A Novel Component Priority-Based and Incremental K-Means Clustering Algorithm for Imputing Incomplete Data

Kuo-Liang Chung[a] (klchung01@gmail.com), Zih-Yi Chen[a]
(M109015041@mail.ntust.edu.tw)

[a] Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, Taipei 10672, Taiwan

Corresponding Author:
Kuo-Liang Chung
Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, Taipei 10672, Taiwan
Email: klchung01@gmail.com

A Novel Component Priority-Based and Incremental K-Means Clustering Algorithm for Imputing Incomplete Data

Kuo-Liang Chung[a], Zih-Yi Chen[a]

[a]*Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, Taipei 10672, Taiwan*

Abstract

To deal with the incomplete data imputation problem, in this paper, we propose a novel and more effective component priority-based and incremental K-means (CPIK-means) algorithm. The proposed algorithm initially imputes the missing attributes in the first-priority component by using a geometry-based correlation strategy, and then based on this imputed component and existing complete components, K-means is applied to partition the dataset. In each cluster, every missing attribute in the second-priority component is imputed by the expectation maximization method, and the imputed first-priority component is updated simultaneously. In the same argument, based on the existing complete components, which consist of the imputed and original complete components, the above clustering-imputation-update process is repeated until all incomplete components are imputed. Detailed experimental results demonstrate that the proposed CPIK-means algorithm outperforms the state-of-the-art algorithms.

Keywords: Accuracy, Clustering, Expectation maximization, Incomplete data imputation, Incremental imputation, K-means.

*Corresponding author.
Email address: klchung01@gmail.com (Kuo-Liang Chung)

1. Introduction

Clustering aims to partition a dataset into nonoverlapping clusters (Duda et al., 1973; Hartigan, 1975) such that each data point is assigned to one cluster with the nearest cluster centroid. Clustering has many applications in pattern recognition, data mining, data compression, image segmentation, big data analysis, and so on. In the past decades, many clustering methods for complete data (Jain et al., 1999; Xu & Wunsch, 2005; Xu & Tian, 2015) have been developed, including K-means (Gersho & Gray, 1992; Lloyd, 1982), K-nearest neighbors (KNN) (Batista et al., 2002), fuzzy C-means (FCM) (Bezdek, 1981), and the Gaussian mixture model (Zivkovic, 2004). Due to its simplicity and effectiveness, great importance has always been attached to K-means in practical clustering applications.

Missing attributes in data points often occur in practical situations (Little & Rubin, 2019; Schafer, 1997; Hart et al., 2000). For example, for one incomplete data point, there may exist some components with missing attributes, leading to bias in conclusions. To overcome this disadvantage, imputing incomplete data is important and challenging. Previously, several incomplete data imputation methods have been developed; these developed methods include the fuzzy C-means (FCM) based imputation methods (Li et al., 2017; Hathaway & Bezdek, 2001), the Gaussian mixture based imputation methods (Zhang et al., 2021), the K-means based imputation methods (Mesquita et al., 2016; Gong et al., 2018; Wang et al., 2019; Hussain & Haris, 2019; Wang & Chen, 2020), and the KNN based imputation method (Batista et al., 2002). Among these incomplete data imputation methods, the K-means based imputation methods are still the simplest. In the following subsection, the related K-means based imputation works are introduced.

1.1. Related works

Before introducing the related K-means based imputation works, we first introduce the heuristic incomplete data imputation approach. The heuristic imputation approach first performs a pure data imputation method on the dataset,

and then performs a clustering method on the imputed dataset. The commonly used pure data imputation methods include mean-filling, zero-filling, the expectation maximization (EM) method (Dempster et al., 1977), the KNN-based imputation (KNNI) method (Batista et al., 2002), and the incomplete-case nearest neighbour imputation (ICKNNI) method (Van Hulse & Khoshgoftaar, 2014). In this study, three clustering methods, namely K-means, FCM, and the Ball K-means (BK-means) method (Xia et al., 2022), are considered. Consequently, twelve heuristic incomplete data imputation methods, namely "mean-filling + K-means", "zero-filling + K-means", "EM + K-means", "KNNI + K-means", "ICKNNI + K-means", "mean-filling + FCM", "zero-filling + FCM", "EM + FCM", "KNNI + FCM", "ICKNNI + FCM", "mean-filling + BK-means", and "EM + BK-means" are included in the comparative methods.

Mesquita *et al.* (Mesquita et al., 2016) proposed a K-means based algorithm associated with soft constraints on observed and imputed features to impute incomplete data. First, the K-nearest neighbors based imputation method is performed on the dataset. Next, some soft constraints are deployed in the complete data points and the incomplete data points. Besides, a new distance function with two extra terms is deployed in the iterative K-means clustering process to enhance the performance. For convenience, their algorithm is called the KSC-OI algorithm.

Gong *et al.* (Gong et al., 2018) proposed a Mahalanobis distance- and the information entropy-based K-means (MIK-means) algorithm to impute incomplete data. In their algorithm, first, mean-filling is used to impute the missing attributes. Next, a Mahalanobis distance is used in the clustering process to partition the dataset. Then, in each cluster, the similarity between each originally missing data point and all other complete data points is calculated. Furthermore, according to the obtained entropy information, the imputed attributes of each imputed data point are updated by considering the corresponding attributes of the most similar first few complete data points in the same cluster.

Wang *et al.* (Wang et al., 2019) first applied mean-filling to impute every missing attribute of the jth, $1 \leqslant j \leqslant d$, component of the data point D_i, $1 \leqslant$

$i \leqslant n$, in the dataset D $= \{D_1, D_2, ...D_n\}$. Next, after performing each iteration of K-means to partition the dataset into k clusters, each cluster, every imputed attribute of the jth component is updated by the mean value of all attributes in the same component. Their improved K-means algorithm is called the update based K-means (UK-means) algorithm. For convenience, their incomplete data computation algorithm is denoted by "UK-means".

Based on a three-way decision strategy (Yao, 2010), Wang and Chen (Wang & Chen, 2020) proposed a three-way ensemble clustering (TWEC) algorithm to impute incomplete data. First, K-means is applied to partition the complete sub-dataset, which only contains complete data points, into k clusters, which is the reason why TWEC only works well for low missing rate cases. After that, each data point D_i with missing attributes is assigned to the closest centroid O_j. Next, in each cluster C_j, $1 \leqslant j \leqslant k$, every missing attribute of the data point D_i is imputed by the corresponding attribute in the centroid O_j, and then the centroid O_j is updated after including the imputed data point D_i. Let $C^{(1)}$ denote the tentative clustering result after the first iteration. In the same way, the above clustering, imputation and update processes are further performed $(t-1)$ iterations, and the tentative clustering results $C^{(2)}$, $C^{(3)}$, ..., and $C^{(t)}$ are recorded. Finally, using these tentative clustering results, a three-way ensemble clustering procedure is applied to output the imputation and clustering results.

All the above-mentioned twelve heuristic imputation methods and the four state-of-the-art methods are included in the comparative methods.

1.2. Contributions

In this paper, we propose a novel and more effective component-priority based and incremental K-means (CPIK-means) algorithm to impute incomplete data. The contributions of this paper are clarified as follows.

(1) According to the number of available attributes of every incomplete component, a sorting-based component-priority vector is built up. The proposed CPIK-means algorithm initially imputes the missing attributes in the

first-priority component by using the geometry-based correlation strategy such that every missing attribute in that dimension is imputed using the higher correlated available attributes in the same dimension.

(2) Then, based on the complete components, which consist of the imputed and original complete components, K-means is used to partition the dataset into k clusters. In each cluster, incrementally, every missing attribute in the next-priority component is imputed by the expectation maximization (EM) method, and the imputed first-priority component is updated simultaneously. We repeat the above clustering-imputation-update process until all incomplete components are imputed.

(3) Based on eight typical datasets, comprehensive experimental results demonstrate that under different missing rates, the proposed CPIK-means incomplete data imputation algorithm achieves better accuracy and clustering performance relative to the above-mentioned twelve heuristic methods and the four state-of-the-art methods, namely KSC-OI (Mesquita et al., 2016), MIK-means (Gong et al., 2018), UK-means (Wang et al., 2019), and TWEC (Wang & Chen, 2020).

The rest of this paper is organized as follows. The following section describes the proposed CPIK-means incomplete data imputation algorithm. Section 3 reports experimental results to demonstrate the accuracy and clustering superiority of the proposed algorithm. Section 4 provides the concluding remarks.

2. The proposed component-priority based and incremental K-means (CPIK-means) incomplete data imputation algorithm

In this section, the proposed CPIK-means incomplete data imputation algorithm is presented. For easy exposition, one example is given to explain how the proposed CPIK-means algorithm works. Given a dataset D with ten 3-dimensional data points, which can be expressed as a 10×3 matrix

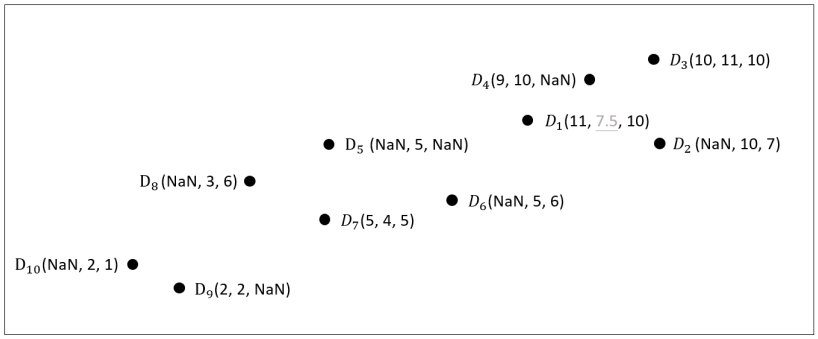
$D[1..10 : 1..3]$, D_1, D_2, ..., and D_{10}, are

$$\begin{aligned}
D_1 =& D[1,*] = (D_{1,1}, D_{1,2}, D_{1,3}) = (11, NaN, 10) \\
D_2 =& D[2,*] = (D_{2,1}, D_{2,2}, D_{2,3}) = (NaN, 10, 7) \\
D_3 =& D[3,*] = (D_{3,1}, D_{3,2}, D_{3,3}) = (10, 11, 10) \\
D_4 =& D[4,*] = (D_{4,1}, D_{4,2}, D_{4,3}) = (9, 10, NaN) \\
D_5 =& D[5,*] = (D_{5,1}, D_{5,2}, D_{5,3}) = (NaN, 5, NaN) \\
D_6 =& D[6,*] = (D_{6,1}, D_{6,2}, D_{6,3}) = (NaN, 5, 6) \\
D_7 =& D[7,*] = (D_{7,1}, D_{7,2}, D_{7,3}) = (5, 4, 5) \\
D_8 =& D[8,*] = (D_{8,1}, D_{8,2}, D_{8,3}) = (NaN, 3, 6) \\
D_9 =& D[9,*] = (D_{9,1}, D_{9,2}, D_{9,3}) = (2, 2, NaN) \\
D_{10} =& D[10,*] = (D_{10,1}, D_{10,2}, D_{10,3}) = (NaN, 2, 1)
\end{aligned} \tag{1}$$

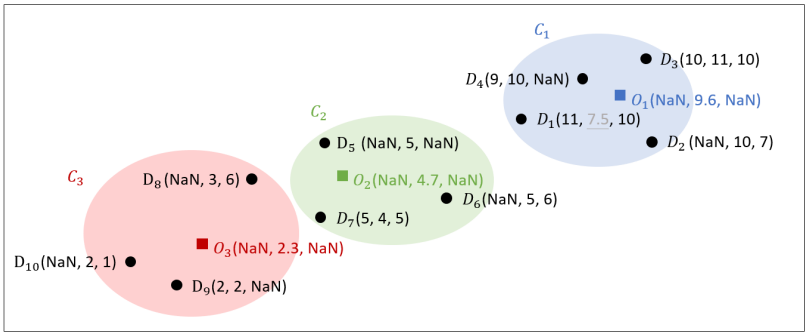
where "NaN" denotes "missing attribute". For example, for the second data point $D_2 = (NaN, 10, 7)$, the first component $D[2,1]$ has a missing attribute, but the second and third components, namely $D[2,2]$ and $D[2,3]$, have the available attributes "10" and "7", respectively.

To determine the incomplete component imputation order, we first count the number of available attributes of every component in the dataset D. For the ten data points in Eq. (1), the first, second, and third components have 5, 9, and 7 available attributes, respectively. Next, we sort the three values, namely 5, 9, and 7, in decreasing order, and the three sorted values are saved in a 1×3 vector $N^{sorted} = (9, 7, 5)$ associated with a 1×3 component-priority vector $I = (2, 3, 1)$ where $I[1] = 2$, $I[2] = 3$, and $I[3] = 1$. Here, "$I[1] = 2$" indicates that the second component of the dataset D has the most available attributes. Differing from the previous incomplete data imputation methods mentioned in the related works, our CPIK-means imputation algorithm is component-priority based and incremental, and it follows the rule: the greater the number of available attributes in one component of D, the higher priority that component has to be imputed.

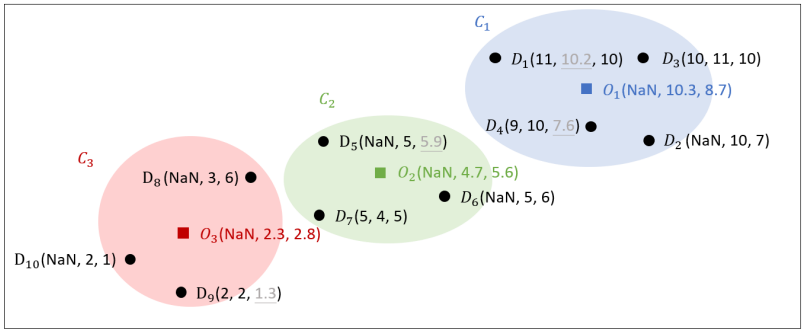
According to our component-priority based and incremental strategy, from



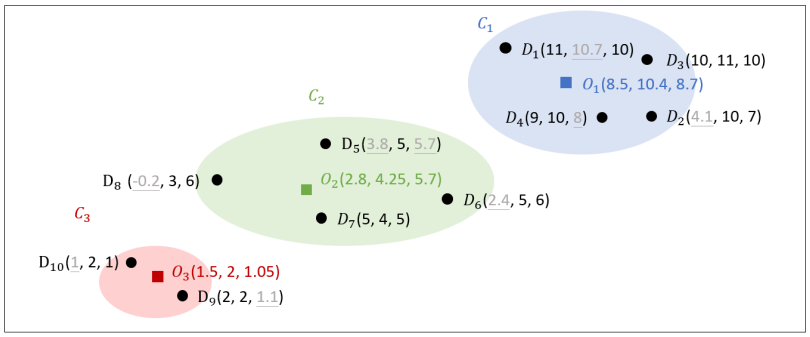
(a) The dataset after imputing the second component of the dataset D.



(b) The clustering result after performing K-means on the second component of D in Fig. 1(a).



(c) The clustering result after performing the clustering-imputation-update process on the second and third components of D in Fig. 1(b).



(d) The final clustering result.

Figure. 1. The simulation of the proposed CPIK-means incomplete data imputation algorithm.

the component-priority vector I, because of $I[1] = 2$, we thus pick up the nine attributes and one missing attribute in the second component of D, namely $D[*, 2]$, as the initial input of our CPIK-means algorithm. The initial input data $D[*, 2]$ are shown below:

$$\begin{aligned} D[*,2] &= (D_{1,2}, D_{2,2}, D_{3,2}, D_{4,2}, D_{5,2}, D_{6,2}, D_{7,2}, D_{8,2}, D_{9,2}, D_{10,2})^T \\ &= (NaN, 10, 11, 10, 5, 5, 4, 3, 2, 2)^T \end{aligned} \tag{2}$$

From the first-priority component, namely the second component $D[*, 2]$ with nine available attributes, we know the missing attribute appears in the second component of the data point D_1. Let $\tilde{D}_1$ denote a 1×2 vector concatenating two available attributes in the data point D_1, yielding $\tilde{D}_1 = (11, 10)$. Considering the data point D_j, $j \neq 1$, in which the attribute $D_{j,2}$ is available and D_j also has two available attributes collocated at the same positions as in D_1, $\tilde{D}_j$ is then constructed by concatenating the two available attributes.

From the first-priority component $D[*, 2]$, we know the attribute $D_{3,2}$ $(= 11)$ is available and it may be used to assist imputing the missing attribute $D_{1,2}$. In addition, the two attributes $D_{3,1}$ $(= 10)$ and $D_{3,3}$ $(= 10)$ are also available and they are used to construct $\tilde{D}_3$, yielding $\tilde{D}_3 = (10, 10)$. In the same argument, we can construct $\tilde{D}_7 = (5, 5)$. We further examine the similarity between $\tilde{D}_1$ and $\tilde{D}_j$, $j \in \{3, 7\}$. In what follows, the reciprocal of the proposed geometry-based

difference metric between $\tilde{D}_1$ and $\tilde{D}_j$ is used to measure the similarity of $\tilde{D}_1$ and $\tilde{D}_j$. The smaller the difference is, the higher the similarity is.

The proposed geometry-based difference metric is defined by

$$DIF(\tilde{D}_1, \tilde{D}_j) = L(\tilde{D}_1, \tilde{D}_j) + \gamma \times \theta(\tilde{D}_1, \tilde{D}_j) \tag{3}$$

with

$$L(\tilde{D}_1, \tilde{D}_j) = \frac{\max\{|\tilde{D}_1|, |\tilde{D}_j|\}}{\min\{|\tilde{D}_1|, |\tilde{D}_j|\}} - 1 \tag{4}$$

$$\theta(\tilde{D}_1, \tilde{D}_j) = \frac{\arccos\{\frac{\tilde{D}_1 \cdot \tilde{D}_j}{|\tilde{D}_1| \times |\tilde{D}_j|}\} \times 180}{\pi} \tag{5}$$

where to balance the length term in Eq. (4) and the angle term in Eq. (5), empirically, the value of γ is set to 10. The lower the value of $DIF(\tilde{D}_1, \tilde{D}_j)$ is, the higher the similarity between $\tilde{D}_1$ and $\tilde{D}_j$ is. After calculating all difference values between $\tilde{D}_1$ and $\tilde{D}_j$, $j \neq 1$, these difference values are sorted into increasing order. Empirically, we only select the five least difference values. As listed in Table 1, the number of data points in one dataset ranges from 150 to 10992, which leads to the robustness of the above selection strategy. For convenience, the above selection strategy is called the geometry-based correlation strategy. Consequently, the proposed algorithm initially imputes the missing attributes in the first-priority component, namely the second component of D_1, by the mean values of the five available attributes, which corresponds to the five least different values, in the same component.

We now want to impute the missing attribute in D_1, namely $D_{1,2}$. By using the difference metric in Eqs. (3)-(5), it yields $DIF(\tilde{D}_1, \tilde{D}_3) = 27.3143$ and $DIF(\tilde{D}_1, \tilde{D}_7) = 28.3655$. According to the geometry-based correlation strategy, the two available attributes, $D_{3,2}$ $(= 4)$ and $D_{7,2}$ $(= 11)$, are used to impute the missing attribute $D_{1,2}$. It thus yields $D_{1,2} = 7.5$ $(= \frac{1}{2}(4 + 11))$. Fig. 1(a)

depicts the dataset after imputing $D_{1,2}$. Next, K-means is performed on the second component $D[*, 2]$, and the clustering result is shown in Fig. 1(b) where the three clusters are expressed as $C_1 = \{D_1, D_2, D_3, D_4\}$, $C_2 = \{D_5, D_6, D_7\}$, and $C_3 = \{D_8, D_9, D_{10}\}$, which are associated with the centroids $O_1 = (NaN,$ 9.6, $NaN)$, $O_2 = (NaN, 4.7, NaN)$, and $O_3 = (NaN, 2.3, NaN)$, respectively.

Furthermore, we consider the second entry of the component-priority vector I, i.e. $I[2] = 3$. In each cluster of Fig. 1(b), we thus select the third component of every data point. Because all missing attributes in the second component have been imputed, we adopt the EM method (Schneider, 2001) to impute all missing attributes in the third component. Note that experimental data indicate that except for the first-priority component using the geometry-based correlation strategy, applying the EM method to impute the subsequent incomplete components achieves better accuracy and clustering performance. Before performing EM to impute all missing attributes in the third component, each of these missing attributes is tentatively imputed by the mean value of the available attributes in the same component. Then, in each cluster of Fig. 1(b), using the imputed second and third components, each imputed attribute in the second component is updated by EM. Furthermore, based on the second component and the third component, K-means is applied to partition the dataset into three clusters. Fig. 1(c) illustrates the clustering result.

We repeat the above clustering-imputation-update process until all incomplete components have been imputed. Finally, Fig. 1(d) depicts the three resultant clusters with the three centroids $O_1 = (8.5, 10.4, 8.7)$, $O_2 = (2.8, 4.25, 5.7)$, and $O_3 = (1.5, 2, 1.05)$.

The whole procedure of the proposed CPIK-means algorithm for imputing incomplete data is listed below. In the proposed algorithm, one termination condition used in K-means involves the sum of square errors (SSE) of the resultant clusters. The definition of SSE is given by

$$SSE = \frac{1}{n} \sum_{i=1}^{n} \sum_{c=1}^{k} (\sum_{j=1}^{d} \|D_{i,j} - O_{c,j}\|^2) \times X_{i,c} \tag{6}$$

where $X_{i,c}$ is defined by

$$X_{i,c} = \begin{cases} 1 & \text{when the data point } D_i \text{ belongs to the cluster } C_c \\ 0 & \text{when the data point } D_i \text{ does not belong to the cluster } C_c \end{cases} \tag{7}$$

Algorithm 1: The proposed CPIK-means incomplete data imputation algorithm

Input: The dataset $D = \{D_{i,j} \in \mathbf{R} | \, 1 \leqslant i \leqslant \text{n and } 1 \leqslant j \leqslant d\}$, the number of clusters k, the maximum iteration number ω, and the convergence tolerance ε.

Output: The resultant k clusters

Step_1: (constructing a sorted component-priority vector) We count the number of available attributes of every component in dataset D. Further, we sort these d numbers to construct a sorted $1 \times d'$ component-priority vector I in decreasing order, where "d'" denotes the number of incomplete components in D and $d' <= d$. Set $p = 1$ where p denotes the component-priority variable.

Step_2: (imputing the first-priority component) From the first-priority component information in $I[p]$, if $d' = d$, the geometry-based correlation strategy is applied to impute each missing attribute in the component $D[*, I[p]]$.

Step_3: (clustering-imputation-update process) Based on the complete components (including the imputed components), we apply K-means to partition the dataset into k clusters. K-means stops when the condition $(iteration\ number > \omega)$ holds or the condition $(\frac{SSE^{(iteration\ number-1)} - SSE^{(iteration\ number)}}{SSE^{(iteration\ number)}} \leqslant \varepsilon)$ holds, where $SSE^{(0)} = 0$. After clustering, if $p < d'$, set $p = p + 1$. Next, in each cluster, every missing attribute in $D[*, I[p]]$ is imputed by using the EM method. Simultaneously, the imputed attributes in $D[*, I[1]]$, ..., and $D[*, I[p-1]]$ are updated at the same time.

Step_4: If $p = d'$, we apply K-means to partition the dataset again, then we report the resultant k clusters as the output; otherwise, go to Step 3.

The whole C++ source code of our CPIK-means method can be accessed from the website: https://github.com/zychen5186/CPIK, in which The values of ω and ε are set to 10 and 0.0001, respectively.

3. Experimental Results

The eight testing datasets, which are downloaded from the website: https://archive.ics.uci.edu/ml/datasets.php, are used to demonstrate the accuracy and clustering benefits of the proposed CPIK-means algorithm relative to the twelve heuristic imputation methods and the four state-of-the-art algorithms, namely KSC-OI (Mesquita et al., 2016), MIK-means (Gong et al., 2018), UK-means (Wang et al., 2019), and TWEC (Wang & Chen, 2020). The eight used datasets are Iris, Wine, Seeds, Glass, Leaf, Movement Libras, Breast Cancer, and Pendigits. Table 1 lists the detailed information of the eight datasets, and in Table 1, #(data points), #(components), and #(classes) denote the number of data points in the dataset, the dimension of each data point, and the number of partitioned clusters, respectively. For example, for the dataset "Iris", there are 150 data points and each data point is 4-dimension, i.e. it has four attributes; the number of partitioned clusters is 3.

The three metrics, namely accuracy (ACC), F-score, and normalized mutual information (NMI) (Amelio & Pizzuti, 2015) are used to compare the accuracy and clustering performance among the comparative methods and the proposed CPIK-means algorithm. In our experiment, the missing ratio of the dataset ranges from 5% to 40%, and the performance of each considered method is demonstrated by a curve connecting eight points corresponding to the missing ratios, 5%, 10%, 15%, ..., 35%, and 40%, respectively.

The execution code of the BK-means method (Xia et al., 2022) can be accessed from the website: https://github.com/syxiaa/ball-k-means. The UK-means method (Wang et al., 2019) can be accessed from the website: https://github.com/wangsiwei2010/k-means-filling, and UK-means has good performance in terms of the above-mentioned three metrics in the best case. However, for

comparison fairness, each considered imputation method is run 50 times with random initialization for each dataset in the implementation and the average accuracy and clustering performance is demonstrated. We have tried our best to tune the parameters required in the twelve heuristic methods and the remaining three state-of-the-art methods. For comparison fairness, the execution codes of all considered methods are run on the same computer with an AMD Ryzen 7 3800X 8-Core Processor 3.89 GHz and 32 GB RAM. The operating system is the Microsoft Windows 10 64-bit operating system. All methods are implemented in the programming language C++ with g++ 6.3.0.

Table 1: The eight used datasets.

dataset	**#(data points)**	**#(components)**	**#(classes)**
Iris	150	4	3
Wine	178	13	3
Seeds	210	7	3
Glass	214	9	6
Leaf	340	14	30
Movement Libras	360	90	15
Breast Cancer	569	30	2
Pen Digits	10992	16	10

In what follows, thorough experimental results demonstrate the accuracy and clustering benefits of the proposed CPIK-means algorithm for imputing incomplete data when compared with the comparative methods.

3.1. Accuracy (ACC) comparison

The ACC metric is defined by

$$ACC = \frac{1}{n}\sum_{i=1}^{k} TP_i \tag{8}$$

where n denotes the number of data points in the dataset. TP_i denotes the number of data points that are correctly classified to the ith cluster, $1 \leqslant i \leqslant k$.

Fig. 2 illustrates the average ACC performance comparison of all considered algorithms. From Fig. 2, under different missing rates, we observe that the proposed CPIK-means algorithm, abbreviated as "Ours", has the highest average ACC performance in red relative to the 16 comparative algorithms. Note that because the TWEC method (Wang & Chen, 2020) only works well for low missing rate cases, we just show a short curve for TWEC (Wang & Chen, 2020).

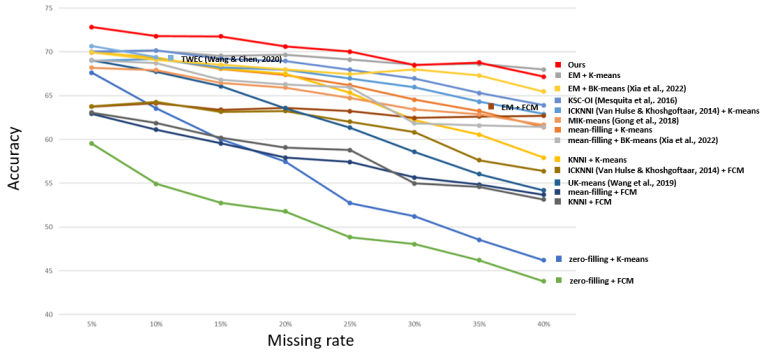


Figure. 2. ACC performance benefit of the proposed CPIK-means algorithm "Ours" relative to the 16 comparative methods.

3.2. The F-score comparison

The F-score metric is expressed by

$$\text{F–score} = \frac{1}{k}\sum_{i=1}^{k} 2 \times \frac{precision_i \times recall_i}{precision_i + recall_i} \tag{9}$$

with

$$precision_i = \frac{TP_i}{TP_i + FP_i} \tag{10}$$

$$recall_i = \frac{TP_i}{TP_i + FN_i} \tag{11}$$

where "precision$_i$" denotes the rate of the number of true positive (TP) data points over the sum of the number of TP data points and false positive (FP) data points in the ith cluster. "recall$_i$" denotes the rate of the number of true

positive (TP) data points over the sum of the number of TP data points and false negative (FN) data points the ith cluster. A greater F-score indicates a better clustering result.

Fig. 3 illustrates the F-score performance comparison of all considered algorithms, and it indicates that our algorithm has the highest average F-score performance among all considered algorithms.

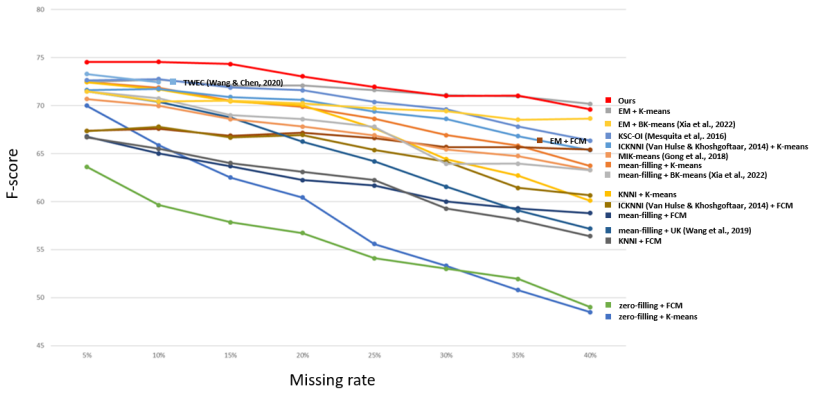


Figure. 3. The F-score performance benefit of the proposed algorithm "Ours" when compared with 16 comparative methods.

3.3. The normalized mutual information (NMI) performance comparison

The normalized mutual information (NMI) is expressed by

$$NMI(Y,C) = \frac{2 \times I(Y;C)}{[H(Y) + H(C)]} \tag{12}$$

with

$$I(Y;C) = H(Y) - H(Y|C) \tag{13}$$

$$H(Y) = -\sum_{i=1}^{n} P(y) \log_2 P(y) \tag{14}$$

$$
\begin{aligned}
H(Y|C) &= \sum_{c \in C} p(c) H(Y|C=c) \\
&= -\sum_{c \in C} p(c) \sum_{y \in Y} p(y|c) \log p(y|c)
\end{aligned} \tag{15}
$$

where $H(Y)$ and $H(C)$ denote the marginal entropy of the ground truth clusters and the partitioned clusters by using the considered clustering method, respectively. $H(Y|C)$ denotes the conditional entropy. $I(Y;C)$ denotes the mutual information between the ground truth clusters and partitioned clusters. A greater value of NMI indicates for a better clustering result.

Fig. 4 illustrates the NMI performance comparison of the proposed algorithm "Ours" and the 16 comparative methods. Fig. 4 demonstrates that the proposed CPIK-means algorithm "Ours" has the highest average NMI performance among all considered algorithms.

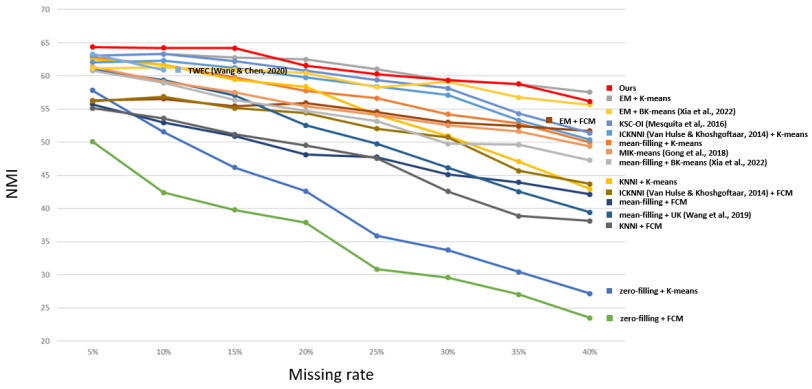


Figure. 4. The NMI performance benefit of the proposed algorithm "Ours" when compared with 16 comparative methods.

4. Conclusion

Our novel component-priority based and incremental K-means (CPIK-means) algorithm has been presented for imputing incomplete data. First, based on the number of available attributes of every incomplete component, a sorting-based

component-priority vector is built up. Next, the proposed algorithm imputes the missing attributes in the first-priority component, and then based on the imputed component and existing complete components, K-means is applied to partition the dataset into k clusters. In each cluster, incrementally, the missing attributes in the component with next order are imputed, and simultaneously, the previously imputed attributes are updated. In line with the same argument, the above clustering-imputation-update process is repeated until all missing attributes are imputed. Based on eight benchmark datasets and three accuracy metrics, detailed experimental results demonstrate that under the missing rate interval [5%, 40%], the proposed algorithm achieves substantial accuracy and clustering improvements relative to the 16 comparative methods.

Acknowledgment

The authors appreciate the proofreading help of Ms. C. Harrington to improve the manuscript. This work was supported by the contracts MOST-108-2221-E-011-077-MY3 and MOST-110-2221-E-011-088-MY3 of the Ministry of Science and Technology, Taiwan.

References

Amelio, A., & Pizzuti, C. (2015). Is normalized mutual information a fair measure for comparing community detection methods. (p. 1584–1585). doi:10.1145/2808797.2809344.

Batista, G. E., Monard, M. C. et al. (2002). A study of k-nearest neighbour as an imputation method. *His*, *87*, 48. doi:10.1109/TIT.1982.1056489.

Bezdek, J. C. (1981). *Pattern recognition with fuzzy objective function algorithms*. Springer Science & Business Media.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical So-*

ciety: Series B, *39*, 1–22. doi:https://doi.org/10.1111/j.2517-6161.1977.tb01600.x.

Duda, R. O., Hart, P. E., & Stork, D. G. (1973). *Pattern classification and scene analysis*. Wiley New York.

Gersho, A., & Gray, R. M. (1992). *Vector quantization and signal compression*. Springer Science & Business Media.

Gong, X., Zhang, J., & Shi, Y. (2018). Research on data filling algorithm based on improved k-means and information entropy. In *2018 IEEE 4th International Conference on Computer and Communications*.

Hart, P. E., Stork, D. G., & Duda, R. O. (2000). *Pattern classification*. Wiley Hoboken.

Hartigan, J. A. (1975). *Clustering algorithms*. John Wiley and Sons, Inc.

Hathaway, R. J., & Bezdek, J. C. (2001). Fuzzy c-means clustering of incomplete data. *IEEE Transactions on Systems*, *31*, 735–744. doi:10.1109/3477.956035.

Hussain, S. F., & Haris, M. (2019). A k-means based co-clustering (kcc) algorithm for sparse, high dimensional data. *Expert Systems with Applications*, *118*, 20–34.

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: a review. *ACM computing surveys*, *31*, 264–323. doi:10.1145/331499.331504.

Li, T., Zhang, L., Lu, W., Hou, H., Liu, X., Pedrycz, W., & Zhong, C. (2017). Interval kernel fuzzy c-means clustering of incomplete data. *Neurocomputing*, *237*, 316–331. doi:10.1109/TIT.1982.1056489.

Little, R. J., & Rubin, D. B. (2019). *Statistical analysis with missing data*. John Wiley and Sons.

Lloyd, S. (1982). Least squares quantization in pcm. *IEEE transactions on information theory*, *28*, 129–137. doi:10.1109/TIT.1982.1056489.

Mesquita, D. P., Gomes, J. P., & Rodrigues, L. R. (2016). K-means for datasets with missing attributes: Building soft constraints with observed and imputed values. In *European Symposium on Artificial Neural Networks*.

Schafer, J. L. (1997). *Analysis of incomplete multivariate data*. CRC press.

Schneider, T. (2001). Analysis of incomplete climate data: Estimation of mean values and covariance matrices and imputation of missing values. *Journal of climate*, *14*, 853–871.

Van Hulse, J., & Khoshgoftaar, T. M. (2014). Incomplete-case nearest neighbor imputation in software measurement data. *Information Sciences*, *259*, 596–610. doi:https://doi.org/10.1016/j.ins.2010.12.017.

Wang, P., & Chen, X. (2020). Three-way ensemble clustering for incomplete data. *IEEE Access*, *8*, 91855–91864. doi:10.1109/ACCESS.2020.2994380.

Wang, S., Li, M., Hu, N., Zhu, E., Hu, J., Liu, X., & Yin, J. (2019). K-means clustering with incomplete data. *IEEE Access*, *7*, 69162–69171. doi:10.1109/ACCESS.2019.2910287.

Xia, S., Peng, D., Meng, D., Zhang, C., Wang, G., Giem, E., Wei, W., & Chen, Z. (2022). Ball k-means: Fast adaptive clustering with no bounds. *IEEE transactions on pattern analysis and machine intelligence*, *44*, 87–99.

Xu, D., & Tian, Y. (2015). A comprehensive survey of clustering algorithms. *Annals of Data Science*, *2*, 165–193. doi:https://doi.org/10.1007/s40745-015-0040-1.

Xu, R., & Wunsch, D. (2005). Survey of clustering algorithms. *IEEE Transactions on neural networks*, *16*, 645–678. doi:10.1109/TNN.2005.845141.

Yao, Y. (2010). Three-way decisions with probabilistic rough sets. *Information Sciences*, *180*, 341–353. doi:https://doi.org/10.1016/j.ins.2009.09.021.

Zhang, Y., Li, M., Wang, S., Dai, S., Luo, L., Zhu, E., Xu, H., Zhu, X., Yao, C., & Zhou, H. (2021). Gaussian mixture model clustering with incomplete data. *ACM Transactions on Multimedia Computing, Communications, and Applications*, *17*, 1–14. doi:10.1145/3408318.

Zivkovic, Z. (2004). Improved adaptive gaussian mixture model for background subtraction. In *Proceedings of the 17th International Conference on Pattern Recognition* (pp. 28–31). doi:10.1109/ICPR.2004.1333992.