on both the records'

484 similarity and the correlations through the features. Therefore, it can enhance imputation
485 effectiveness and efficiency.

486   The pseudo-code of FCKI is presented in Algorithm 5. The definitions of notations used
487 in Algorithm 5 are presented in Table 2. The main steps of the FCKI algorithm are explained in
488 the following:

489   Step 1 – Determine the optimal number of clusters c in dataset D using the elbow method
490 then partition dataset D into c fuzzy clusters $\{C_1, C_2,…, C_c\}$ using the fuzzy c-means algorithm.

491   Step 2 – Select a cluster C_i from Clusters.

492   Step 3 – Feed cluster C_i into KI algorithm to impute each missing record of C_i and return
493 imputed cluster $C_i′$.

494   Step 4 – Add imputed cluster $C_i′$ into D′.

495   Step 5 – Remove cluster C_i from Clusters.

496   Step 6 – Return iteratively to step 2 as long as Clusters $\neq \oslash$ for imputing the rest of the
497 incomplete clusters.

Algorithm 5: FCKI

Input: A dataset D_{N*M} with missing values
Output: A dataset D′ with all missing values imputed
Begin
     c = Elbow(D) /* Determine the optimal number of clusters in dataset D using elbow method
*/
     Clusters = FCM(D, c) /* Partition dataset D into c fuzzy clusters $\{C_1, C_2,…, C_c\}$ using fuzzy
c-means algorithm */
     foreach Cluster $C_i \in$ Clusters **do**
         $C_i′$ = KI(C_i) /* Feed cluster C_i into KI algorithm to impute each missing record of C_i and
return imputed cluster $C_i′$ */
         D′ = D′ U $C_i′$ /* Add imputed cluster $C_i′$ into D′ */
     End
     Return complete dataset D′
End

498

499 **5- Results and Discussion**

500 **5.1 Experimental Setup**

501   The experiments are performed on machine 1, which is configured with 2×4 core Intel
502 i7-7500U processor and 8 GB RAM. The source codes used in this work for all the various

503 imputation methods were implemented in Python 3.8.0 with the help of some Scikit-Learn
504 packages (Pedregosa et al., 2011).

505

5.2 Dataset Description

507   The proposed missing data imputation methods, KI, and FCKI, are evaluated on fifteen
508 datasets, which are shown in Table 3. These datasets are found in UCI Machine Learning
509 Repository. These datasets are commonly used in related work. These datasets are selected
510 according to three factors. The first one is that these datasets were used in many previous and
511 related works, therefore can be used for the comparison. The second factor is that these datasets
512 have no missing values, so we can generate incomplete datasets. This is critical for assessing
513 accuracy. Third factor is that these datasets are different in volume and the number of instances
514 and attributes. The evaluation depends on comparing KI and FCKI with ten different methods of
515 missing data imputation. These ten methods are mean imputation (Ravi & Krishna, 2014), kNNI
516 (Batista & Monard, 2003; Rahman & Islam, 2013b; Liu et al., 2015), SoftImpute (Mazumder,
517 Hastie & Tibshirani, 2010), SVDImpute (Troyanskaya et al., 2001), traditional iterative
518 imputation (Little & Rubin, 2002; Buuren & Groothuis-Oudshoorn, 2011; Pedregosa et al.,
519 2011), EMI (Schneider, 2001; Junninen et al., 2004), DMI (Rahman & Islam, 2011), KDMI
520 (Rahman & Islam, 2013b), KEMI (Razavi-Far et al., 2020), and KEMI[+] (Razavi-Far et al.,
521 2020). The selected datasets do not contain realistic missing values. Algorithms are used to
522 generate missing values in a predefined ratio for selected datasets to simulate the various types of
523 missing data; MCAR, MAR, MNAR were provided in Section 4. These algorithms randomly
524 generate missing data for achieving results as close as possible to reality. The dataset positions to
525 be modified are selected randomly based on the specified criterion for each missing data type.
526 The missing values are generated for each dataset with different missing ratios ranging from 1%
527 to 20% of total attribute values for each type of missing data; MCAR, MAR, MNAR. These lead
528 to the twelve different collections generated missing data. These collections are derived from
529 multiplying four missing ratios and three missing types over each dataset. The evaluation of the
530 proposed methods is applied over these generated 180 different datasets, derived from
531 multiplying fifteen datasets and twelve missing collections. The evaluation depends on using
532 three different imputation error measures, which are the root mean square error (RMSE),
533 the normalized root mean square error (NRMSE), and the mean absolute error (MAE).

534   The incomplete sonar dataset's visualization with MCAR missing data type and 10%
535 missing ratio is illustrated in Figure 3. This visualization can assist in showing missing and non-
536 missing values in the dataset. The non-missing values are represented by black cells, and the
537 missing values are represented by white cells. The graph shown in Figure 3 was generated using
538 a python library called missingno (Bilogur, 2018). This library provides the ability to know how
539 missing values are distributed via informative visualizations. The number of missing values in
540 each feature for an incomplete sonar dataset with MCAR missing data type and 10% missing
541 ratio is illustrated in Figure 4. The missing data is MCAR when missing in the dataset occurs
542 entirely at random, and there is no specific pattern can be determined as shown in Figure 3. The

543 incomplete sonar dataset's visualization through MAR missing data type and 10% missing ratio
544 is illustrated in Figure 5. The number of missing values in each feature for an incomplete sonar
545 dataset with MAR missing data type and 10% missing ratio is illustrated in Figure 6. The
546 missing data is MAR when a specific pattern can be determined. The probability that the value of
547 a particular variable is missed or not for any observation is depending on the values of other
548 variables, so a common factor can be found in all observations that have missing values. The
549 representation of the incomplete Sonar dataset with MNAR missing data type and 10% missing
550 ratio is illustrated in Figure 7. The number of missing values in each feature for an incomplete
551 Sonar dataset with MNAR missing data type and 10% missing ratio is shown in Figure 8. The
552 missing data is MNAR when the data that cause others to be missing are unobserved.

553 ## 5.3 Evaluation Criteria

554 The methods which are chosen as competitors are Mean Imputation, kNNI, SoftImpute,
555 SVDimpute, Iterative Imputation, EMI, DMI, KDMI, KEMI, and KEMI$^+$. These methods are
556 used because the literature review mentioned that there is a need to be applied for many
557 applications with different sizes of datasets and they have been shown to be effective. Some
558 recent research relied on these methods for comparison. Moreover, most of these methods rely
559 on data similarity, which is the basis of the proposed methods. These methods gradually vary
560 from simple to advanced. These methods are compared to the proposed missing data imputation
561 methods, KI, and FCKI in terms of three imputation performance measures; the root mean square
562 error (RMSE), the normalized root mean square error (NRMSE), and the mean absolute error
563 (MAE) which are given by the following equations:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(P_i - O_i)^2} \qquad (2)$$

564

$$\text{NRMSE} = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(P_i - O_i)^2}}{O_{max} - O_{min}} \qquad (3)$$

565

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|P_i - O_i| \qquad (4)$$

566 P and O are the predicted value and the observed value, respectively. N is the total number of
567 records. O_{max} and O_{min} are the maximum observed value and the minimum observed value,
568 respectively.

569 ## 5.4 Results

570   The average values of RMSE obtained over twelve experiments for each dataset utilizing
571   each imputation method are shown in Table 4. The best imputation method is the one with the
572   lowest RMSE value among all of these imputation methods. The results show that FCKI
573   followed by KI performs significantly better than mean, kNNI, SoftImpute, SVDimpute,
574   Iterative Imputation, EMI, DMI, KDMI, KEMI, and KEMI$^+$. The reported results in Table 4
575   show that FCKI and KI have the lowest average RMSE among the other imputation methods for
576   all datasets (in Table 4, see the bold entries). The worst result was obtained by mean imputation.
577   This is because the mean imputation entirely ignores the correlations between the features.
578   Standard deviation of the average values of RMSE for each imputation method is presented in
579   the last row of Table 4. In statistics, standard deviation is a measure of distribution. It is used to
580   determine the spread and variation of a series of data values. A low standard deviation value
581   means that the data are less spread out, while a high standard deviation value indicates that the
582   data in a collection are distributed apart from their mean average values. The reported results of
583   standard deviation in Table 4 show that FCKI and KI have the lowest standard deviation among
584   the other imputation methods. This indicates that FCKI and KI methods can provide stable
585   imputation results over different datasets. The average NRMSE and MAE values derived from
586   twelve experiments for each dataset utilizing each imputation method are presented in Appendix
587   A as Table A1 and Table A2, respectively.

588   A boxplot of the imputation performance evaluation obtained over the datasets mentioned
589   in Table 3 with various missing data types; MCAR, MAR, MNAR, and missing ratios; 1%, 5%,
590   10%, 20% is illustrated in Figure 9. The box part of the boxplot is described by two lines at the
591   25th percentile and 75th percentile. The 25th percentile is the value at which 25% of the RMSE
592   values are less than this value. Thus, the middle 50% of the RMSE values fall between the 25th
593   percentile and the 75th percentile. The distance between the upper and lower lines of the box is
594   called the interquartile range (IQR), which is a common measure of the spread of the RMSE
595   values. A line inside the box is the median. The solid dashes represent the 1st and 99th
596   percentiles. The circles represent outliers. The solid triangles represent the averaged RMSE value
597   obtained by each imputation method. The 180 RMSE values obtained by each technique are in
598   each box. Figure 9 shows that the medians of the RMSE values for FCKI and KI methods are
599   lower than those of the other ten imputation methods, which indicates that FCKI and KI methods
600   have less error for missing data imputation. In practice, the method of missing data imputation
601   with a strong statistical property is that it has a lower RMSE. The distances between the 25th and
602   75th percentiles of RMSE values for FCKI and KI methods are the smallest and the normal
603   ranges of RMSE values for FCKI and KI methods are also the smallest, which shows that FCKI
604   and KI methods can provide stable imputation results over different datasets with various
605   missing data types; MCAR, MAR, MNAR, and missing ratios; 1%, 5%, 10%, 20%. The solid
606   triangles indicate that FCKI and KI have the lowest average overall RMSE values. The results
607   show that FCKI and KI outperform other imputation methods, and they are the most stable
608   techniques. Boxplots for distribution of NRMSE and MAE values obtained over the datasets
609   mentioned in Table 3 with various missing data types; MCAR, MAR, MNAR, and missing

610   ratios; 1%, 5%, 10%, 20% are presented in Appendix B as Figure B1 and Figure B2,
611   respectively.

612   The average values of RMSE and its distribution for all experiments implemented on all
613   used datasets mentioned in Table 3 are illustrated in Figure 10. The fifteen average values of
614   RMSE obtained by each technique are in each box. The average value of RMSE values also
615   indicates that FCKI and KI outperform other imputation techniques. The boxplots of the average
616   value of NRMSE and MAE values and its distribution for all experiments implemented on all
617   datasets mentioned in Table 3 are presented in Appendix C as Figure C1 and Figure C2,
618   respectively.

619   The average value of RMSE values for all datasets achieved by applying each imputation
620   method to each missing data type are shown in Table 5. The results show that FCKI and KI
621   outperform other imputation methods for all missing data types (in Table 5, see the bold entries).
622   The average value of NRMSE and MAE values for all datasets achieved by applying each
623   imputation method to each missing data type are presented in Appendix D as Table D1 and Table
624   D2, respectively.

625   The average value of RMSE values for all datasets achieved by applying each imputation
626   method to each missing ratio are shown in Table 6. The results show that FCKI and KI
627   outperform other imputation methods for all missing ratios (in Table 6, see the bold entries). The
628   average value of NRMSE and MAE values for all datasets achieved by applying each imputation
629   method to each missing ratio are presented in Appendix E as Table E1 and Table E2,
630   respectively.

631   The average value of RMSE values for MCAR, MAR, and MNAR missing data types,
632   respectively, overall datasets achieved by applying each imputation method to each missing ratio
633   are illustrated in Figures 11–13. The lowest average value of RMSE values are achieved by
634   applying FCKI and KI, and thus they outperform the other methods of imputation, as shown in
635   Figures 11–13. The average value of NRMSE and MAE values for MCAR, MAR, and MNAR
636   missing data types, respectively, for datasets achieved by applying each imputation method to
637   each missing ratio, are presented in Appendix F as Figures F1–F3 and Figures F4–F6,
638   respectively.

639   The missing data imputation methods are tested in order to compare them statistically.
640   First, a Friedman rank test with a significance level $\alpha = 0.05$ is performed. This test is used to
641   compare the variations between these imputation methods. It decides if one or more imputation
642   methods have significantly different performance. The test statistic is 141.420 and the
643   corresponding p-value is 0 for the obtained RMSE. Since the p-value is less than the significance
644   level of 0.05, the null hypothesis is rejected, and it is concluded that at least one of these
645   strategies has a different effect. As a result, a post-hoc test can be used to further assess the
646   variations statistically. This is used to compare all the algorithms in a pairwise manner, and it is
647   based on the absolute difference between the imputation methods' average rankings. For a

648   significance level $\alpha = 0.05$, the critical difference (CD) is 0.64. The null hypothesis that two
649   approaches have the same performance is rejected if the difference between their average
650   rankings is greater than CD. A diagram of the critical difference for the post-hoc Nemenyi test
651   (Gardner & Brooks, 2017) is shown in Figure 14. It compares all imputation methods with each
652   other in terms of RMSE. It illustrates the average rank of each imputation method, wherever the
653   first rank represents the method with the lowest RMSE. The methods are connected by thick
654   lines if they are not significantly different. This figure shows that KEMI$^+$ is around the third
655   rank, KEMI is around the fourth rank. They are followed by KDMI, DMI, Iterative Imputation,
656   SoftImpute, SVDimpute, EMI, kNNI, and Mean. It shows that mean is the least accurate method.
657   This figure shows that FCKI and KI outperform significantly other methods.

658   The evolution of the runtimes, which is studied concerning the dataset's volume, is
659   considered another matter of concern. The experiments investigate the scalability of the missing
660   data imputation methods by assessing runtimes for all datasets with different missing ratios
661   ranging from 1% to 20% of total attribute values for each type of missing data; MCAR, MAR,
662   MNAR. The type of time that is measured for these experiments is wall-clock time. The wall-
663   clock time measures the total time to execute a program on a computer. This time is measured
664   using a python module named time. The paper only shows the runtimes for advanced techniques;
665   DMI, KDMI, KEMI, KEMI$^+$, KI, FCKI. They take more time than other traditional techniques to
666   pay the price of a significantly better quality of imputation. The average execution times (in
667   seconds) achieved over twelve experiments for each dataset utilizing each advanced imputation
668   method are shown in Table 7. The average execution times of FCKI which are presented in
669   Table 7 include the clustering process but do not include finding the best c value. The results
670   show that FCKI performs significantly better than DMI, KDMI, KEMI, KEMI$^+$, and KI. Table 7
671   Indicates that FCKI has the lowest average execution time among the other advanced imputation
672   techniques for all datasets. The bold entries in Table 7 show the minimum and the maximum
673   average execution times among the datasets for FCKI algorithm. The minimum average
674   execution time is obtained by Zoo dataset. This time is very short because the dataset has only
675   101 records and 17 features. The maximum average execution time is obtained by Pseudo
676   Periodic Synthetic dataset. This time is large because the dataset has 100,000 records and ten
677   features. However, FCKI takes significantly less average execution time than other advanced
678   imputation techniques. The Average execution times (in seconds) overall datasets achieved by
679   applying each advanced imputation method to each missing data type are shown in Table 8. The
680   results indicate that FCKI outperforms the other advanced imputation methods for all missing
681   data types. The Average execution times (in seconds) overall datasets achieved by applying each
682   advanced imputation method to each missing ratio are shown in Table 9. The results Indicate that
683   FCKI outperforms the other advanced imputation methods for all missing ratios.

5.5 Discussion

685   In this paper, two missing data imputation methods are proposed, named KI and FCKI,
686   that aim to learn both similarities between the records and correlation among the features within

687   the dataset. Improving the similarity among records can result in improving the imputation
688   performance. This can further be improved considering the correlation among the features. KI
689   applies only one level of similarity using the kNN algorithm. To improve the similarity, KI
690   estimates a suitable k value automatically for the kNN by creating a random missing value r_{iz} for
691   the missing record R_i that includes an actual missing value r_{ij}. For each possible k value, kNN
692   finds the k-most similar records to R_i. The algorithm then imputes the missing value r_{iz} through
693   k-nearest neighbors using kNNI. The algorithm then calculates the root mean squared error
694   (RMSE) using the imputed value and the actual value of r_{iz}. The best value of k is extracted from
695   the set of k-nearest neighbors that deliver the minimal value of RMSE. The algorithm then Finds
696   the k-nearest neighbors of missing record R_i using kNN algorithm employing the best k value
697   found in the previous step. The iterative imputation method is then used to impute the missing
698   values of the missing record R_i by using the global correlation structure among the selected
699   records. FCKI differs from KI, in that FCKI applies two levels of similarity to achieve a higher
700   imputation accuracy before imputing the missing values through the iterative imputation. For the
701   first level of similarity, fuzzy c-means clustering is selected. The similarities of all records
702   belonging to the cluster are higher than the similarities of all the whole dataset records. Fuzzy c-
703   means algorithm is selected because the records can belong to multiple clusters at the same time.
704   This can lead to further improvement for similarity. For the second level of similarity, kNN is
705   selected. It finds the best k records that are the utmost similar to the missing record by using the
706   Euclidean distance measure. FCKI, similar to KI finds the best k value for the kNN
707   automatically.

708   The results show that FCKI followed by KI performs significantly better than mean,
709   kNNI, SoftImpute, SVDimpute, Iterative Imputation, EMI, DMI, KDMI, KEMI, and KEMI[+].
710   The reported results in Table 4 show that FCKI and KI have the lowest averaged RMSE among
711   the other imputation methods for all the datasets mentioned in Table 3. The average RMSE
712   values in KI and FCKI are better than KEMI[+,] which is the best competitor by 35.42% and
713   40.69%, respectively. In the case of MCAR missing data type, the proposed imputation
714   techniques, KI and FCKI, can impute missing values in a dataset that does not contain any
715   complete record (see Figure 3). They only obtain records that have missing values then impute
716   each incomplete record using the set of records that haven't missed values in corresponding
717   attributes containing missing values in the required incomplete record. This also increases the
718   number of records that can be similar to the required incomplete record that will be used for
719   imputation so that the accuracy of imputation can be improved. Unlike KEMI and KEMI[+], they
720   divide the entire dataset into complete and incomplete records and then impute the missing
721   values using only complete records, so if the dataset does not contain any complete record, the
722   algorithms will stop and will not be able to impute any incomplete record. DMI and KDMI also
723   divide the entire dataset into complete and incomplete records and then build trees from
724   complete records only, so if the dataset does not contain any complete record, the algorithms will
725   not be able to build any tree. They will not be able to impute any incomplete records. The

726 proposed imputation techniques, KI and FCKI, can only handle numerical features, not
727 categorical features.

728   FCKI requires less computational time than other advanced missing data imputation
729 methods; DMI, KDMI, KEMI, KEMI+, KI because FCKI searches a cluster, instead of the
730 whole dataset, to find the best k-nearest neighbors. The average execution time in FCKI is better
731 than DMI by 34.11%, KDMI by 70.46%, KEMI by 86.23%, KEMI$^+$ by 88.6%, and KI by
732 86.72%. DMI and kDMI make utilization of a decision tree for the horizontal partitioning. This
733 is computationally expensive. They build a decision tree for each feature having missing values
734 in the dataset, considering this feature as a class attribute. If there are a large number of features
735 having missing values, DMI and KDMI will build a large number of trees, even if the dataset
736 contains a small number of records.

737   Table 10 shows a comparison between KI and FCKI. The reported results in Table 10
738 show that the average RMSE value overall datasets of FCKI is less than KI because FCKI
739 applies two levels of similarity before imputing the missing values while KI applies only one
740 level of similarity. So FCKI can achieve a higher imputation accuracy. The reported results in
741 Table 10 show also that FCKI has lower average execution time overall datasets than KI.
742 because FCKI searches a cluster, instead of the whole dataset, to find the best k-nearest
743 neighbors while KI is expensive because it searches within the entire dataset to find the most
744 similar records for each missing record.

746- Conclusions and Future Work

746   Two missing data imputation methods are proposed in this work. The first technique,
747 called KI, consolidates k-nearest neighbors and iterative imputation algorithms for imputing the
748 missing data for a dataset. This technique applies only one level of similarity using the kNN
749 algorithm. A suitable k value is estimated automatically for the kNN. The best k-nearest
750 neighbors for each missing record is discovered based on records similarity. The iterative
751 imputation method is then used to impute the missing values of the incomplete records by using
752 the global correlation structure among the selected records. This technique can improve missing
753 data imputation accuracy. However, it is expensive for a large dataset because it is required to
754 search within the entire dataset to find the most similar records for each missing record, so an
755 enhanced hybrid missing data imputation method is proposed, called FCKI, which is an
756 extension of KI. It integrates fuzzy c-means, k-nearest neighbors, and iterative imputation
757 algorithms for imputing the missing data for a dataset. This technique focuses on improving time
758 efficiency for the proposed missing data imputation algorithm as well as missing data imputation
759 accuracy. It uses fuzzy c-means clustering for the dataset to divide records of the dataset into c
760 fuzzy clusters where the records in the same cluster are more similar to each other. Then, it
761 imputes each cluster separately using KI algorithm. FCKI applies two levels of similarity. This
762 technique has the advantage of tackling missing values based on the similarity of the set of
763 records instead of the whole dataset. FCKI can improve time efficiency because It does not have

764 many iterations for imputing missing values in the dataset. It also focuses on both the similarity
765 of data records and the correlation among the features. Therefore, it can enhance imputation
766 efficiency and effectiveness where the most efficient imputation method should impute
767 incomplete dataset with the least amount of time and the most effective imputation method
768 should achieve the highest imputation accuracy.

769   The performance of the proposed imputation methods is assessed by experimenting them
770 with fifteen available datasets through various missing ratios for each type of missing data;
771 MCAR, MAR, MNAR, and, then, compared with ten competitors in terms of three imputation
772 performance measures, which are the root mean square error (RMSE), the normalized root mean
773 square error (NRMSE), and the mean absolute error (MAE). The attained results show that the
774 proposed imputation techniques, KI and FCKI, outperform other competitors in terms of
775 imputation accuracy. The average RMSE values in KI and FCKI are better than KEMI$^+$, which is
776 the best competitor by 35.42% and 40.69%, respectively. The results also show that FCKI
777 requires less computational time than other advanced missing data imputation methods; DMI,
778 KDMI, KEMI, KEMI$^+$, KI. The average execution time in FCKI is better than DMI by 34.11%,
779 KDMI by 70.46%, KEMI by 86.23%, KEMI$^+$ by 88.6%, and KI by 86.72%.

780   The proposed imputation techniques, KI and FCKI, can only handle numerical features.
781 The proposed schemes could also be expanded to use imputation methods that can deal with
782 categorical features and heterogeneous datasets to impute different features, which appear to be
783 useful directions for future research. Future work also includes evaluating the proposed methods
784 on real big datasets to compare the accuracy on varieties of datasets. The behavior of the
785 proposed imputation techniques needs to be evaluated for missing ratios greater than 20% in
786 order to see if this can affect the quality of the missing data imputation. We also plan to explore
787 whether the proposed imputation techniques, KI and FCKI, is useful for data mining tasks such
788 as classification. In order to reduce the computational time, the imputation process for FCKI
789 algorithm can be parallelized and we will also try to replace kNN with another method such as
790 support vector machine (SVM) or random forests (RFs) to reduce the computational time for
791 imputing the missing values in a large-scale dataset while maintaining the high imputation
792 accuracy.

793 ## References

794 Andridge RR, Little RJA. 2010. A Review of Hot Deck Imputation for Survey Non-response.
795   *International Statistical Review* 78:40–64. DOI: 10.1111/j.1751-5823.2010.00103.x.

796 Azimi I, Pahikkala T, Rahmani AM, Niela-Vilén H, Axelin A, Liljeberg P. 2019. Missing data
797   resilient decision-making for healthcare IoT through personalization: A case study on
798   maternal health. *Future Generation Computer Systems* 96:297–308. DOI:
799   10.1016/j.future.2019.02.015.

800 Batista GEAPA, Monard MC. 2003. An analysis of four missing data treatment methods for
801   supervised learning. *Applied Artificial Intelligence* 17:519–533. DOI: 10.1080/713827181.

802 Bethlehem J. 2009. *Applied Survey Methods: A Statistical Perspective*. Hoboken, NJ, USA: John
803   Wiley & Sons, Inc. DOI: 10.1002/9780470494998.

804 Bezdek JC, Ehrlich R, Full W. 1984. FCM: The fuzzy c-means clustering algorithm. *Computers*
805   *& Geosciences* 10:191–203. DOI: 10.1016/0098-3004(84)90020-7.

806 Bilogur A. 2018. Missingno: a missing data visualization suite. *The Journal of Open Source*
807   *Software* 3:547. DOI: 10.21105/joss.00547.

808 Buuren S van, Groothuis-Oudshoorn K. 2011. mice: Multivariate Imputation by Chained
809   Equations in R. *Journal of Statistical Software* 45:1–67. DOI: 10.18637/jss.v045.i03.

810 Chapman AD, Speers L. 2005. *Principles of data quality, version 1.0.*

811 Cheng KO, Law NF, Siu WC. 2012. Iterative bicluster-based least square framework for
812   estimation of missing values in microarray gene expression data. *Pattern Recognition*
813   45:1281–1289. DOI: 10.1016/j.patcog.2011.10.012.

814 Deb R, Liew AW-C. 2016. Missing value imputation for the analysis of incomplete traffic
815   accident data. *Information Sciences* 339:274–289. DOI: 10.1016/j.ins.2016.01.018.

816 Feng X, Wu S, Srivastava J, Desikan P. 2015. Automatic instance selection via locality
817   constrained sparse representation for missing value estimation. *Knowledge-Based Systems*
818   85:210–223. DOI: 10.1016/j.knosys.2015.05.007.

819 Fouad KM, Elbably D. 2020. Intelligent approach for large-scale data mining. *International*
820   *Journal of Computer Applications in Technology* 63:93–113. DOI:
821   10.1504/IJCAT.2020.107906.

822 Fouad KM, El Shishtawy T, Altae AA. 2018. Enhanced Learning Approach for Diseases
823   Diagnostic. *Asian Journal of Information Technology* 17:202–211. DOI:
824   10.3923/ajit.2018.202.211.

825 Ganti V, Gehrke J, Ramakrishnan R. 1999. Mining very large databases. *Computer* 32:38–45.
826   DOI: 10.1109/2.781633.

827 García S, Luengo J, Herrera F. 2015. Data Preprocessing in Data Mining. *Intelligent Systems*
828   *Reference Library* 72.

829 Garciarena U, Santana R. 2017. An extensive analysis of the interaction between missing data
830   types, imputation methods, and supervised classifiers. *Expert Systems with Applications*
831   89:52–65. DOI: 10.1016/j.eswa.2017.07.026.

832 Gardner J, Brooks C. 2017. A Statistical Framework for Predictive Model Evaluation in
833   MOOCs. In: *Proceedings of the Fourth (2017) ACM Conference on Learning @ Scale*.
834   New York. ACM, 269–272. DOI: 10.1145/3051457.3054002.

835 Han J, Kamber M. 2013. *Data Mining: Concepts and Techniques*. Morgan Kaufmann.

836 Houari R, Bounceur A, Tari AK, Kecha MT. 2014. Handling Missing Data Problems with
837   Sampling Methods. In: *2014 International Conference on Advanced Networking Distributed*
838   *Systems and Applications*. IEEE, 99–104. DOI: 10.1109/INDS.2014.25.

839   Hu M, Salvucci S. 2001. *A Study of Imputation Algorithms*.

840   Jiang C, Yang Z. 2015. CKNNI: An Improved KNN-Based Missing Value Handling Technique.
841   In: *International Conference on Intelligent Computing (ICIC 2015)*. Cham. Springer, 441–
842   452. DOI: 10.1007/978-3-319-22053-6_47.

843   Junninen H, Niska H, Tuppurainen K, Ruuskanen J, Kolehmainen M. 2004. Methods for
844   imputation of missing values in air quality data sets. *Atmospheric Environment* 38:2895–
845   2907. DOI: 10.1016/j.atmosenv.2004.02.026.

846   Kodinariya TM, Makwana PR. 2013. Review on determining number of Cluster in K-Means
847   Clustering. *International Journal of Advance Research in Computer Science and
848   Management Studies* 1:2321–7782.

849   Kwok T, Smith K, Lozano S, Taniar D. 2002. Parallel Fuzzy c- Means Clustering for Large Data
850   Sets. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial
851   Intelligence and Lecture Notes in Bioinformatics)*. 365–374. DOI: 10.1007/3-540-45706-
852   2_48.

853   Li D, Deogun J, Spaulding W, Shuart B. 2004. Towards Missing Data Imputation: A Study of
854   Fuzzy K-means Clustering Method. In: *International Conference on Rough Sets and
855   Current Trends in Computing (RSCTC 2004)*. Berlin, Heidelberg: Springer, 573–579. DOI:
856   10.1007/978-3-540-25929-9_70.

857   Li L, Du B, Wang Y, Qin L, Tan H. 2020. Estimation of missing values in heterogeneous traffic
858   data: Application of multimodal deep learning model. *Knowledge-Based Systems*
859   194:105592. DOI: 10.1016/j.knosys.2020.105592.

860   Little RJA, Rubin DB. 2002. *Statistical Analysis with Missing Data*. Hoboken, NJ, USA: John
861   Wiley & Sons, Inc. DOI: 10.1002/9781119013563.

862   Liu Z, Liu Y, Dezert J, Pan Q. 2015. Classification of incomplete data based on belief functions
863   and K-nearest neighbors. *Knowledge-Based Systems* 89:113–125. DOI:
864   10.1016/j.knosys.2015.06.022.

865   Mazumder R, Hastie T, Tibshirani R. 2010. Spectral regularization algorithms for learning large
866   incomplete matrices. *Journal of Machine Learning Research* 11:2287−2322.

867   Müller H, Naumann F, Freytag J-C. 2003. Data quality in genome databases. In: *Eighth
868   International Conference on Information Quality (IQ 2003)*. 269–284.

869   Patil BM, Joshi RC, Toshniwal D. 2010. Missing Value Imputation Based on K-Mean Clustering
870   with Weighted Distance. In: *Communications in Computer and Information Science*. 600–
871   609. DOI: 10.1007/978-3-642-14834-7_56.

872   Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Müller A,
873   Nothman J, Louppe G, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A,
874   Cournapeau D, Brucher M, Perrot M, Duchesnay É. 2011. Scikit-learn: Machine Learning
875   in Python. *Journal of Machine Learning Research* 12:2825–2830.

876   Pinzon-Morales R-D, Baquero-Duarte K-A, Orozco-Gutierrez A-A, Grisales-Palacio V-H. 2011.
877   Pattern Recognition of Surface EMG Biological Signals by Means of Hilbert Spectrum and

878   Fuzzy Clustering. In: Arabnia H, Tran Q eds. *Advances in Experimental Medicine and*
879   *Biology*. New York, NY: Springer, 201–209. DOI: 10.1007/978-1-4419-7046-6_20.

880   Poolsawad N, Moore L, Kambhampati C, Cleland JGF. 2012. Handling missing values in data
881   mining - A case study of heart failure dataset. In: *Proceedings - 2012 9th International*
882   *Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2012*. DOI:
883   10.1109/FSKD.2012.6233860.

884   Rahman MG, Islam MZ. 2011. A decision tree-based missing value imputation technique for
885   data preprocessing. In: *The 9th Australasian Data Mining Conference (AusDM 11)*.
886   Ballarat, Australia, 41–50.

887   Rahman MG, Islam MZ. 2013a. Data Quality Improvement by Imputation of Missing Values. In:
888   *International Conference on Computer Science and Information Technology (CSIT-2013)*.
889   Yogyakarta, Indonesia, 82–88.

890   Rahman MG, Islam MZ. 2013b. kDMI: A Novel Method for Missing Values Imputation Using
891   Two Levels of Horizontal Partitioning in a Data set. In: *International Conference on*
892   *Advanced Data Mining and Applications (ADMA 2013)*. Hangzhou, China, 250–263. DOI:
893   10.1007/978-3-642-53917-6_23.

894   Rahman MG, Islam MZ. 2014. FIMUS: A framework for imputing missing values using co-
895   appearance, correlation and similarity analysis. *Knowledge-Based Systems*. DOI:
896   10.1016/j.knosys.2013.12.005.

897   Rahman MG, Islam MZ. 2016. Missing value imputation using a fuzzy clustering-based EM
898   approach. *Knowledge and Information Systems* 46:389–422. DOI: 10.1007/s10115-015-
899   0822-y.

900   Ravi V, Krishna M. 2014. A new online data imputation method based on general regression
901   auto associative neural network. *Neurocomputing* 138:106–113. DOI:
902   10.1016/j.neucom.2014.02.037.

903   Razavi-Far R, Cheng B, Saif M, Ahmadi M. 2020. Similarity-learning information-fusion
904   schemes for missing data imputation. *Knowledge-Based Systems* 187:104805. DOI:
905   10.1016/j.knosys.2019.06.013.

906   Razavi-Far R, Zio E, Palade V. 2014. Efficient residuals pre-processing for diagnosing multi-
907   class faults in a doubly fed induction generator, under missing data scenarios. *Expert*
908   *Systems with Applications* 41:6386–6399. DOI: 10.1016/j.eswa.2014.03.056.

909   Salgado CM, Azevedo C, Proença H, Vieira SM. 2016. Missing Data. In: *Secondary Analysis of*
910   *Electronic Health Records*. Cham: Springer International Publishing, 143–162. DOI:
911   10.1007/978-3-319-43742-2_13.

912   Salleh MNM, Samat NA. 2017. FCMPSO: An Imputation for Missing Data Features in Heart
913   Disease Classification. In: *IOP Conference Series: Materials Science and Engineering*.
914   Melaka, Malaysia, 012102. DOI: 10.1088/1757-899X/226/1/012102.

915   Santos MS, Abreu PH, Wilk S, Santos J. 2020. How distance metrics influence missing data
916   imputation with k-nearest neighbours. *Pattern Recognition Letters* 136:111–119. DOI:
917   10.1016/j.patrec.2020.05.032.

918   Schneider T. 2001. Analysis of Incomplete Climate Data: Estimation of Mean Values and
919   Covariance Matrices and Imputation of Missing Values. *Journal of Climate* 14:853–871.
920   DOI: 10.1175/1520-0442(2001)014<0853:AOICDE>2.0.CO;2.

921   Sefidian AM, Daneshpour N. 2019. Missing value imputation using a novel grey based fuzzy c-
922   means, mutual information based feature selection, and regression model. *Expert Systems*
923   *with Applications* 115:68–94. DOI: 10.1016/j.eswa.2018.07.057.

924   Sefidian AM, Daneshpour N. 2020. Estimating missing data using novel correlation
925   maximization based methods. *Applied Soft Computing* 91:106249. DOI:
926   10.1016/j.asoc.2020.106249.

927   Soley-bori M. 2013. Dealing with missing data: Key assumptions and methods for applied
928   analysis. *PM931 Directed Study in Health Policy and Management*.

929   Sree Dhevi AT. 2014. Imputing missing values using Inverse Distance Weighted Interpolation
930   for time series data. In: *2014 Sixth International Conference on Advanced Computing*
931   *(ICoAC)*. IEEE, 255–259. DOI: 10.1109/ICoAC.2014.7229721.

932   Troyanskaya O, Cantor M, Sherlock G, Brown P, Hastie T, Tibshirani R, Botstein D, Altman
933   RB. 2001. Missing value estimation methods for DNA microarrays. *Bioinformatics* 17:520–
934   525. DOI: 10.1093/bioinformatics/17.6.520.

935   Young W, Weckman G, Holland W. 2011. A survey of methodologies for the treatment of
936   missing values within datasets: limitations and benefits. *Theoretical Issues in Ergonomics*
937   *Science* 12:15–43. DOI: 10.1080/14639220903470205.

938   Zhang C, Qin Y, Zhu X, Zhang J, Zhang S. 2006. Clustering-based Missing Value Imputation
939   for Data Preprocessing. In: *2006 IEEE International Conference on Industrial Informatics*.
940   Singapore: IEEE, 1081–1086. DOI: 10.1109/INDIN.2006.275767.

941

Figure 1

The proposed KI scheme.

D is the incomplete dataset, D´ is the imputed dataset of D, D_{mis} is the incomplete subset of D, R_i is the incomplete target record from D_{mis}, $R_i´$ is the imputed record of R_i, A_{mis} is the set of attributes that contain missing values in R_i, P is the pool, Temp is a copy of the pool P, $r_{iz}´$ is the imputed value of r_{iz}, N_p is the number of records in P, d_k is k nearest neighbors of R_i, S is a subset made up of d_k and R_i

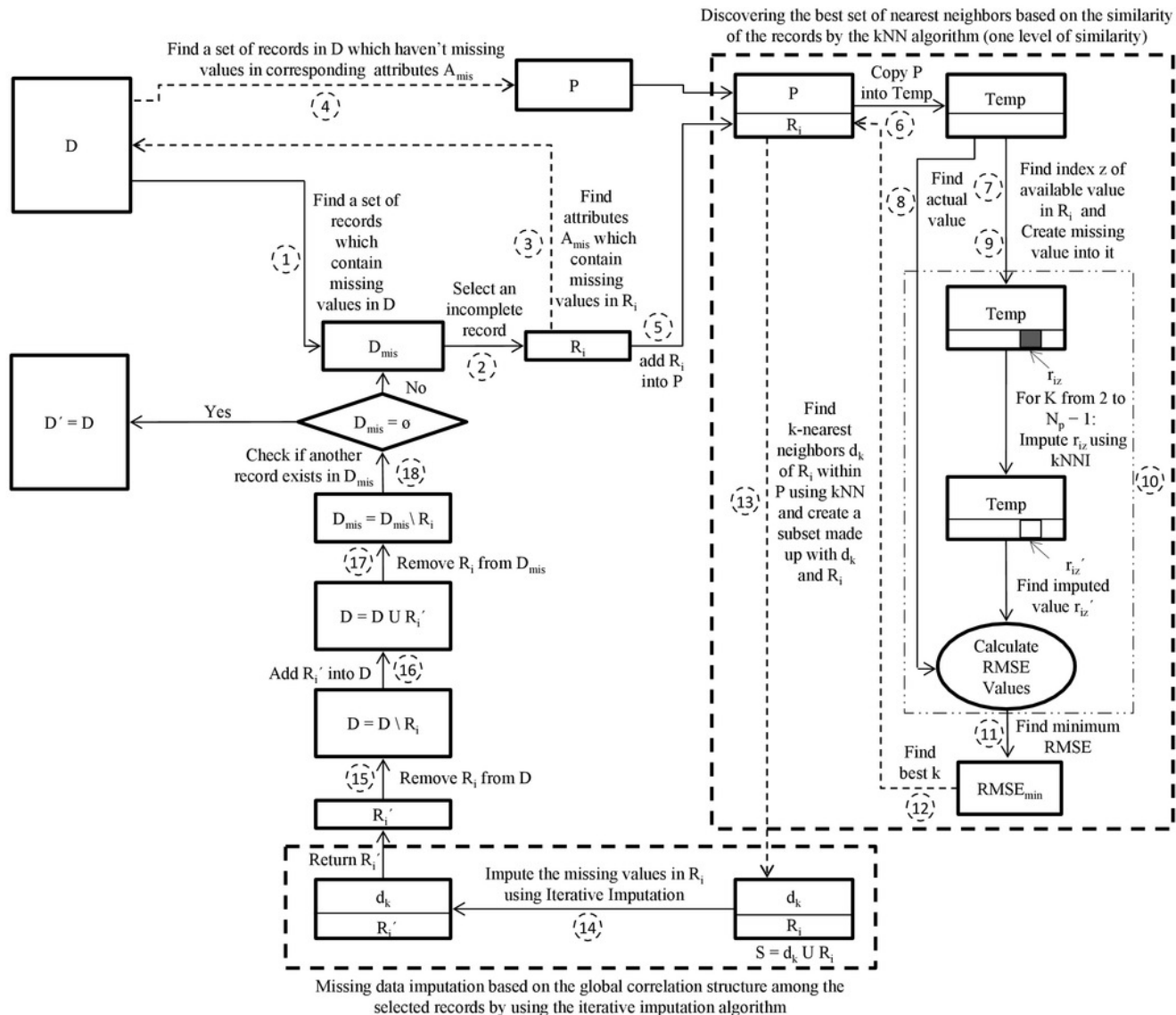



Figure 2

The proposed FCKI scheme.

D is the Incomplete dataset, D´ is the Imputed dataset of D, c is the optimal number of clusters in dataset D, Clusters is the fuzzy clusters that partitioned using fuzzy c-means algorithm, C_i is the target cluster from Clusters, $C_i´$ is the Imputed cluster of C_i

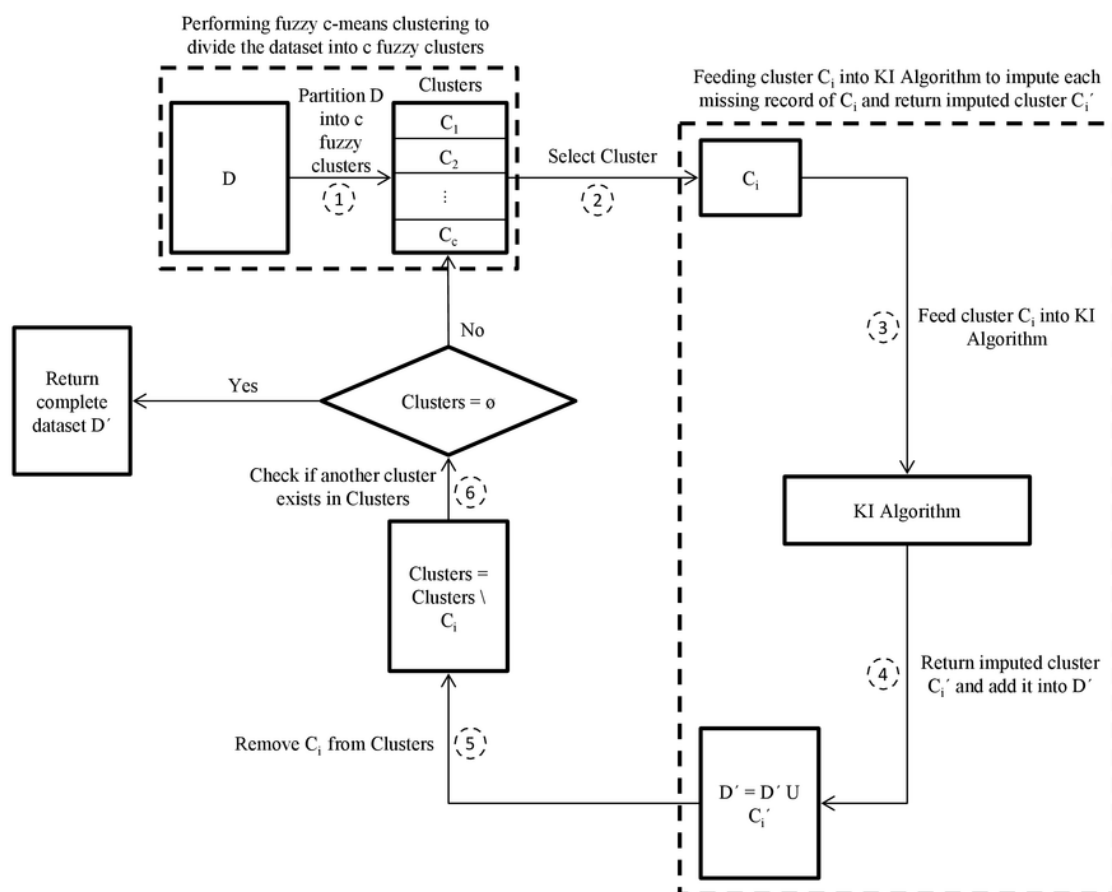



Figure 3

The visualization of the sonar dataset with MCAR missing data type and 10% missing ratio.

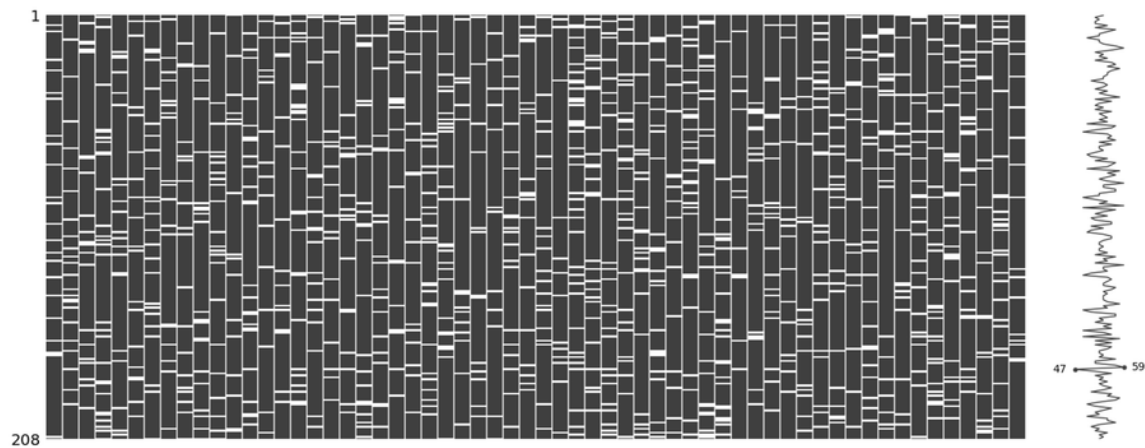



Figure 4

The number of missing values in each feature for an incomplete Sonar dataset with MCAR missing data type and 10% missing ratio.

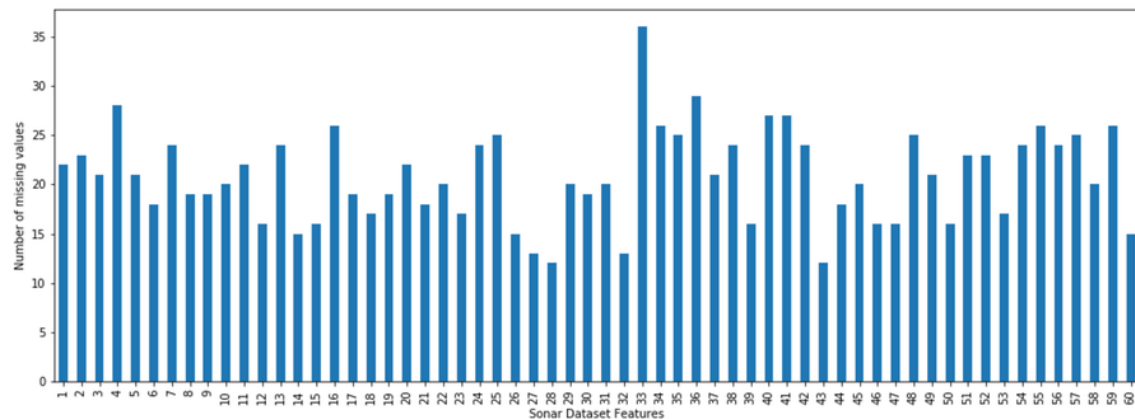



Figure 5

The visualization of sonar dataset with MAR missing data type and 10% missing ratio.

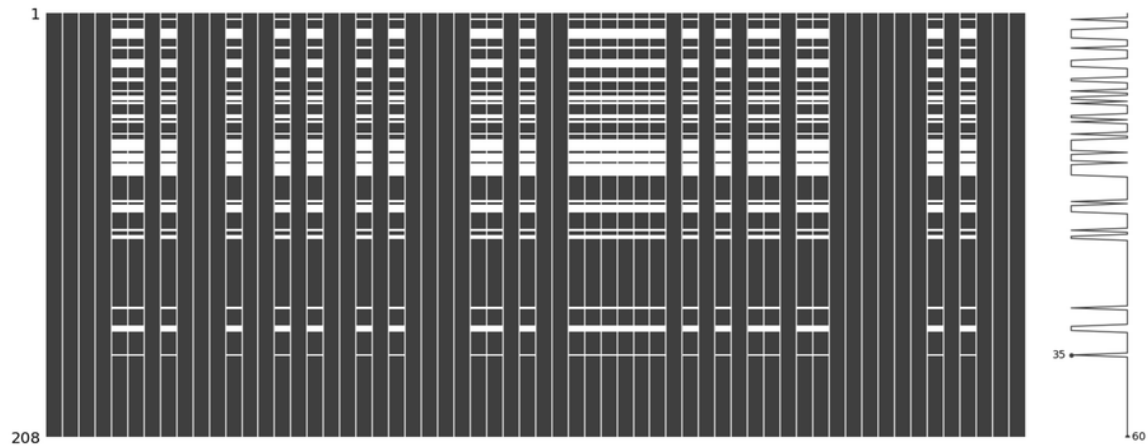



Figure 6

The number of missing values in each feature for an incomplete Sonar dataset with MAR missing data type and 10% missing ratio.

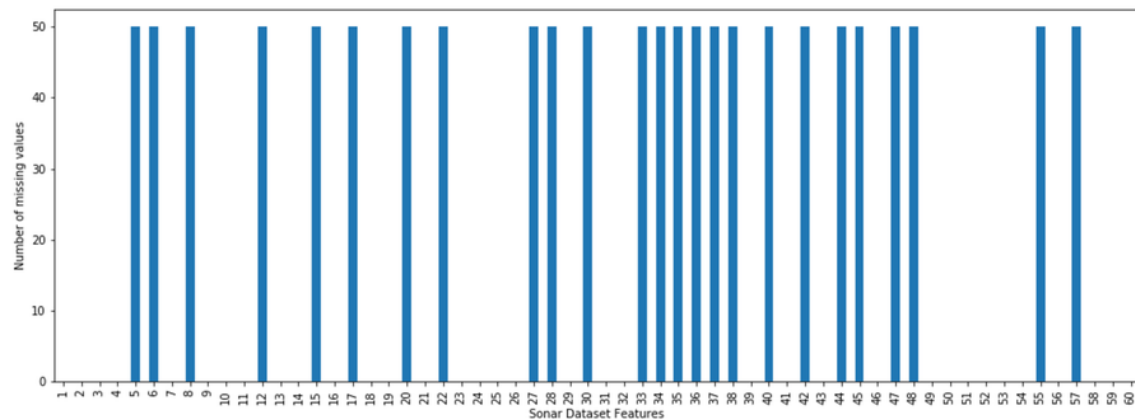



Figure 7

The visualization of sonar dataset with MNAR missing data type and 10% missing ratio.

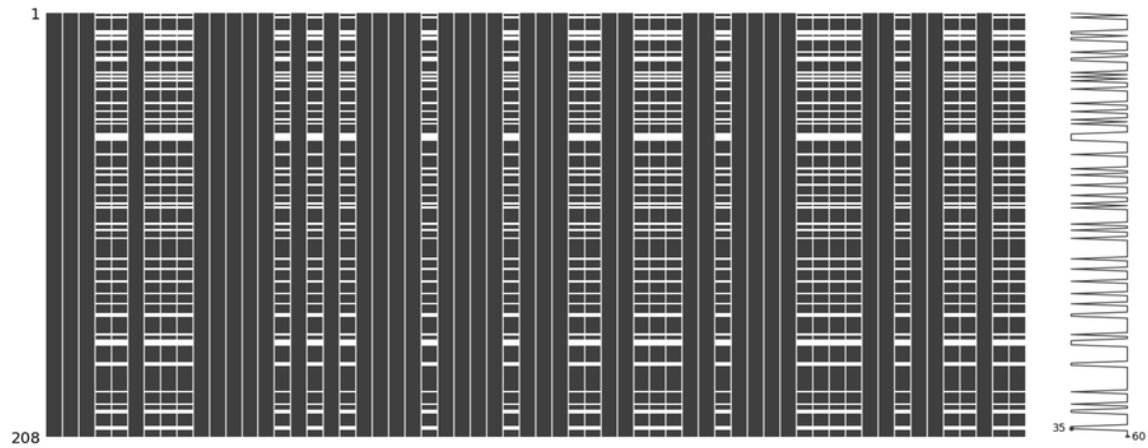



Figure 8

The number of missing values in each feature for an incomplete Sonar dataset with MNAR missing data type and 10% missing ratio.

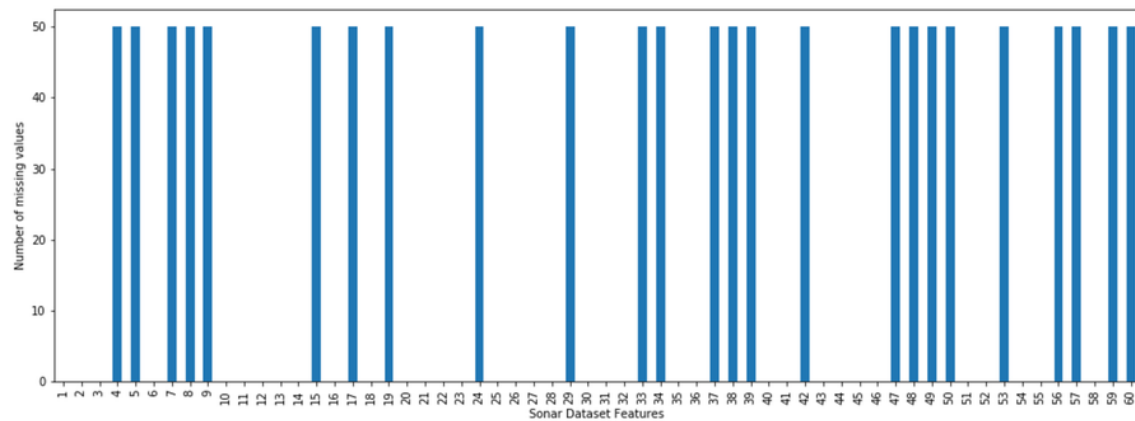



Figure 9

Distribution of the RMSE values obtained by each method.

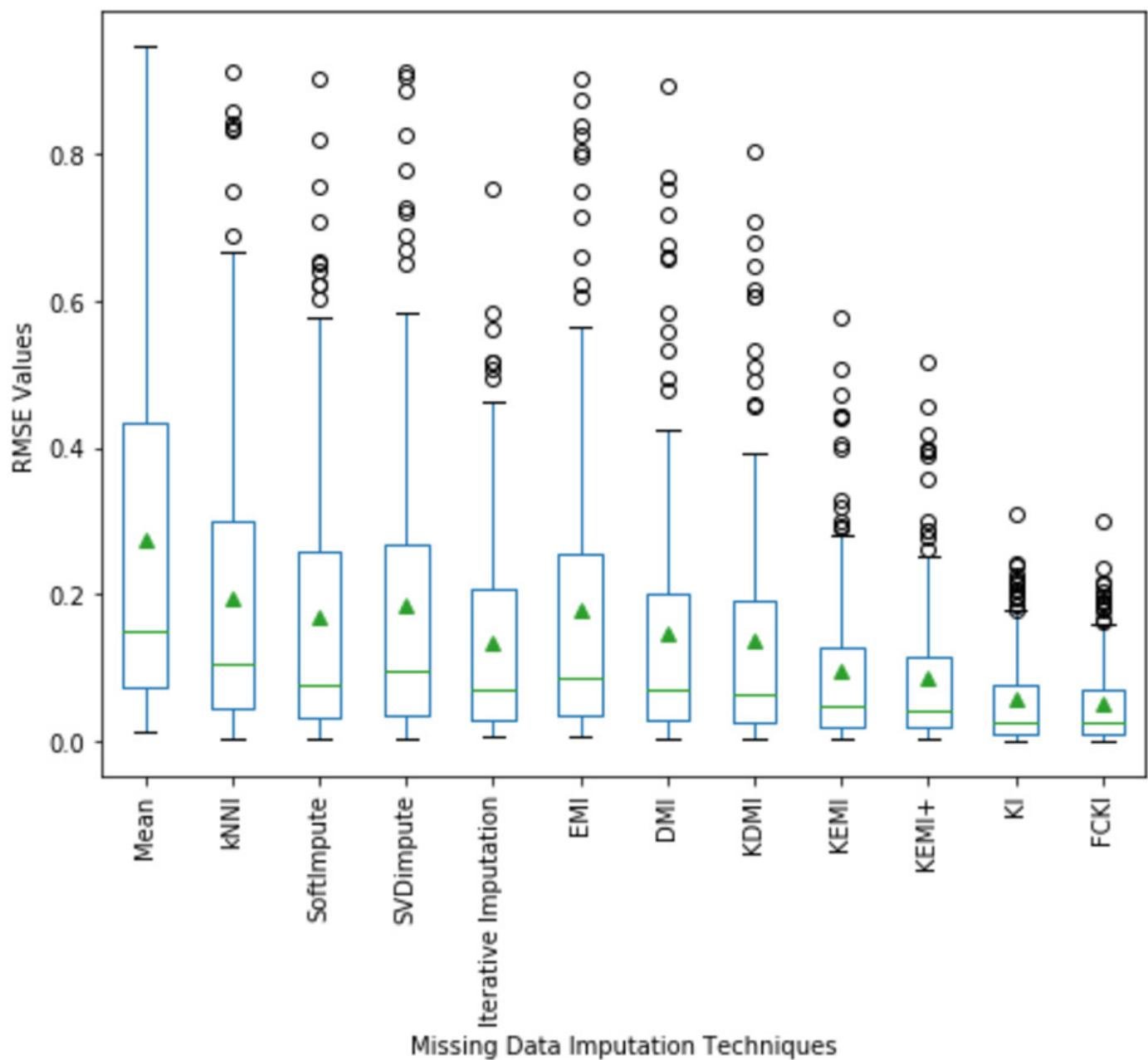



Figure 10

The average value of RMSE values and their distribution are obtained by each method.

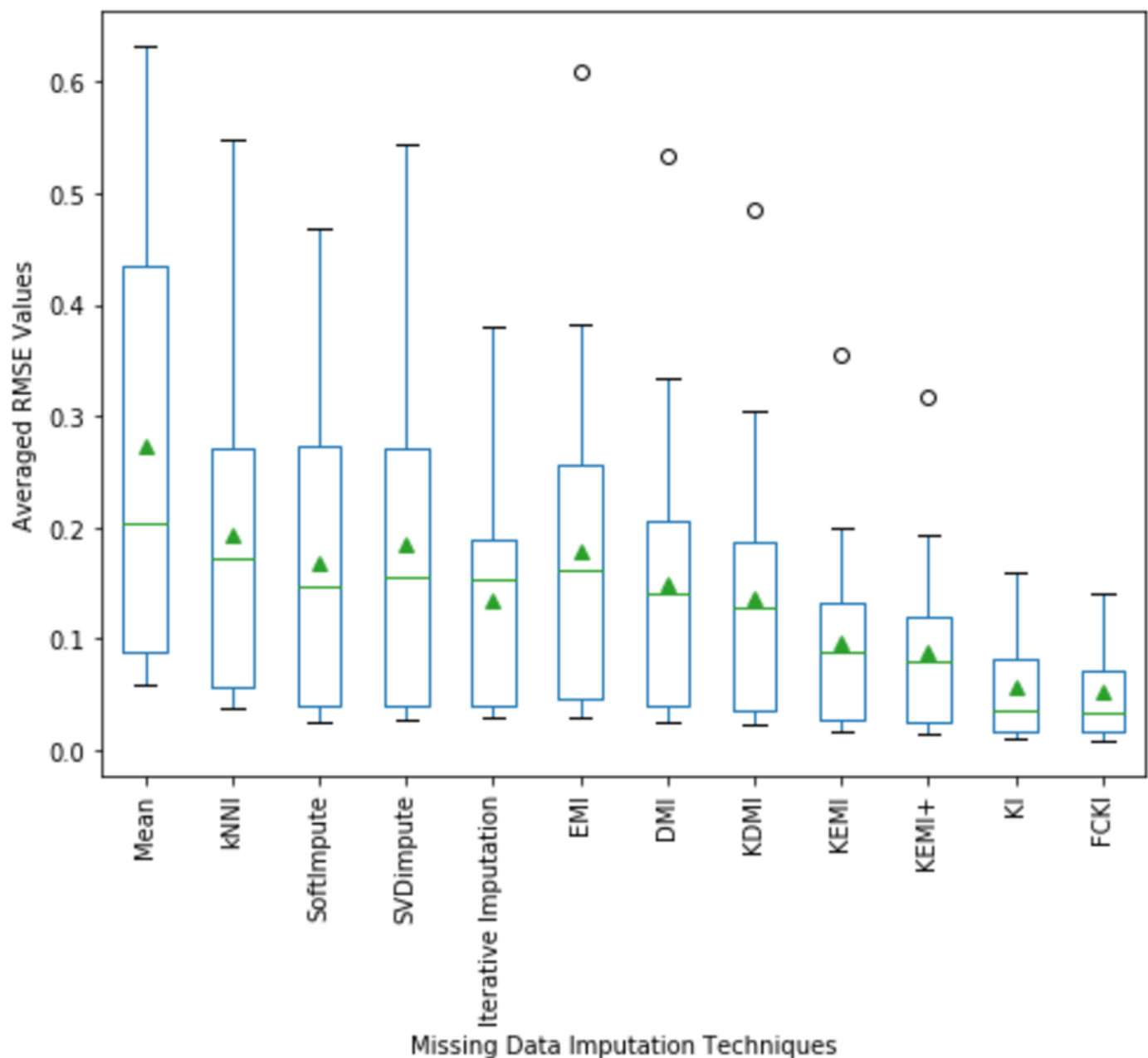



Figure 11

The average value of RMSE values for MCAR missing data type overall datasets achieved by applying each imputation method to each missing ratio.

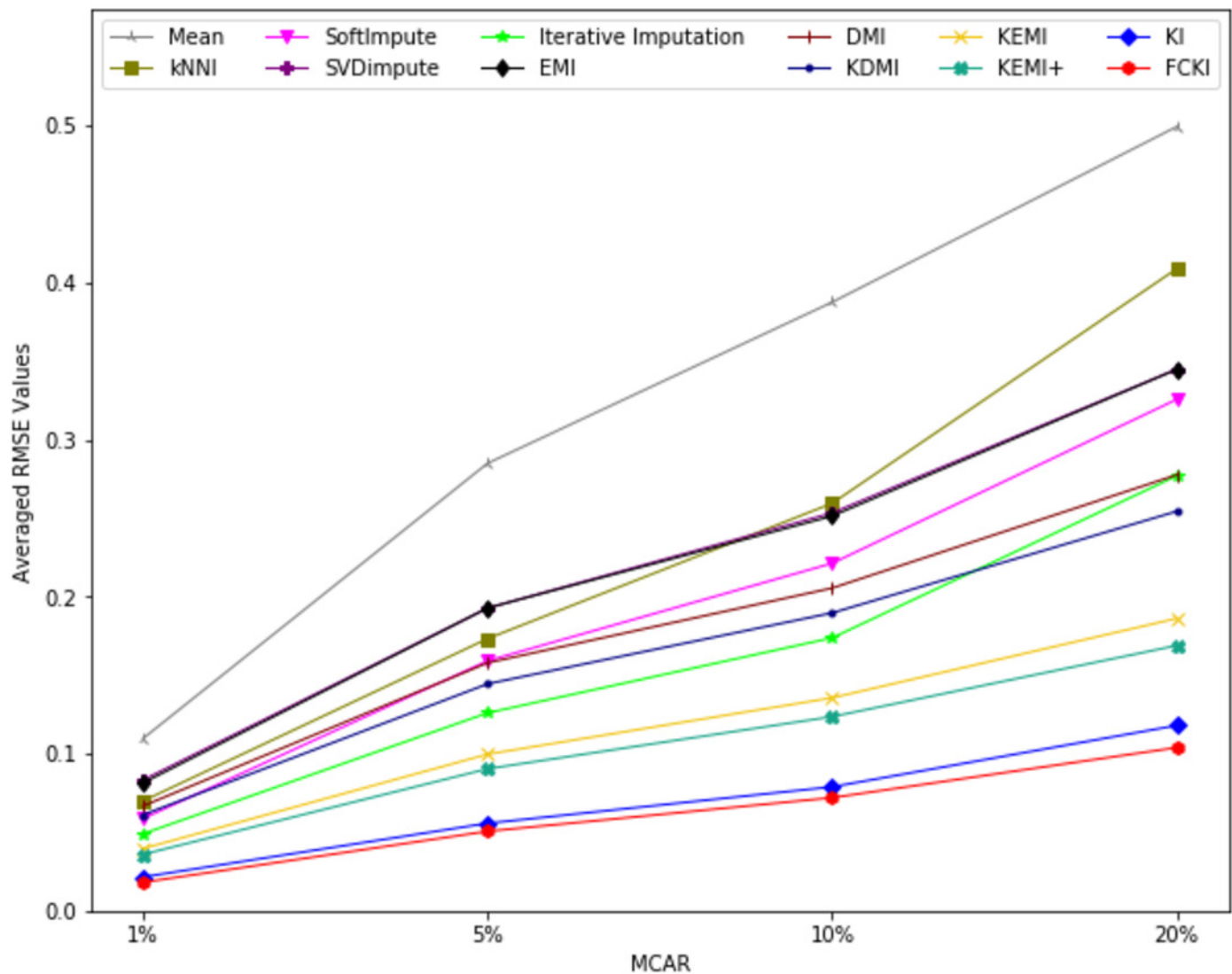



Figure 12

The average value of RMSE values for MAR missing data type for datasets achieved by applying each imputation method to each missing ratio.

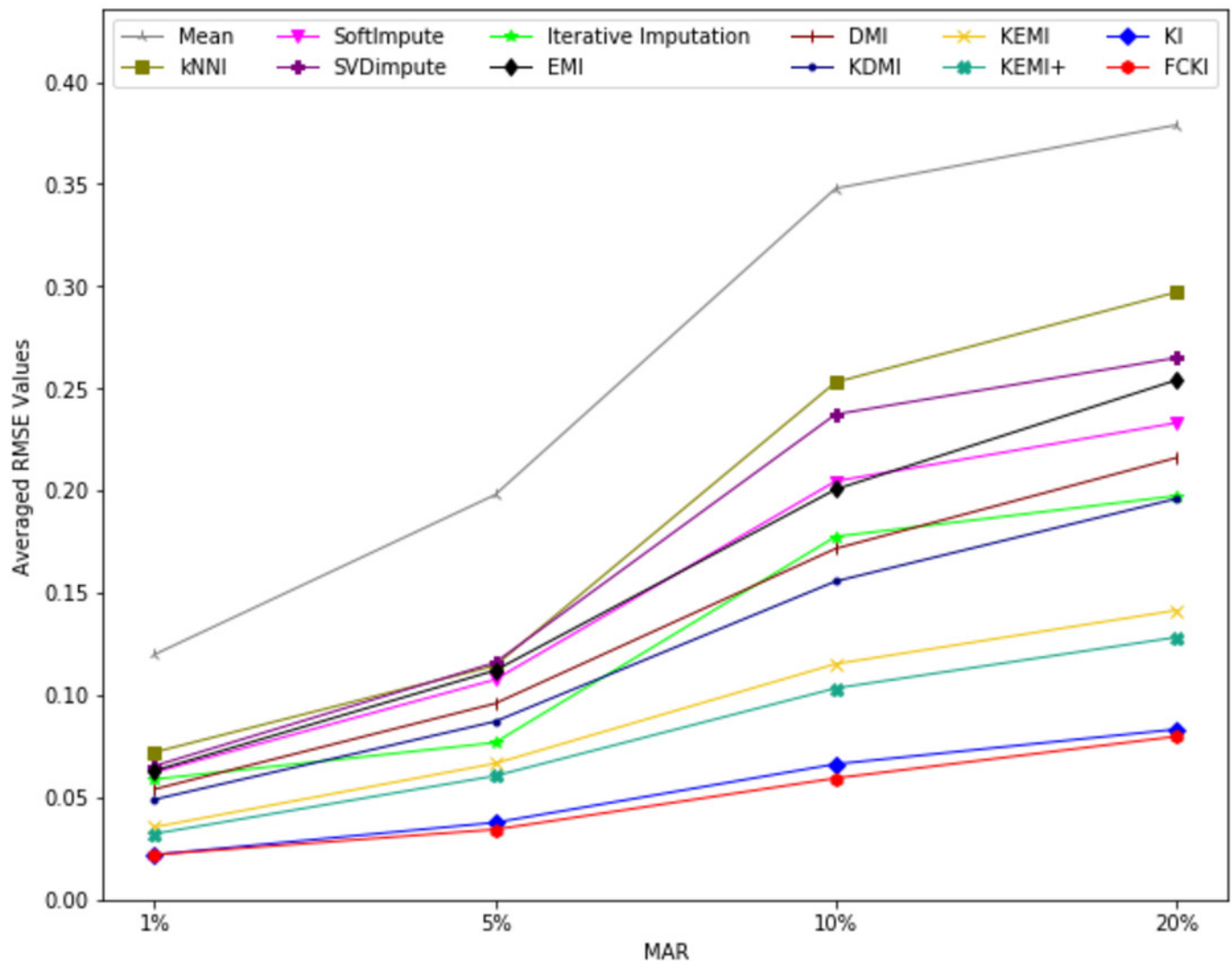



Figure 13

The average value of RMSE values for MNAR missing data type for datasets achieved by applying each imputation method to each missing ratio.

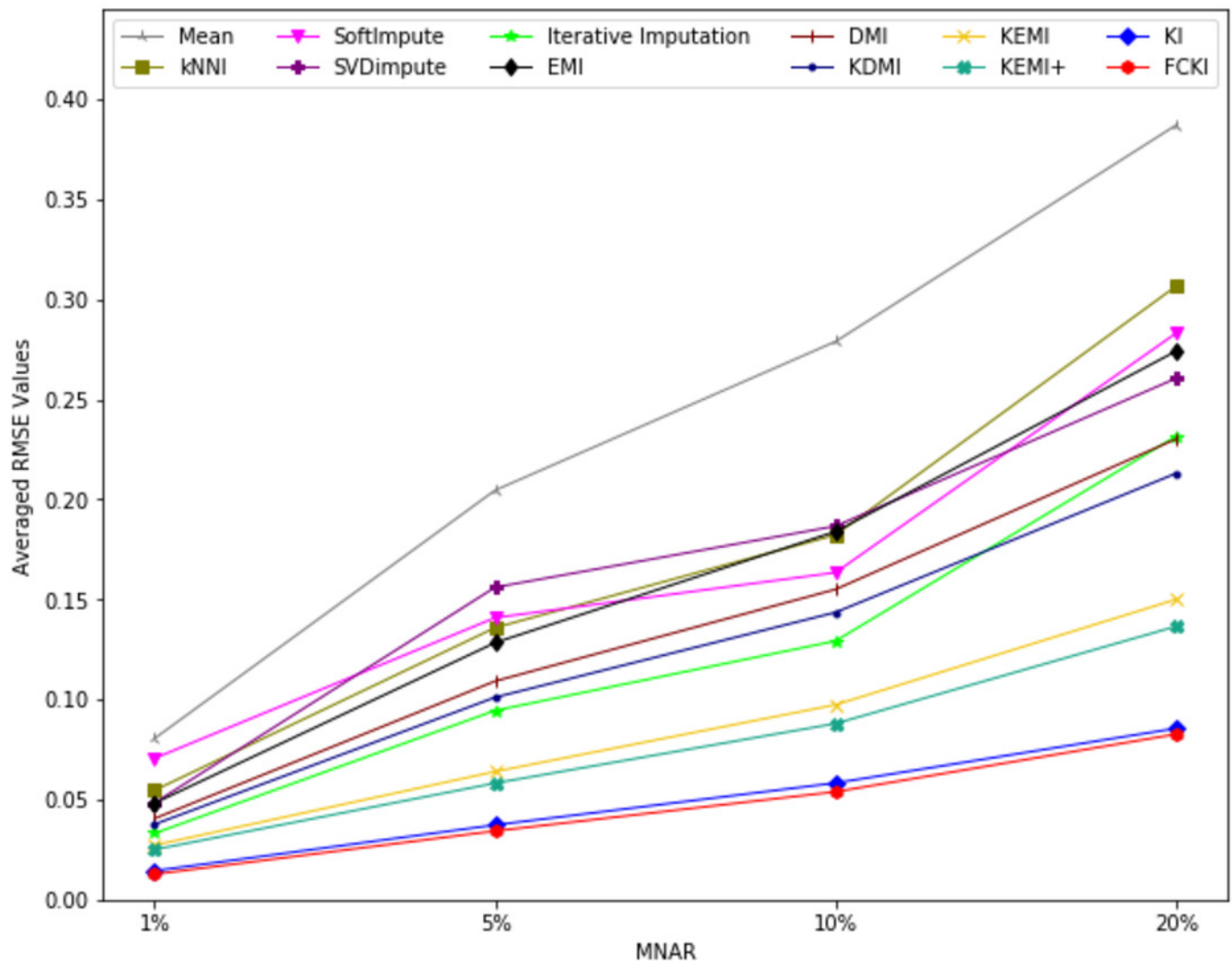



Figure 14

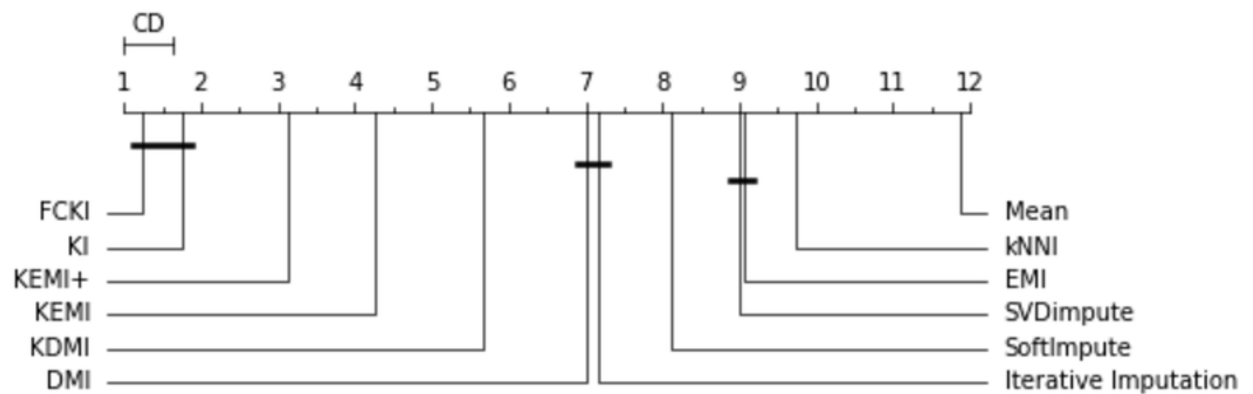
The critical difference for the Nemenyi test.

The definitions of notations used in Algorithm 4.

Table 1: The definitions of notations used in Algorithm 4.

Notation	Definition
D	Incomplete dataset
D′	Imputed dataset of D
N	Total number of records in D
M	Total number of attributes in D
D_{mis}	Incomplete subset of D
R_i	Incomplete target record from D_{mis}
R_i'	Imputed record of R_i
A_{mis}	Set of attributes that contain missing values in R_i
A_j	Incomplete attribute within R_i
P	Pool that contains a set of records in D which haven't missing values in corresponding attributes A_{mis}
Temp	Copy of the pool P
r_{iz}'	Imputed value of r_{iz}
N_p	Number of records in P
d_k	k nearest neighbors of R_i
S	Subset made up of d_k and R_i

The definitions of notations used in Algorithm 5.

2 **Table 2:** The definitions of notations used in Algorithm 5.

Notation	Definition
D	Incomplete dataset
D´	Imputed dataset of D
N	Total number of records in D
M	Total number of attributes in D
c	Optimal number of clusters in dataset D
Clusters	Fuzzy clusters that partitioned using fuzzy c-means algorithm
C_i	Target cluster from Clusters
$C_i´$	Imputed cluster of C_i

The datasets used in these experiments.

2 **Table 3:** The datasets used in these experiments.

Datasets	Number of records	Number of numerical features	Dataset size (KB)
Zoo	101	17	3.49
Iris	150	4	2.36
Sonar	208	60	83.8
Glass	214	9	9.80
Ecoli	336	7	11.2
Leaf	340	16	40.4
Ionosphere	351	34	73.3
Movement libras	360	90	250
QSAR fish toxicity	908	7	30.3
Yeast	1484	8	52.8
Abalone	4177	8	183
Anuran Calls (MFCCs)	7195	22	1780
Letter	20000	16	676
Sensorless Drive Diagnosis	58509	48	23800
Pseudo Periodic Synthetic	100000	10	11500

The average RMSE values over twelve experiments were obtained for each dataset.

Table 4: The average RMSE values over twelve experiments were obtained for each dataset.

Datasets	Mean	kNNI	SoftImpute	SVDimpute	Iterative Imputation	EMI	DMI	KDMI	KEMI	KEMI+	KI	FCKI
Zoo	0.4457	0.2483	0.2310	0.2575	0.1837	0.2219	0.1935	0.1766	0.1077	0.0965	**0.0757**	**0.0632**
Iris	0.5946	0.2878	0.3338	0.4114	0.1866	0.3818	0.3333	0.3044	0.1525	0.1369	**0.0903**	**0.0745**
Sonar	0.0916	0.0571	0.0289	0.0267	0.0289	0.0471	0.0410	0.0375	0.0302	0.0271	**0.0177**	**0.0194**
Glass	0.4818	0.3482	0.3694	0.2660	0.2314	0.2906	0.2173	0.1982	0.1466	0.1314	**0.1076**	**0.1120**
Ecoli	0.0797	0.0694	0.0550	0.0449	0.0455	0.0419	0.0363	0.0332	0.0277	0.0248	**0.0208**	**0.0223**
Leaf	0.2031	0.1717	0.1625	0.1660	0.1544	0.1775	0.1509	0.1358	0.1181	0.1074	**0.0642**	**0.0546**
Ionosphere	0.3264	0.2392	0.1464	0.1558	0.1937	0.1613	0.1404	0.1282	0.0984	0.0884	**0.0701**	**0.0683**
Movement libras	0.1012	0.0444	0.0244	0.0624	0.0479	0.0520	0.0452	0.0413	0.0289	0.0259	**0.0131**	**0.0123**
QSAR fish toxicity	0.4263	0.3573	0.3091	0.3141	0.2530	0.2923	0.2244	0.2159	0.2003	0.1926	**0.1354**	**0.1258**
Yeast	0.0655	0.0580	0.0392	0.0308	0.0353	0.0288	0.0250	0.0228	0.0225	0.0201	**0.0137**	**0.0137**
Abalone	0.4009	0.2534	0.2370	0.2751	0.1677	0.2017	0.1756	0.1606	0.0882	0.0791	**0.0353**	**0.0329**
Anuran Calls (MFCCs)	0.0868	0.0373	0.0382	0.0359	0.0293	0.0437	0.0380	0.0347	0.0252	0.0226	**0.0151**	**0.0133**
Letter	0.6320	0.5476	0.4684	0.5444	0.3794	0.6101	0.5328	0.4864	0.3546	0.3180	**0.1606**	**0.1416**
Sensorless Drive Diagnosis	0.1019	0.0574	0.0420	0.1366	0.0643	0.0885	0.0524	0.0478	0.0346	0.0310	**0.0208**	**0.0183**
Pseudo Periodic Synthetic	0.0597	0.1327	0.0534	0.0348	0.0305	0.0302	0.0263	0.0240	0.0175	0.0157	**0.0105**	**0.0092**
Standard Deviation	0.2009	0.1450	0.1420	0.1525	0.1022	0.1588	0.1370	0.1255	0.0887	0.0807	**0.0472**	**0.0430**

The average value of RMSE values for all datasets achieved by applying each imputation method to each missing data type.

Table 5: The average value of RMSE values for all datasets achieved by applying each imputation method to
each missing data type.

Datasets	Mean	kNNI	SoftImpute	SVDimpute	Iterative Imputation	EMI	DMI	KDMI	KEMI	KEMI$^+$	KI	FCKI
MCAR	0.3204	0.2279	0.1912	0.2186	0.1565	0.2176	0.1769	0.1625	0.1153	0.1047	**0.0686**	**0.0612**
MAR	0.2612	0.1842	0.1519	0.1709	0.1277	0.1575	0.1358	0.1232	0.0906	0.0819	**0.0527**	**0.0492**
MNAR	0.2378	0.1699	0.1646	0.1629	0.1222	0.1587	0.1338	0.1239	0.0847	0.0770	**0.0489**	**0.0459**

The average value of RMSE values for all datasets achieved by applying each imputation method to each missing ratio.

Table 6: The average value of RMSE values for all datasets achieved by applying each imputation method to
each missing ratio.

Datasets	Mean	kNNI	SoftImpute	SVDimpute	Iterative Imputation	EMI	DMI	KDMI	KEMI	KEMI⁺	KI	FCKI
1%	0.1033	0.0655	0.0639	0.0657	0.0470	0.0643	0.0538	0.0492	0.0341	0.0310	**0.0193**	**0.0176**
5%	0.2292	0.1412	0.1359	0.1549	0.0991	0.1444	0.1210	0.1109	0.0768	0.0697	**0.0436**	**0.0398**
10%	0.3382	0.2317	0.1965	0.2259	0.1602	0.2121	0.1774	0.1630	0.1161	0.1050	**0.0678**	**0.0618**
20%	0.4219	0.3375	0.2807	0.2902	0.2354	0.2910	0.2431	0.2228	0.1605	0.1457	**0.0962**	**0.0892**

The average execution times (in seconds) over twelve experiments achieved by using each dataset of the fifteen datasets for advanced methods.

Table 7: The average execution times (in seconds) over twelve experiments achieved by using each dataset of the fifteen datasets for advanced methods.

Datasets	DMI	KDMI	KEMI	KEMI⁺	KI	FCKI
Zoo	6.9302	15.4543	7.4657	20.4393	10.3779	**4.3305**
Iris	5.9100	13.1793	11.9682	30.2816	13.7762	3.8477
Sonar	83.5382	186.2902	65.1894	193.8505	169.6246	53.4663
Glass	34.8523	77.7207	52.9551	145.4900	70.0241	23.1727
Ecoli	90.0214	200.7477	125.9442	280.3394	153.6133	59.6401
Leaf	46.0662	102.7276	139.3200	348.8740	210.7502	30.1334
Ionosphere	160.7483	358.4687	237.8846	583.1687	291.7119	103.0051
Movement libras	510.6320	1138.7093	384.2715	1174.0173	861.1263	320.5707
QSAR fish toxicity	279.0096	622.1914	2809.2265	4340.2099	2645.1605	194.9056
Yeast	1817.4238	4052.8550	8701.8106	17678.8337	9117.6113	1196.9684
Abalone	5077.1513	11322.0474	24401.2329	29223.1462	25279.0786	3352.4283
Anuran Calls (MFCCs)	8779.8995	19579.1758	41981.3752	50185.9836	43475.5194	5786.6943
Letter	24335.2487	54267.6046	116691.7426	139713.4244	120909.3370	16048.3162
Sensorless Drive Diagnosis	71623.0829	159719.4750	342999.0374	411265.5116	355690.0018	47175.6676
Pseudo Periodic Synthetic	122172.0769	272443.7315	585583.7128	702675.4554	607250.3102	**80491.5812**

The Average execution times (in seconds) overall datasets achieved by applying each advanced imputation method to each missing type.

2 **Table 8:** The Average execution times (in seconds) overall datasets achieved by applying each
3 advanced imputation method to each missing type.

Datasets	DMI	KDMI	KEMI	KEMI$^+$	KI	FCKI
MCAR	25446.1030	56744.8098	103294.8530	124868.2018	107286.7907	**16553.4577**
MAR	10634.5571	23715.0623	65024.9285	78416.2668	67353.9784	**7084.7802**
MNAR	10923.8582	24360.2037	56518.8458	68287.3365	58588.8357	**7330.7078**

The Average execution times (in seconds) overall datasets achieved by applying each advanced imputation method to each missing ratio.

2 **Table 9:** The Average execution times (in seconds) overall datasets achieved by applying each
3 advanced imputation method to each missing ratio.

Datasets	DMI	KDMI	KEMI	KEMI⁺	KI	FCKI
1%	4250.1756	9477.8915	23609.3678	28534.4288	24501.2017	**1847.9024**
5%	15623.3074	34839.9754	63508.3529	76767.5670	65889.8793	**8222.7933**
10%	21334.9199	47576.8715	94694.0024	114281.9859	98208.6669	**13334.3250**
20%	21464.2882	47865.3626	117973.1133	142511.7585	122373.0583	**17886.9068**

A comparison between KI and FCKI.

Table 10:1 A comparison between KI and FCKI.

Proposed Imputation Method	KI	FCKI
Purpose	Improving missing data imputation accuracy only.	Improving time efficiency as well as imputation accuracy.
Components	• It integrates k-nearest neighbors and iterative imputation. • It applies only one level of similarity using kNN algorithm.	• It integrates fuzzy c-means, k-nearest neighbors, and iterative imputation. • It applies two levels of similarity using FCM and kNN algorithms.
Average RMSE value	0.0567	0.0521
Average execution time (S)	77743	10323