Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2016, Article ID 4321928, 8 pages
http://dx.doi.org/10.1155/2016/4321928

Research Article

Robust K-Median and K-Means Clustering Algorithms for Incomplete Data

Jinhua Li,[1] Shiji Song,[1] Yuli Zhang,[1,2] and Zhen Zhou[3]

[1] *Department of Automation, TNList, Tsinghua University, Beijing 100084, China*
[2] *Department of Industrial Engineering, Tsinghua University, Beijing 100084, China*
[3] *School of Management, Capital Normal University, Beijing 100089, China*

Correspondence should be addressed to Shiji Song; shijis@mail.tsinghua.edu.cn and Zhen Zhou; zhouzhen@cnu.edu.cn

Received 25 May 2016; Accepted 31 October 2016

Academic Editor: Hung-Yuan Chung

Incomplete data with missing feature values are prevalent in clustering problems. Traditional clustering methods first estimate the missing values by imputation and then apply the classical clustering algorithms for complete data, such as K-median and K-means. However, in practice, it is often hard to obtain accurate estimation of the missing values, which deteriorates the performance of clustering. To enhance the robustness of clustering algorithms, this paper represents the missing values by interval data and introduces the concept of robust cluster objective function. A minimax robust optimization (RO) formulation is presented to provide clustering results, which are insensitive to estimation errors. To solve the proposed RO problem, we propose robust K-median and K-means clustering algorithms with low time and space complexity. Comparisons and analysis of experimental results on both artificially generated and real-world incomplete data sets validate the robustness and effectiveness of the proposed algorithms.

1. Introduction

In the field of data mining and machine learning, it is a common occurrence that the considered data sets contain several observations with missing feature values. Such incomplete data occur in a wide array of application domains due to various reasons, including improper collection process of data sets, high cost to obtain some feature values, and missing response in the questionnaire. For example, online shopping users may only rate a small fraction of the available books, movies, or songs, which leads to massive amounts of missing feature values, Marlin [1]. Theoretical study of pattern recognition for incomplete data is first conducted by Sebestyen [2] under certain probabilistic assumptions. Expectation maximization algorithms have also been proposed to compute maximum likelihood estimates for missing data in Dempster et al. [3]. Early empirical studies on incomplete data are reported by Dixon [4] and Jain and Dubes [5].

Clustering analysis has been regarded as an effective method to extract useful features and explore potential data patterns. Due to the presence of missing feature values, there is an urgent need to cluster incomplete data in many fields, such as image analysis [6], information retrieval [7], and clinical medicine [8]. To cluster incomplete data, the basic approach is the two-step method, which first estimates the missing feature values using imputation and then applies the classical clustering methods. Troyanskaya et al. [9] investigate three imputation based clustering methods for gene microarray data, including the singular value decomposition, weighted K-nearest neighbors (KNN), and row average methods. Troyanskaya et al. [9] conclude that the KNN method appears to provide a more robust and sensitive result for missing value estimation than others. Miyamoto et al. [10] also use a similar imputation based fuzzy c-means (FCM) method to handle incomplete data. Acuna and Rodriguez [11] and Farhangfar et al. [12] compare the performance of different imputation methods for missing values, including single imputation methods, such as the mean, median, hot deck, and Naive-Bayes methods and the polytomous regression based multiple imputation method for classification problems. Saravanan and Sailakshmi [13] propose fuzzy probabilistic c-means algorithms to impute the missing values using the genetic algorithm.

Besides the imputation based methods, Hathaway and Bezdek [14] propose four strategies to make the classical FCM clustering algorithm applicable to incomplete data. The simplest whole data strategy (WDS) deletes all incomplete samples and applies the FCM algorithm to the remaining complete data. This strategy is only useful when only a few incomplete samples include missing values. To calculate distances of missing data in the process of implementing FCM, the partial distance strategy (PDS) can be used. PDS has also been used in pattern recognition in Dixon [4] and fuzzy clustering with missing values in Miyamoto et al. [10] and Timm and Kruse [15]. The third and fourth strategies can be viewed as iterative imputation based methods. The optimal completion strategy (OCS) imputes the missing values by the maximum likelihood estimate in an iterative optimization procedure, and the nearest prototype strategy (NPS) is a simple modification of OCS, in which missing elements are imputed considering only the nearest prototype. Clustering methods without elimination or imputation for incomplete data have also been proposed. Shibayama [16] uses the principal component analysis (PCA) method to capture the structure of incomplete data and Honda and Ichihashi [17] propose linear fuzzy clustering methods based on the local PCA. Zhang and Chen [18] propose a kernel-based FCM clustering algorithm for incomplete data, which estimates the missing feature values based on the fuzzy membership and cluster prototype. Sadaaki et al. [19] further combine the linear fuzzy clustering with PDS, OCS, and NPS proposed by Hathaway and Bezdek [14].

Both direct imputation and iterative imputation (such as OCS, NPS) methods assume that the miss feature value can be well estimated by a single value. However, it is usually hard to obtain accurate estimates of the missing values, and thus clustering methods based on imputation are sensitive to the estimation accuracy. To address this issue, Li et al. [20] use nearest-neighbor intervals to represent the missing values and extend FCM by defining new interval distance function for interval data. Interval data have been verified as an effective way to handle the missing values and further used to propose effective clustering methods. Li et al. [21] also represent the missing values by interval data but search for appropriate imputations of missing values in the intervals using the genetic algorithm. Wang et al. [22] use an improved back-propagation (BP) neural network to estimate the interval data for missing values. Zhang et al. [23] propose an improved interval construction method based on preclassification results and use the particle swarm optimization to search for the optimal clustering. Zhang et al. [8] represent the missing values by probabilistic information granules and design an efficient trilevel alternating optimization method to find both the optimal clustering results and the optimal missing values simultaneously.

Recently, robust optimization has been widely accepted as an effective method to handle uncertain or missing data and used in the field of data mining and machine learning, such as the minimax probability machine [24–27], robust support vector machines [28, 29], and robust quadratic regression [30]. This paper aims at designing robust clustering algorithms for incomplete data. The improved interval construction method based on preclassification is used to obtain the interval data for missing values. Based on the interval data representation, we present robust K-median and K-means clustering algorithms. Different from the existing algorithms, which use either the interval distance function or optimal imputation [20, 21, 23], we reformulate the clustering problem as a minimax robust optimization problem based on interval data.

Specifically, for given cluster prototype and membership matrices, we introduce a concept of robust clustering objective function, which is the maximum of clustering objective function when the missing values vary in the constructed intervals. Then the proposed algorithms aim at finding optimal cluster prototype and membership matrices, which minimize the robust clustering objective function. For both robust K-median and K-mean clustering problems, we give equivalent reformulations for the robust objective function and present effective solution methods. Compared with existing methods, the proposed algorithms are insensitive to estimation errors of the constructed intervals, especially when the missing rate is high. Comparisons and analysis of numerical experimental results on UCI data sets also validate the effectiveness of the proposed robust algorithms.

Compared with existing algorithms, the advantages of the proposed robust clustering algorithms are twofold. First, our algorithms can cluster incomplete data without imputation for the missing feature values and provide robust clustering results, which are insensitive to estimation errors. Our experiments also validate the effectiveness of the proposed algorithm in terms of robustness and accuracy by comparison with existing algorithms. Second, the proposed algorithms are easy to understand and implement. Specifically, the time complexity of the robust K-median and K-means clustering algorithms is $O(nmKT)$ and $O(nm(K+\log n)T)$, respectively, where n is the number of objects, m is the dimension of features, K is the number of clusters, and T is the number of iterations. Our algorithms have similar computation complexity to the classical K-median and K-means clustering algorithms and are more efficient than the clustering algorithms for incomplete data proposed by Zhang et al. [8] with the time complexity of $O(nmK^2T)$ (when $\log n \leq K^2$ for the robust K-means clustering algorithm).

The paper is organized as follows. Section 2 reviews the classical K-median and K-mean algorithms and presents the robust K-median and K-means clustering problems. Section 3 gives effective algorithms for the proposed robust optimization problems. Section 4 reports experimental results. Finally, we conclude this paper with further research direction in Section 5.

2. Robust Clustering Algorithms

2.1. K-Median and K-Means Clustering for Complete Data. Consider the problem of clustering a set of n objects $I = \{1, \dots, n\}$ into K clusters. For each object $i \in I$, we have a set of m features $\{x_{ij} : j \in J\}$, where x_{ij} describes the jth features of the object i quantitatively. Let $x_i = (x_{i1}, \ldots, x_{im})^{\mathrm{T}}$ be the feature vector of the object i and $X = (x_1, \ldots, x_n)$ be the feature matrix or data set.

The task of clustering can be reformulated as an optimization problem, which minimizes the following clustering objective function:

$$\min \quad J(U,V) = \sum_{k=1}^{K} \sum_{i \in I} u_{ik} \left\| x_i - v_k \right\|_p^p, \tag{1}$$

under the following constraints:

$$\sum_{k=1}^{K} u_{ik} = 1, \quad u_{ik} \in \{0,1\}, \ \forall i \in I, \ k = 1, \ldots, K, \tag{2}$$

where $p = 1, 2$. For $k = 1, \ldots, K$, $v_k \in R^m$ is the kth cluster prototypes and, for any $i \in I$, u_{ik} indicates whether the object i belongs to the kth cluster. K-median and K-means are effective algorithms to solve the clustering problem for $p = 1$ and $p = 2$, respectively. In the following, let the cluster prototype matrix $V = [v_1, \ldots, v_K] \in R^{m \times K}$ and the membership matrix $U = [u_1, \ldots, u_n] \in R^{K \times n}$, where $v_i = (v_{i1}, \ldots, v_{im})^{\mathrm{T}}$ and $u_i = (u_{i1}, \ldots, u_{iK})^{\mathrm{T}}$.

Both algorithms solve the clustering problem in iterative ways as follows.

Step 1. Set iteration index $t = 0$ and randomly select K different objects as the initial cluster prototypes $\{v_k^t : k = 1, \ldots, K\}$.

Step 2. Let $t = t + 1$, and update the membership matrix U^t by fixing the cluster prototype matrix V^{t-1}. For any $i \in I$, randomly select $k^* \in \arg\min\{\|x_i - v_k^{t-1}\|_p : k = 1, \ldots, K\}$, and set $u_{ik^*}^t = 1$ and, for any $k \neq k^*$, set $u_{ik}^t = 0$.

Step 3. Update the cluster prototype matrix V^t by fixing the membership matrix U^t. When $p = 1$, for any $k = 1, \ldots, K$ and $j \in J$, set v_{kj}^t as the median of the jth feature values of these objects in cluster k. When $p = 2$, for any $k = 1, \ldots, K$, set v_k^t as the centroid of these objects in cluster k; that is, $v_k^t = (1/\sum_{i \in I} u_{ik}) \sum_{i \in I} u_{ik} x_i$.

Step 4. If, for any $i \in I$ and $k = 1, \ldots, K$, we have $u_{ik}^t = u_{ik}^{t-1}$, then stop and return to U and V; otherwise, go to Step 2.

2.2. Robust K-Median and K-Means Clustering for Incomplete Data.

Due to various reasons, the feature matrix X may contain missing components. For example, when $|J| = 3$, for a certain object $i \in I$, we may have $x_i = (1, 0.5, ?)^{\mathrm{T}}$, which indicates that the third-feature value of object i is missing. We refer to a data set X as an incomplete data set if it contains at least one missing feature value for some objects; that is, there exists at least one $i \in I$ and $j \in J$, such that $x_{ij} = ?$. To describe the missing data set, for any $i \in I$, we further partition the feature set of i into two subsets:

$$J_i^0 = \left\{ j : x_{ij} = \ ?, \ \forall j \in J \right\}, \quad J_i^1 = J \setminus J_i^0. \tag{3}$$

In practice, it is difficult to obtain accurate estimations of missing feature values. Thus, in this paper, we represent missing values by intervals. Specifically, for any $i \in I$, we use an interval $[x_{ij}^-, x_{ij}^+]$ to represent unknown missing feature value where $j \in J_i^0$ and use $\overline{x}_{ij}$ to represent known feature value where $j \in J_i^1$. To simplify notations, in the following, let $\overline{x}_{ij} = (x_{ij}^- + x_{ij}^+)/2$ and $\delta_{ij} = (x_{ij}^+ - x_{ij}^-)/2$ for any $j \in J_i^0$ and $\delta_{ij} = 0$ for any $j \in J_i^1$. For details on how to construct these intervals for missing values, see Li et al. [20] and Zhang et al. [23].

This paper aims at designing robust clustering methods, such that the worst-case performance of the cluster output can be guaranteed. The logic of the proposed method can be explained as a two-player game: a clustering decision-maker first makes clustering decision, and then an adversarial player chooses values of missing features from certain intervals. Thus, a robust clustering decision-maker will select the cluster, such that the worst-case cluster objective function is minimized.

To introduce robust clustering problem, we first define the following robust cluster objective function:

$$J^R(U,V) = \max \left\{ \sum_{k=1}^{K} \sum_{i \in I} u_{ik} \left\| \overline{x}_i + y_i - v_k \right\|_p^p : y_i \in [-\delta_i, \delta_i], \ \forall i \in I \right\}, \tag{4}$$

where $\delta_i = (\delta_{i1}, \ldots, \delta_{im})^{\mathrm{T}}$, $y_i = (y_{i1}, \ldots, y_{im})^{\mathrm{T}}$, and y_{ij} represents the uncertainty in the jth feature of the object i. Thus, the robust clustering problem can be formulated as follows:

$$(\text{RCP}) \quad \min \left\{ J^R(U,V) : \text{subject to } (2) \right\}. \tag{5}$$

(RCP) is a discrete minimax problem. When there is no missing data, that is, $J_i^0 = \emptyset$ for any $i \in I$, (RCP) reduces to the classical clustering problem (1). Since problem (1) is NP-hard problem [31, 32], finding the global optimal solution of (RCP) is a challenging task. In the next section, we propose effective robust K-median and K-means algorithms for (RCP).

3. Algorithms

3.1. Robust K-Median Clustering Algorithm.

In this subsection, we provide a robust K-median clustering algorithm for (RCP) when $p = 1$. We first show how to simplify the robust cluster objective function.

$$J^R(U,V) = \sum_{i \in I} \sum_{j \in J} \max \left\{ \sum_{k=1}^{K} u_{ik} \left| \overline{x}_{ij} + y_{ij} - v_{kj} \right| : -\delta_{ij} \le y_{ij} \le \delta_{ij} \right\} \tag{6}$$

$$= \sum_{i\in I}\sum_{j\in J}\max\left\{\sum_{k=1}^{K}u_{ik}\left|\overline{x}_{ij}-\delta_{ij}-v_{kj}\right|,\sum_{k=1}^{K}u_{ik}\left|\overline{x}_{ij}+\delta_{ij}-v_{kj}\right|\right\} \quad (7)$$

$$= \sum_{i\in I}\sum_{j\in J}\sum_{k=1}^{K}u_{ik}\max\left\{\left|\overline{x}_{ij}-\delta_{ij}-v_{kj}\right|,\left|\overline{x}_{ij}+\delta_{ij}-v_{kj}\right|\right\}, \quad (8)$$

where (7) uses the fact that the maximum of a convex function over a convex set is attained at extreme points and (8) uses constraints (2). Since $\max\{|x-y|,|x+y|\}=|x|+|y|$ and $\delta_{ij}=0$, for any $i\in I$ and $j\in J_i^1$, we further have

$$J^R(U,V)=\sum_{k=1}^{K}\sum_{i\in I}\sum_{j\in J}u_{ik}\left|\overline{x}_{ij}-v_{kj}\right|+\sum_{k=1}^{K}\sum_{i\in I}\sum_{j\in J_i^0}u_{ik}\delta_{ij}. \quad (9)$$

Equation (9) shows that the existence of missing values increases the cluster objective function. Based on (9), the robust K-median clustering algorithm can be given in Algorithm 1.

Algorithm 1 (robust K-median clustering algorithm).

Input. The feature matrix $\overline{X}$, interval size δ_{ij} $(i\in I,\ j\in J)$ and K.

Output. The cluster prototype matrix V^* and membership matrix U^*.

Step 1 (initialization). Set iteration index $t=0$ and randomly select K different rows from $\overline{X}$ as the initial cluster prototypes $\{v_k^t : k=1,\ldots,K\}$.

Step 2. Let $t=t+1$ and update U^t by fixing V^{t-1}.

For any $i\in I$, randomly select $k^*\in\arg\min\{\sum_{j\in J}|\overline{x}_{ij}-v_{kj}^{t-1}|+\sum_{j\in J_i^0}\delta_{ij} : k=1,\ldots,K\}$, and set $u_{ik^*}^t=1$ and, for any $k\neq k^*$, set $u_{ik}^t=0$.

Step 3. Update V^t by fixing U^t:

For any $k=1,\ldots,K$, let $I_k=\{i\in I : u_{ik}^t=1\}$. For any $j\in J$, set v_{kj}^t as the median of $\{\overline{x}_{ij} : i\in I_k\}$.

Step 4 (stop criterion). If $u_{ik}^t = u_{ik}^{t-1}$ for any $i\in I$ and $k=1,\ldots,K$, then stop and return to $U^*=U^t$ and $V^*=V^t$; otherwise, go to Step 2.

3.2. Robust K-Means Clustering Algorithm. In this subsection, a robust K-median clustering algorithm for (RCP) when $p=2$ is proposed. Similarly to the analysis of $J^R(U,V)$ when $p=1$, we first simply the robust cluster objective function as follows:

$$\begin{aligned} J^R&(U,V)\\ &=\sum_{i\in I}\sum_{j\in J}\max\left\{\sum_{k=1}^{K}u_{ik}\left(\overline{x}_{ij}+y_{ij}-v_{kj}\right)^2 : -\delta_{ij}\le y_{ij}\le\delta_{ij}\right\}\\ &=\sum_{i\in I}\sum_{j\in J}\max\left\{\sum_{k=1}^{K}u_{ik}\left(\overline{x}_{ij}-\delta_{ij}-v_{kj}\right)^2,\sum_{k=1}^{K}u_{ik}\left(\overline{x}_{ij}+\delta_{ij}-v_{kj}\right)^2\right\}\\ &=\sum_{i\in I}\sum_{j\in J}\sum_{k=1}^{K}u_{ik}\max\left\{\left(\overline{x}_{ij}-\delta_{ij}-v_{kj}\right)^2,\left(\overline{x}_{ij}+\delta_{ij}-v_{kj}\right)^2\right\}. \end{aligned} \quad (10)$$

Since $\max\{(x-y)^2,(x+y)^2\}=x^2+y^2+2|x||y|$, we have

$$\begin{aligned} J^R(U,V)&=\sum_{k=1}^{K}\sum_{i\in I}u_{ik}\\ &\cdot\left(\left\|\overline{x}_i-v_k\right\|_2^2+\sum_{j\in J_i^0}\left(2\delta_{ij}\left|\overline{x}_{ij}-v_{kj}\right|+\delta_{ij}^2\right)\right). \end{aligned} \quad (11)$$

To minimize $J^R(U,V)$, we need to update U and V in an alternative manner. Specifically, when the value of V is fixed, each object $i\in I$ can be assigned to any cluster in the following index set:

$$\arg\min \left\{\left\|\overline{x}_i-v_k\right\|_2^2+\sum_{j\in J_i^0}\left(2\delta_{ij}\left|\overline{x}_{ij}-v_{kj}\right|+\delta_{ij}^2\right) : k=1,\ldots,K\right\}. \quad (12)$$

When the value of U is fixed, for each cluster $k=1,\ldots,K$, let $I_k=\{i\in I : u_{ik}=1\}$. Then the optimal value of v_k^* can be obtained by solving the following piecewise convex optimization problem:

$$\min \sum_{i\in I_k}\sum_{j\in J}\left(\left(\overline{x}_{ij}-v_{kj}\right)^2+2\delta_{ij}\left|\overline{x}_{ij}-v_{kj}\right|\right). \quad (13)$$

Note that optimization problem (13) is decomposable in j. Thus, to obtain the optimal value of v_{kj}^*, it is sufficient to solve the following subproblem:

$$\min \quad f\left(v_{kj}\right)=\sum_{i\in I_k}\left(\left(\overline{x}_{ij}-v_{kj}\right)^2+2\delta_{ij}\left|\overline{x}_{ij}-v_{kj}\right|\right). \quad (14)$$

Procedure 1 (procedure of solving the Subproblem (14)).

Input. Given k and j, I_k, $\overline{X}$ and δ_{ij} $(i\in I_k)$.

Output. v_{kj}^*.

Step 1 (ranking). Rank $\{\overline{x}_{ij} : i\in I_k\}$ in the increasing order. To simplify notations, in the following, we omit indices k and j, and suppose $\overline{x}_{1j}\le\cdots\le\overline{x}_{n_kj}$, where $n_k=|I_k|$.

Step 2. Identify potential minimum points.

For $l = 1, \dots, n_k + 1$, calculate $v^l = (\sum_{i=1}^{n_k} x_i + \sum_{i=l}^{n_k} \delta_i - \sum_{i=1}^{l-1} \delta_i)/n_k$.

Step 3. Return to $v^*_{kj} = \arg\min\{f(v) : v = v^l,\ 1 \le l \le n_k + 1,\ v = \overline{x}_{ij},\ i \in I_k\}$.

Subproblem (14) is a piecewise convex quadratic optimization problem and can be solved by Procedure 1.

Procedure 1 solves Subproblem (14) by enumerating all potential minimum points. It is easy to see that Procedure 1 can be implemented in $\mathcal{O}(n_k \log n_k)$ time if the ranking step uses effective sorting methods, such as the Heapsort.

Based on the above discussion, the robust K-means clustering algorithm can be described in Algorithm 2.

Algorithm 2 (robust K-means clustering algorithm).

Input. The feature matrix $\overline{X}$, interval size δ_{ij} $(i \in I, j \in J)$ and K.

Output. The cluster prototype matrix V^* and membership matrix U^*.

Step 1 (initialization). Set iteration index $t = 0$ and randomly select K different rows from $\overline{X}$ as the initial cluster prototypes $\{v_k^t : k = 1, \dots, K\}$.

Step 2. Let $t = t + 1$ and update U^t by fixing V^{t-1}:

For any $i \in I$, randomly select k^* that belongs to the index set (12).

For any $k \ne k^*$, set $u_{ik}^t = 0$.

Step 3. Update V^t by fixing U^t.

For any $k = 1, \dots, K$ and $j \in J$, obtain v_{kj}^t using Procedure 1.

Step 4 (stop criterion). If $u_{ik}^t = u_{ik}^{t-1}$ for any $i \in I$ and $k = 1, \dots, K$, then stop and return to $U^* = U^t$ and $V^* = V^t$; otherwise, go to Step 2.

3.3. Computational Complexity. It is well known that the time complexity of the classical K-median and K-means algorithms is $\mathcal{O}(nmKT)$, where $n = |I|$ is the number of objects, $m = |J|$ is the dimension of features, K is the number of clusters, and T is the number of iterations. We will show that the proposed robust K-median clustering algorithm has an $O(nmKT)$ time complexity and the robust K-means clustering algorithm has an $O(nm(K + \log n)T)$ time complexity.

Specifically, the initialization step of Algorithm 1 takes $\mathcal{O}(mK)$ time to initialize the cluster prototype matrix. For a given cluster prototype matrix, Algorithm 1 takes $\mathcal{O}(nmK)$ time to update the membership matrix. Note that the median of n scalar can be computed in $\mathcal{O}(n)$ time [33]. Let $|I_k| = n_k$ and we have $\sum_{k=1}^{K} n_k = n$. Therefore, Step 3 of Algorithm 1 can be implemented in $\mathcal{O}(nK + m\sum_{k=1}^{K} n_k) = O((K + m)n)$ time. The last step of Algorithm 1 takes $\mathcal{O}(nK)$ time. Therefore, the time complexity of the robust K-median clustering algorithm is $O(nmKT)$.

For the robust K-means clustering algorithm, it is easy to see that the first two steps of Algorithm 2 take $\mathcal{O}(mK)$ and $\mathcal{O}(nmK)$ time, respectively. Let $|I_k| = n_k$. For given k and j, Procedure 1 takes $\mathcal{O}(n_k \log n_k)$ time to compute v_{kj}^t. Therefore, Step 3 of Algorithm 2 takes $\mathcal{O}(mn \log n)$ time since $m \sum_{k=1}^{K} n_k \log n_k \le mn \log n$ time. Note that the last step of Algorithm 2 also takes $\mathcal{O}(nK)$. Thus, the time complexity of the robust K-means clustering algorithms is $O(nm(K + \log n)T)$.

In addition, it is easy to see that both the robust K-median and robust K-means clustering algorithms have a space complexity of $O((m + n)K)$. Therefore, compared with the classical K-median and robust K-means algorithms, the proposed robust clustering algorithms consume same computation resources.

4. Numerical Experiments

In this section, we compare the proposed robust clustering algorithms with others on two data sets from the UCI machine learning repository. Section 4.1 describes the data sets and experimental setup, and Section 4.2 reports and discusses the experimental results.

4.1. Data Sets and Experimental Setup. Two widely used data sets, Iris and Seeds, are used to test the performance of the proposed algorithms. The Iris data consists of 150 objects and each object has four features of Iris flowers, including sepal length, sepal width, petal length, and petal width. The Iris data includes three clusters, Setosa, Versicolour, and Virginica, and each cluster contains 50 objects. The optimal cluster prototypes of the Iris data have been reported by Hathaway and Bezdek [34]. The Seeds data set consists of 210 kernels of three different varieties of wheat, and each kernel has seven real-valued features, including area, perimeter, compactness, length of kernel, width of kernel, asymmetry coefficient, and length of kernel groove.

We generate the missing values under the missing completely at random (MCAR) mechanism as in Hathaway and Bezdek [14] and Li et al. [20]. Specifically, we randomly select a specified percentage of components and designate them as missing. To make the incomplete data tractable, we also make sure that the following constraints are satisfied:

(1) each object retains at least one feature;

(2) each feature has at least one value present in the incomplete data set.

In addition to the Iris and Seeds data sets with artificially generated missing values, we also test the proposed algorithms on a real-world incomplete data set and the Stone Flakes data set [35], which consists of 79 eight-dimensional attribute stone flake objects in the prehistoric era. These objects belong to three different historic ages. The Stone Flakes data set is incomplete and there are 6 incomplete objects with 10 missing feature values.

Table 1: Performance of different K-median algorithms on the IRIS data.

%	Misclassification rate						Prototype error					
	WDS	PDS	NPS	RKM1			WDS	PDS	NPS	RKM1		
				0.05	0.10	0.15				0.05	0.10	0.15
0	17.2	17.2	17.2	17.2	17.2	17.2	0.126	0.126	0.126	0.126	0.126	0.126
5	18.2	20.9	18.1	16.7	16.5	17.3	0.213	0.322	0.217	0.165	0.160	0.178
10	20.1	23.0	19.8	17.3	17.1	18.5	0.263	0.419	0.283	0.218	0.221	0.249
15	24.9	26.3	24.2	19.2	19.3	20.2	0.488	0.599	0.493	0.563	0.553	0.572
20	25.1	28.2	26.5	21.6	22.1	23.6	2.319	2.773	2.185	1.391	1.508	1.857

Table 2: Misclassification rates of different K-median algorithms on the Seeds data.

%	WDS	PDS	NPS	RKM1		
				0.05	0.10	0.15
0	10.62	10.62	10.62	10.62	10.62	10.62
5	11.76	13.48	12.43	10.14	10.23	11.73
10	11.24	21.81	14.95	12.24	11.67	12.49
15	13.38	22.67	17.05	12.81	12.65	13.35
20	17.86	32.48	16.05	14.38	14.19	15.94

Li et al. [20] use the q nearest neighbors to construct intervals for missing feature values and, from their numerical experiments, $q = 6$ is a good choice. To further test the impact of the interval size on the clustering performance of the proposed robust clustering algorithms, the interval for the missing value x_{ij} is constructed as $[(1-\theta)\overline{x}_{ij}, (1+\theta)\overline{x}_{ij}]$, where $\overline{x}_{ij}$ is estimated by the q nearest neighbors and $\theta \in (0, 1)$.

4.2. Results and Discussion. We first test and compare the performance of the proposed robust K-median (labelled "RKM1") on both Iris and Seeds data sets under different missing rates from 0% to 20%. The classical K-median algorithms have also been modified based on WDS, PDS, and NPS to handle incomplete data sets. Since the performance of K-median algorithm depends on the initial cluster prototypes, we repeat each algorithm 100 times and report the averaged performance.

Tables 1 and 2 report the averaged performance of different K-median algorithms on the incomplete Iris and Seeds data, respectively. The first column in each table gives the missing rate. The second to seventh columns give the averaged misclassification rates by comparison with the true clustering result, where the fifth to seventh columns correspond to the RKM1 algorithms with different values of θ ranging from 0.05 to 0.15. In Table 1, the eighth to thirteenth columns give the averaged cluster prototype errors of different algorithms, which are calculated by

$$\left\|V^{*} - \widetilde{V}\right\|_{1} = \sum_{k=1}^{K}\sum_{j\in I}\left|v_{kj}^{*} - \widetilde{v}_{kj}\right|, \tag{15}$$

where V^{*} represents the cluster prototypes given by a certain K-median algorithm and $\widetilde{V}$ is the actual cluster prototypes of the Iris data set without missing values. Since the actual cluster prototypes of the Seeds data set are unknown, such results are not reported in Table 2.

From Tables 1 and 2, we have the following observations.

(1) When there is no missing value, that is, the missing rate is equal to zero, all K-median algorithms give the same results. As the missing rate increases, in most cases, both the misclassification rate and prototype error of all algorithms become larger.

(2) When the missing rate is small, the missing data have little adverse effect on the performance of the proposed RKM1. For example, the misclassification rate of RKM1 when the missing rate is around 5% is even smaller than that of RKM1 when the missing rate is zero.

(3) When the missing rate is large, compared with the WDS, PDS, and NPS based K-median algorithms, RKM1 provides clustering results with lower numbers of misclassification and prototype errors.

(4) Experimental results also show that the interval size affects the performance of RKM1. Specifically, as the value of θ increases from 0.05 to 0.15, for most cases, the misclassification rate of RKM1 first decreases and then increases. However, when the missing rate is high (20%), RKM1 with a small value of θ provides the best clustering performance.

The proposed robust K-means algorithm (labelled "RKM2") is also tested on both Iris and Seeds data sets and compared with the WDS, PDS, and NPS based K-means algorithms. Tables 3 and 4 report the averaged performance of these algorithms by repeating each algorithm 100 times.

Tables 3 and 4 also validate the robustness of the proposed RKM2 against the missing values. When there are missing values, RKM2 provides robust cluster results with smaller misclassification rate and prototype error compared with the WDS, PDS, and NPS based K-means algorithms. For example, when the missing rate is 5%, the misclassification rate given by RKM2 with $\theta = 0.10$ on the Seeds data set is only 10.34%, while the best misclassification rate given by other K-means algorithms is 12.10%. The impact of the interval size on the performance of RKM2 is similar to that of RKM1; that is, for most cases the RKM2 with $\theta = 0.10$ provides the best clustering performance in terms of both misclassification rate and prototype error.

Table 3: Performance of different K-means algorithms on the IRIS data.

%	Misclassification rate						Prototype error					
	WDS	PDS	NPS	RKM2			WDS	PDS	NPS	RKM2		
				0.05	0.10	0.15				0.05	0.10	0.15
0	17.8	17.8	17.8	17.8	17.8	17.8	0.165	0.165	0.165	0.165	0.165	0.165
5	19.2	21.5	19.1	17.1	17.4	18.3	0.243	0.147	0.238	0.485	0.513	0.626
10	21.3	23.0	20.1	18.1	18.3	19.2	0.193	0.208	0.316	0.761	0.729	0.831
15	25.1	26.1	24.8	21.2	22.5	23.7	0.52	0.637	0.496	1.653	1.721	1.796
20	25.8	27.0	26.7	24.8	24.4	25.6	2.641	2.871	2.373	2.587	2.673	2.639

Table 4: Misclassification rates of different K-means algorithms on the Seeds data.

%	WDS	PDS	NPS	RKM2		
				0.05	0.10	0.15
0	10.76	10.76	10.76	10.76	10.76	10.76
5	12.10	13.29	12.71	10.62	10.34	11.26
10	11.05	22.62	15.38	13.67	13.28	14.61
15	13.48	23.67	17.38	14.33	13.97	15.56
20	19.00	35.86	16.95	15.81	15.16	16.33

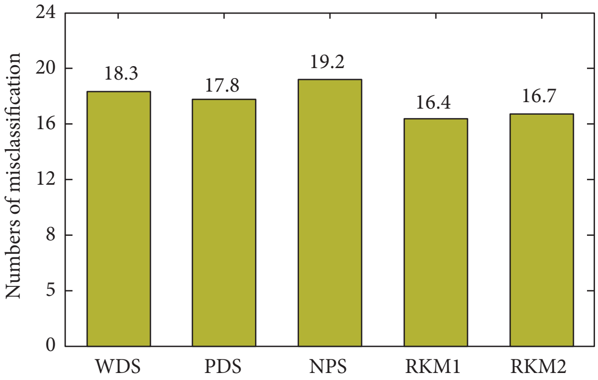

Figure 1: Numbers of misclassification of different algorithms on the Stone Flakes data set.

Finally, we test the performance of the proposed robust clustering algorithm on a real-world incomplete data set, the Stone Flakes data set. From the above discussion, we set $\theta = 0.10$ for both RKM1 and RKM2. Figure 1 demonstrates the numbers of misclassification of different algorithms. From Figure 1, we see that RKM1 provides the lowest misclassification rate and RKM2 provides the second best performance.

5. Conclusion

This paper considers the clustering problem for incomplete data. To reduce the effect of missing values on the performance of clustering results, this paper represents the missing values by interval data and introduces the concept of robust cluster objective function, which is defined as the worst-case cluster objective function when the missing values vary in the constructed intervals. Then, we propose a robust clustering model which aims at minimizing the robust cluster objective function. Robust K-median and K-means algorithms are designed to solve the proposed robust clustering problem. The time complexity of the robust K-median and K-means clustering algorithms is $O(nmKT)$ and $O(nm(K + \log n)T)$, respectively. Numerical experiments on both artificially generated and real-world incomplete data sets show that the proposed algorithms are robust against the missing data and provide better clustering performance by comparison with the existing WDS, PDS, and NPS based K-median and K-means algorithms.

Both K-median and K-means algorithms solve clustering incomplete data with hard constraints; that is, each object only belongs to one cluster. To solve clustering incomplete data with soft constraints, we will further study the robust fuzzy K-median and K-robust clustering algorithms in the future.

Competing Interests

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

Acknowledgments

This work was supported by the Major Program of the National Natural Science Foundation of China under Grant nos. 41427806 and 61503211, the Natural Science Foundation of Beijing under Grant 9152002, and the Project of China Ocean Association under Grant DYXM-125-25-02.

References

[1] B. M. Marlin, *Missing data problems in machine learning [Ph.D. thesis]*, University of Toronto, Toronto, Canada, 2008.

[2] G. S. Sebestyen, *Decision-Making Processes in Pattern Recognition*, ACM Monograph Series, 1962.

[3] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 39, no. 1, pp. 1–38, 1977.

[4] J. K. Dixon, "Pattern recognition with partly missing data," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 10, pp. 617–621, 1979.

[5] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1988.

[6] X. Zhou, R. Zhao, F. Yu, and H. Tian, “Intuitionistic fuzzy entropy clustering algorithm for infrared image segmentation,” *Journal of Intelligent & Fuzzy Systems*, vol. 30, no. 3, pp. 1831–1840, 2016.

[7] H. P. Lai, M. Visani, A. Boucher, and J.-M. Ogier, “Unsupervised and interactive semi-supervised clustering for large image database indexing and retrieval,” *Fundamenta Informaticae*, vol. 130, no. 2, pp. 201–218, 2014.

[8] L. Zhang, W. Lu, X. Liu, W. Pedrycz, and C. Zhong, “Fuzzy c-means clustering of incomplete data based on probabilistic information granules of missing values,” *Knowledge-Based Systems*, vol. 99, pp. 51–70, 2016.

[9] O. Troyanskaya, M. Cantor, G. Sherlock et al., “Missing value estimation methods for DNA microarrays,” *Bioinformatics*, vol. 17, no. 6, pp. 520–525, 2001.

[10] S. Miyamoto, O. Takata, and K. Umayahara, “Handling missing values in fuzzy c-means,” in *Proceedings of the 3rd Asian Fuzzy Systems Symposium*, pp. 139–142, Masan, Korea, June 1998.

[11] E. Acuna and C. Rodriguez, “The treatment of missing values and its effect on classifier accuracy,” in *Classification, Clustering, and Data Mining Applications*, pp. 639–647, Springer, New York, NY, USA, 2004.

[12] A. Farhangfar, L. Kurgan, and J. Dy, “Impact of imputation of missing values on classification error for discrete data,” *Pattern Recognition*, vol. 41, no. 12, pp. 3692–3705, 2008.

[13] P. Saravanan and P. Sailakshmi, “Missing value imputation using fuzzy possibilistic c means optimized with support vector regression and genetic algorithm,” *Journal of Theoretical and Applied Information Technology*, vol. 72, no. 1, pp. 34–39, 2015.

[14] R. J. Hathaway and J. C. Bezdek, “Fuzzy c-means clustering of incomplete data,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 31, no. 5, pp. 735–744, 2001.

[15] H. Timm and R. Kruse, “Fuzzy cluster analysis with missing values,” in *Proceedings of the IEEE Conference of the North American Fuzzy Information Processing Society (NAFIPS ’98)*, pp. 242–246, Pensacola Beach, Fla, USA, 1998.

[16] T. Shibayama, “A pca-like method for multivariate data with missing values,” *Japanese Journal of Educational Psychology*, vol. 40, no. 2, pp. 257–265, 1992.

[17] K. Honda and H. Ichihashi, “Linear fuzzy clustering techniques with missing values and their application to local principal component analysis,” *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 2, pp. 183–193, 2004.

[18] D.-Q. Zhang and S.-C. Chen, “Clustering incomplete data using kernel-based fuzzy c-means algorithm,” *Neural Processing Letters*, vol. 18, no. 3, pp. 155–162, 2003.

[19] M. Sadaaki, I. Hidetomo, and H. Katsuhiro, *Algorithms for Fuzzy Clustering: Methods in C-means Clustering with Applications*, Springer, Berlin, Germany, 2008.

[20] D. Li, H. Gu, and L. Zhang, “A fuzzy c-means clustering algorithm based on nearest-neighbor intervals for incomplete data,” *Expert Systems with Applications*, vol. 37, no. 10, pp. 6942–6947, 2010.

[21] D. Li, H. Gu, and L. Zhang, “A hybrid genetic algorithm–fuzzy c-means approach for incomplete data clustering based on nearest-neighbor intervals,” *Soft Computing*, vol. 17, no. 10, pp. 1787–1796, 2013.

[22] B. L. Wang, L. Y. Zhang, L. Zhang, Z. H. Bing, and X. H. Xu, “Missing data imputation by nearest-neighbor trained BP for fuzzy clustering,” *Journal of Information & Computational Science*, vol. 11, no. 15, pp. 5367–5375, 2014.

[23] L. Zhang, Z. Bing, and L. Zhang, “A hybrid clustering algorithm based on missing attribute interval estimation for incomplete data,” *Pattern Analysis and Applications*, vol. 18, no. 2, pp. 377–384, 2015.

[24] G. Lanckriet, L. E. Ghaoui, C. Bhattacharyya, and M. I. Jordan, “Minimax probability machine,” in *Advances in Neural Information Processing Systems*, pp. 801–807, 2001.

[25] K. Huang, H. Yang, I. King, M. R. Lyu, and L. Chan, “The minimum error minimax probability machine,” *Journal of Machine Learning Research*, vol. 5, no. 4, pp. 1253–1286, 2004.

[26] Y. Wang, Y. Zhang, J. Yi, H. Qu, and J. Miu, “A robust probability classifier based on the modified X^2-distance,” *Mathematical Problems in Engineering*, vol. 2014, Article ID 621314, 11 pages, 2014.

[27] S. Song, Y. Gong, Y. Zhang, G. Huang, and G.-B. Huang, “Dimension reduction by minimum error minimax probability machine,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2016.

[28] T. B. Trafalis and R. C. Gilbert, “Robust support vector machines for classification and computational issues,” *Optimization Methods and Software*, vol. 22, no. 1, pp. 187–198, 2007.

[29] H. Xu, C. Caramanis, and S. Mannor, “Robustness and regularization of support vector machines,” *The Journal of Machine Learning Research*, vol. 10, pp. 1485–1510, 2009.

[30] Y. Wang, Y. Zhang, F. Zhang, and J. Yi, “Robust quadratic regression and its application to energy-growth consumption problem,” *Mathematical Problems in Engineering*, vol. 2013, Article ID 210510, 10 pages, 2013.

[31] P. Drineas, A. Frieze, R. Kannan, S. Vempala, and V. Vinay, “Clustering large graphs via the singular value decomposition,” *Machine Learning*, vol. 56, no. 1–3, pp. 9–33, 2004.

[32] D. Aloise, A. Deshpande, P. Hansen, and P. Popat, “NP-hardness of Euclidean sum-of-squares clustering,” *Machine Learning*, vol. 75, no. 2, pp. 245–248, 2009.

[33] M. Blum, R. W. Floyd, V. Pratt, R. L. Rivest, and R. E. Tarjan, “Time bounds for selection,” *Journal of Computer and System Sciences*, vol. 7, no. 4, pp. 448–461, 1973.

[34] R. J. Hathaway and J. C. Bezdek, “Optimization of clustering criteria by reformulation,” *IEEE Transactions on Fuzzy Systems*, vol. 3, no. 2, pp. 241–245, 1995.

[35] M. Lichman, *Uci Machine Learning Repository*, School of Information and Computer Sciences, University of California, Irvine, Calif, USA, 2015.

Advances in Operations Research
Advances in Decision Sciences
Journal of Applied Mathematics
Algebra
Journal of Probability and Statistics
The Scientific World Journal
International Journal of Differential Equations
International Journal of Combinatorics
Advances in Mathematical Physics
Journal of Complex Analysis
Journal of Mathematics
Mathematical Problems in Engineering
Abstract and Applied Analysis
Discrete Dynamics in Nature and Society
International Journal of Mathematics and Mathematical Sciences
Journal of Discrete Mathematics
Journal of Function Spaces
International Journal of Stochastic Analysis
Journal of Optimization