Data-missing k-means based on intra-cluster and inter-cluster distances

Jiaji Qiu
College of Mathematics and Computer Science, Zhejiang Normal UniversityJinhua
Chinaqiujiaji@zjnu.edu.cn

Huiying Xu †
College of Mathematics and Computer Science, Zhejiang Normal UniversityJinhua
Chinaxhy@zjnu.edu.cn

Xinzhong Zhu
College of Mathematics and Computer Science, Zhejiang Normal UniversityBeijing Geekplus Technology Co., Ltd,Jinhua
Chinazxz@zjnu.edu.cn

ABSTRACT

This paper proposes a method that reduces the intra-cluster distance and increases the inter-cluster distance in the k-means problem with missing data. Filling in missing data, calculating intra-cluster distances between clusters, and clustering problems are integrated into one function, and solved through loop iterations. Finally, the method is applied to 4 UCI datasets, and the results show that the method has good effect.

CCS CONCEPTS

• **Computing methodologies** → Machine learning; Learning paradigms; Unsupervised learning; Cluster analysis.

KEYWORDS

k-means, incomplete data, clustering

ACM Reference Format:
Jiaji Qiu, Huiying Xu †, and Xinzhong Zhu. 2022. Data-missing k-means based on intra-cluster and inter-cluster distances. In *2022 6th International Conference on Electronic Information Technology and Computer Engineering (EITCE 2022), October 21–23, 2022, Xiamen, China.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3573428.3573701

1 INTRODUCTION

Clustering, especially k-means method [1] [2] [3] [4] is a common data analysis method, which is widely used in many fields, including computer visions [5], pattern recognition and machine learning.

Although k-means has a wide range of application scenarios and can achieve effective results, most methods need to be based on complete data to have better results. In actual work, data may be lost due to various reasons such as sensor failure, measurement error, disk damage, etc. In addition to the method of directly deleting incomplete data, the existing research generally fills the missing data first, and then performs clustering, which is represented by the EM algorithm. This type of method has obvious shortcomings: because the two steps of clustering and filling are separated, the filled data cannot be effectively used for the clustering task, so the clustering effect completely depends on the original filled data. Therefore, the filling effect is often not good. Wang et al. [6] proposes a method that can unify filling and clustering into one objective function, and improve the effect through the results of each clustering, so that filling and clustering can be optimized alternately. This method has achieved good results, but by studying the loss function, it is found that this method only calculates the minimization of the distance between each cluster, while ignoring the maximization of the distance between different clusters, which may cause the data at the edge to be wrong distinction. [7]

In order to solve the situation that the maximum distance between clusters is not fully considered, this paper proposes a missing k-means method based on the distance between clusters within clusters (DMKIC). DMKIC method integrates the calculation of intra-cluster distance, data filling, and clustering into a loss function, and performs the calculation through multi-step iteration, and validates the algorithm on several data sets. algorithm has better performance.

The main contributions of this paper are as follows:

(1) A loss function that can integrate the distance between clusters within clusters, filling missing data, and clustering tasks is proposed and solved through multi-step iteration.

(2) Experiments on real data datasets show that DMKIC has better performance compared with other similar methods.

2 RELATED WORKS

2.1 K-means Clustering

k-means obtains the clustering results by iteratively computing the cluster centers and the data distribution matrix. We suppose k is the number of clustering, X = {x1,x2,. . .,xn} is the dataset includes n data, U is the distribution matrix which is a 0-1 matrix have n*k elements, because of the characteristics of clustering, each row of U has one and only one element that is 1, and the rest are 0. Z = {z1,z2,. . .,zn} is the set of each cluster center.

Based on the above information, we can write the loss function of k-means:

$$J(\mathbf{U},\mathbf{Z}) = \sum_{i=1}^{n}\sum_{m=1}^{k}\mathbf{U}_{im}\|\mathbf{x}_i - \mathbf{z}_m\|^2 \quad s.t.\ \sum_{m=1}^{k}\mathbf{U}_{im} = 1 \tag{1}$$

Where $n_c = \sum_{i=1}^{n}\mathbf{U}_{ic}$ and$\mathbf{Z}_m = \frac{1}{n}\sum_{i=1}^{n}\mathbf{U}_{ic}\mathbf{x}_i$ are the number and the center of each cluster.

Equation 1) can be solved by an iterative method.

1) Randomly select k points as the initial cluster center.

EITCE 2022, October 21–23, 2022, Xiamen, China

ACM ISBN 978-1-4503-9714-8/22/10…$15.00
https://doi.org/10.1145/3573428.3573701

2) From the k centers, generate a new distribution matrix U, assigning each sample to its nearest center.

3) Calculate new cluster centers Z based on distribution matrix U and data matrix X.

By alternately performing steps 2 and 3 until the equation 1) is less than the initially set threshold, the final result of the clustering can be obtained.

2.2 K-means with incomplete data

Based on the k-means method, k-means with incomplete data introduces the missing part of the data as a variable, and iteratively optimizes. Divide each xi into two parts xi(ci) and xi(mi), which mean the complete part and missing part of each data point. The loss function of this method can be described as:

$$\begin{aligned} &J(\mathbf{U},\mathbf{Z},\mathbf{X}) = \sum_{i=1}^{n}\sum_{m=1}^{k}\mathbf{U}_{im}\|\mathbf{x}_i-\mathbf{z}_m\|^2 \\ &s.t.\ \sum_{m=1}^{k}\mathbf{U}_{im}=1, \mathbf{x}_i(c_i)=x_i^c \end{aligned} \tag{2}$$

Where $n_c = \sum_{i=1}^{n}\mathbf{U}_{ic}$ and$\mathbf{Z}_m = \frac{1}{n}\sum_{i=1}^{n}\mathbf{U}_{ic}\mathbf{x}_i$ are the number and the center of each cluster. The complete part of xi, xi(ci) will not change during iteration while the missing part, xi(mi) will do.

Compared with k-means, the iterative optimization of equation 2) increases the part of optimize missing data, as follows:

1) Randomly select k points as the initial cluster center.

2) From the k centers, generate a new distribution matrix U, assigning each sample to its nearest center.

3) Calculate new cluster centers Z based on distribution matrix U and data matrix X.

4) Calculate the missing part of X based on cluster centers Z and distribution matrix U.

By alternately performing steps 2,3 and 4 until the loss function equation 2) is less than the initially set threshold, the final result of the clustering can be obtained.

3 DATA-MISSING K-MEANS BASED ON INTRA-CLUSTER AND INTER-CLUSTER DISTANCES

On the basis of equation 2), we add the optimization of maximizing the distance between different clusters, so that the data at the edge can also obtain a better clustering effect.

3.1 Function

The loss function for clustering is as follows:

$$\begin{aligned} &J(\mathbf{U},\mathbf{Z},\mathbf{X}) = \sum_{i=1}^{n}\sum_{m=1}^{k}\mathbf{U}_{im}\|\mathbf{x}_i-\mathbf{z}_m\|^2 \\ &-\lambda\sum_{i=1}^{n}\sum_{m=1}^{k}\mathbf{U}^*_{im}\|\mathbf{x}_i-\mathbf{z}_m\|^2 \\ &s.t.\ \sum_{m=1}^{k}\mathbf{U}_{im}=1, \mathbf{x}_i(c_i)=x_i^c \end{aligned} \tag{3}$$

Where U* im is a matrix equals to 1n*k-Uim, λ is a preset hyper-parameter which weight the intra-cluster and inter-cluster distance. When λ is set to 0, the algorithm degenerates to k-means with incomplete data method.

3.2 Optimization

In order to facilitate the solution, we rewrite equation 3) into the following form:

$$\begin{aligned} &J(\mathbf{U},\mathbf{Z},\mathbf{X}) = (1+\lambda)\sum_{i=1}^{n}\sum_{m=1}^{k}\mathbf{U}_{im}\|\mathbf{x}_i-\mathbf{z}_m\|^2 \\ &-\sum_{i=1}^{n}\sum_{m=1}^{k}\mathbf{U}_{im}\|\mathbf{x}_i-\mathbf{z}_m\|^2 \\ &s.t.\ \sum_{m=1}^{k}\mathbf{U}_{im}=1, \mathbf{x}_i(c_i)=x_i^c \end{aligned} \tag{4}$$

1) Randomly select k points as the initial cluster center.

2) Update U with Z and X fixed.

According to the characteristics of the distribution matrix U, set Uim as 1 if xi is closest to zim than other z, while set other u in same row as 0.

3) Update Z with U and X fixed.

Taking the derivative of equation 4) with respect to Z, we get:

$$\begin{aligned} &\frac{\partial J(\mathrm{U,X,Z})}{\partial \mathbf{z}_m} = 2(1+\lambda)\sum_{i=1}^{m}\mathbf{U}(\mathbf{z}_m-\mathbf{x}_i) \\ &-2\lambda\sum_{i=1}^{m}(\mathbf{z}_m-\mathbf{x}_i) \end{aligned} \tag{5}$$

Let equation 5) =0, we can get:

$$\mathbf{z}_m = \frac{(1+\lambda)\sum_{i=1}^{n}\mathbf{U}_m\mathbf{x}_i - \lambda\sum_{i=1}^{n}\mathbf{x}_i}{(1+\lambda)\sum_{i=1}^{n}\mathbf{U}_m - \lambda n} \tag{6}$$

4) Update X with U and Z fixed.

Divide each x into 2 parts: the complete part xi(ci) and the missing part xi(mi). Keep the complete part fixed during optimization. We can rewrite equation 4) as:

$$\begin{aligned} &J(\mathbf{U},\mathbf{Z},\mathbf{X}) = (1+\lambda)\sum_{i=1}^{n}\sum_{m=1}^{k}\mathbf{U}_{im}\|\mathbf{x}_i(c_i)-\mathbf{z}_m(c_i)\|^2 \\ &+(1+\lambda)\sum_{i=1}^{n}\sum_{m=1}^{k}\mathbf{U}_{im}\|\mathbf{x}_i(m_i)-\mathbf{z}_m(m_i)\|^2 \\ &-\sum_{i=1}^{n}\sum_{m=1}^{k}\mathbf{U}_{im}\|\mathbf{x}_i(c_i)-\mathbf{z}_m(c_i)\|^2 \\ &-\sum_{i=1}^{n}\sum_{m=1}^{k}\mathbf{U}_{im}\|\mathbf{x}_i(m_i)-\mathbf{z}_m(m_i)\|^2 \\ &s.t.\ \sum_{m=1}^{k}\mathbf{U}_{im}=1, \mathbf{x}_i(c_i)=x_i^c \end{aligned} \tag{7}$$

Because the complete part of x is constant, equation 7) can be rewrite as:

$$\begin{aligned} &J(\mathbf{U},\mathbf{Z},\mathbf{X}) = \\ &(1+\lambda)\sum_{i=1}^{n}\sum_{m=1}^{k}\mathbf{U}_{im}\|\mathbf{x}_i(m_i)-\mathbf{z}_m(m_i)\|^2 \\ &-\sum_{i=1}^{n}\sum_{m=1}^{k}\mathbf{U}_{im}\|\mathbf{x}_i(m_i)-\mathbf{z}_m(m_i)\|^2 \\ &s.t.\ \sum_{m=1}^{k}\mathbf{U}_{im}=1 \end{aligned} \tag{8}$$

Taking the derivate of equation 8) with respect to X, we get:

$$\frac{\partial J(\mathbf{U},\mathbf{X},\mathbf{Z})}{\partial \mathbf{x}(m_i)} = 0 \tag{9}$$

$$\mathbf{x}_i(m_i) = \sum_{m=1}^{k}\mathbf{U}_{im}\frac{(1+\lambda)\mathbf{z}_m - \lambda\sum_{m=1}^{k}\mathbf{z}_m}{1+\lambda-\lambda n}(m_i) \tag{10}$$

Table 1: Dataset Information

Dataset	Samples	Dimensions	Cluster number
Wine	178	13	3
Iris	150	4	3
Glass	214	9	2
Ovarian	216	100	2

We can fill in the corresponding positions mi with the values computed for the missing positions in each iteration by equation 10).

4 EXPERIMENT

We test the proposed algorithm on some UCI dataset. They are wine, iris, glass and ovarian, which can be downloaded on UCI website. Details of them are listed in Table 1. We will illustrate the experimental results through the experimental setup, the selection of evaluation metrics, the study of parameters, and the final performance.

4.1 Compared Algorithm

DKMIC is an algorithm calculate intra-cluster and inter-cluster distances algorithm on incomplete data k-means clustering. To show the effect of it, we have chosen several methods to compare with it, including,

K-means with incomplete data: A method of iterative clustering by fusing padding and clustering.[6]

KNN-Filling: Calculate k-nearest neighbors' mean value to fill the missing part. [8]

Zero-Filling: Standardizes the incomplete data and fill zero into missing part.

Mean-Filling: Fill the missing part with mean values.[9]

Expectation Maximum: Fill the missing part by estimating the model parameters.[10]

4.2 Experiment Setup

In all our experiments, it is assumed that the true number of clusters is known. We use clustering accuracy (ACC), normalized mutual information (NMI) and F-score to evaluate the clustering performance of each algorithm. We repeat 100 times for each experiment to reduce the effect by random initialization. For our proposed algorithm DKMIC, we set the hyperparameter λ as [0.01:0.01:0.20] and the missing rate is 10%-60%.

4.3 Experiment Result

The ACC, NMI and F-score of 10% missing rate of 4 datasets have been listed on Table 2.

Our proposed method DKMIC has achieved nice effect on the 4 datasets. DKMIC outperforms the KWID method by 3.3%, 5.7%, 4.8%, 1.7% of ACC, 7.2%, 4.5%, 2.2%, 7.9% of NMI, 1.7%, 4.1%, 4.5%, 2.5% of F-score on 4 datasets. On all datasets, our proposed method DKMIC show better effect than KWID.

From Figure 1, Figure 2, Figure 3 and Figure 4, we can find that in the same dataset with different missing rates, our method can

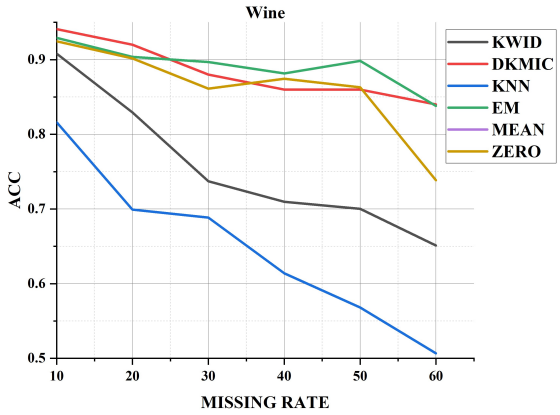

Figure 1: Result of Wine

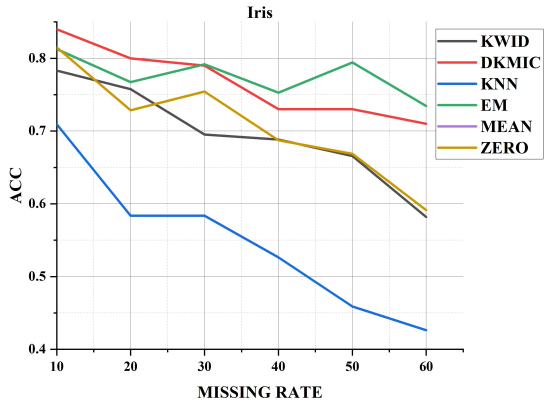

Figure 2: Result of Iris

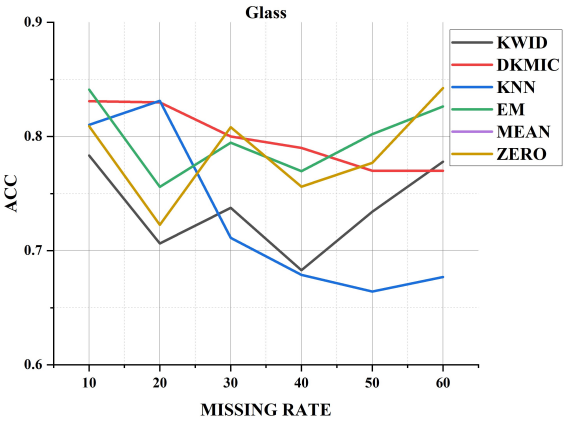

Figure 3: Result of Glass

Table 2: Result of 10% Missing Rate

	KWID	KNN	ZERO	MEAN	EM	DKMIC
	ACC					
Wine	90.8	81.6	92.5	92.5	92.9	94.1
Iris	78.3	70.1	81.5	81.5	81.3	84
Glass	78.3	81	80.9	80.9	83.2	83.1
Ovarian	84.2	86.1	85.1	85.1	85.5	85.9
	NMI					
Wine	76.2	62.2	79.7	79.7	81.4	83.4
Iris	60.5	47.6	63.6	63.7	64.6	65
Glass	25	19.8	25.1	25.1	28.2	27.9
Ovarian	41	49.2	47.3	47.3	48	48.9
	F-SCORE					
Wine	91.3	82.2	92.8	92.9	93.1	93
Iris	80.2	73.3	82.8	82.8	81.9	84.3
Glass	79.7	80.2	80.8	81	83.2	84.2
Ovarian	84.1	86.1	85.1	85.1	85.5	86.6

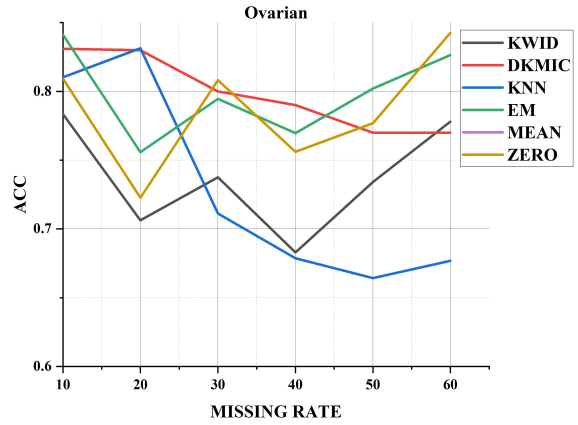

Figure 4: Result of Ovarian

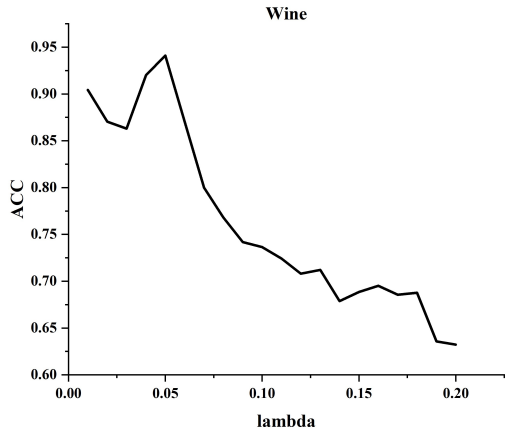

Figure 5: ACC of Wine by lambda

maintain a certain robustness and outperform other methods on the whole.

The hyperparameter lambda is used to adjust the role of intra-cluster and inter-cluster distances in the entire clustering process. The selection of lambda should not be too large, otherwise it will easily lead to function divergence. From Figure 5, taking the wine data set as an example, as the lambda increases, the performance of the algorithm DKMIC increases, and reaches the highest value when lambda = 0.05, and then shows a cliff-like decline, indicating that when the lambda is too large, the wine data set will make the algorithm The performance is unstable, therefore, the algorithm with lambda=0.05 will tend to be stable.

5 CONCLUSION

In clustering problems, most methods usually only consider the minimum distance between sample points within a cluster, while ignoring the maximum distance between different clusters. We propose a k-means method that can fuse the inter-cluster distance and missing data imputation within the cluster, and fuse them into a loss function iterative optimization, so as to better distinguish the samples located at the edge of the cluster, and in the four Experiments were carried out on the dataset and showed good results.

In future work, we will try to introduce sample dimension weights to adjust the importance of different dimension features of samples, and try to reduce the use of hyperparameters to increase the generality of the algorithm.

ACKNOWLEDGMENTS

This work was supported by the Natural Science Foundation of China (project no. 61976196), Outstanding Talents of “Ten Thousand Talents Plan” in Zhejiang Province (project no. 2018R51001),

Zhejiang Provincial Natural Science Foundation of China (project no. LZ22F030003).

REFERENCES

[1] Jain, A. K. 2010. Data clustering: 50 years beyond K-means. Pattern recognition letters, 31(8), 651-666.
[2] Feng, J., Zhang, Y., Yue, G., Liu, X., Su, H., & Zhang, P. F. 2018. Atherosclerotic Plaque Pathological Analysis by Unsupervised $ K $-Means Clustering. IEEE Access, 6, 21530-21535.
[3] Munir, M. U., Javed, M. Y., & Khan, S. A. 2012. A hierarchical k-means clustering based fingerprint quality classification. Neurocomputing, 85, 62-67.
[4] Peng, K., Leung, V. C., & Huang, Q. 2018. Clustering approach based on mini batch kmeans for intrusion detection system over big data. IEEE Access, 6, 11897-11906.
[5] Lin, X., & Li, C. T. 2016. Large-scale image clustering based on camera fingerprints. IEEE Transactions on Information Forensics and Security, 12(4), 793-808.
[6] 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.
[7] Wu, S., & Chow, T. W. 2004. Clustering of the self-organizing map using a clustering validity index based on inter-cluster and intra-cluster density. Pattern Recognition, 37(2), 175-188.
[8] García-Laencina, P. J., Sancho-Gómez, J. L., Figueiras-Vidal, A. R., & Verleysen, M. 2009. K nearest neighbours with mutual information for simultaneous classification and missing data imputation. Neurocomputing, 72(7-9), 1483-1493.
[9] Aste, M., Boninsegna, M., Freno, A., & Trentin, E. 2015. Techniques for dealing with incomplete data: a tutorial and survey. Pattern Analysis and Applications, 18(1), 1-29.
[10] Dempster, A. P., Laird, N. M., & Rubin, D. B. 1977. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society: Series B (Methodological), 39(1), 1-22.