International Journal of Database Theory and Application
Vol.9, No.4 (2016), pp.97-106
http://dx.doi.org/10.14257/ijdta.2016.9.4.09

An Optimizing Algorithm of Non-Linear K-Means Clustering

Chen Ning and Zhang Hongyi

Mechanical and Energy Engineering College, Jimei University
cn1972@jmu.edu.cn
School of Optoelectronic & Communication Engineering, Xiamen University of Technology
zhanghongyi@xmut.edu.cn

Abstract

Kernel K-means clustering (KKC) is an effective nonlinear extension of K-means clustering, where all the samples in the initial space are mapped into the feature space and then K-means clustering is performed based on the mapped data. However, all the mapped data are expressed by the implicit form, which causes the initial cluster centers can't be selected flexibly. Once the selected initial cluster centers aren't suitable, it tends to fall into local optimal solutions and can't guarantee stable result. Based on a standard orthogonal basis of the sub-space spanned by all the mapped data, a novel improving non-linear algorithm of KKC is presented in this paper. The novel algorithm can express the mapped data using the explicit form, which make it very flexible to select the initial cluster centers as the linear K-means clustering does. Moreover, the computational complexity of the presented algorithm is also significantly reduced compared to that of KKC. The results of simulation experiments illustrate the proposed method can eliminate the sensitivity to the initial cluster centers and simplify computational processing.

Keywords*: k-means clustering, the orthogonal basis, kernel method*

1. Introduction

As one of the core data mining techniques, clustering analysis shows widely application in many fields, such as machine learning, patter recognition, image processing *et al*. K-means clustering algorithm is one of the most popular methods for clustering analysis because its effectiveness and easy operation [1-3]. However, K-means clustering is only a linear algorithm in essence. It is unsuitable for highly complex and nonlinear data distributions. In contrast, as a nonlinear extension of K-means clustering, kernel K-means clustering (KKC) [4-6] can capture the nonlinear features of the data. The mainly idea of KKC is as followed. Using a nonlinear mapping, all the samples in the initial space are mapped into a high-dimensional or infinite-dimensional space, which is called the feature space, and then K-means clustering is performed based on the mapped samples in the feature space. Using the kernel trick, all the computation of KKC can be done by the kernel function, where the kernel function is the inner product operation of the feature space in session. So KKC can not only capture the nonlinear features of the initial data but also avoid the expression of the complex nonlinear mapping.

Unfortunately, as all the data in the feature space are expressed by the implicit form, the drawback of KKC is the initial cluster centers can't be selected flexibly. Once the initial cluster centers aren't suitable, it tends to fall into local optimal solutions and can't guarantee stable result.

In order to deal with the above problem, a novel improving nonlinear algorithm is presented in this paper. The main idea is as follows. Using a standard orthogonal basis of the sub-space spanned by all the mapped data mapped onto the feature space, all the mapped data is expressed in the explicit form. Subsequently, some existing optimizing

ISSN: 2005-4270 IJDTA

initial cluster centers algorithms of K-means clustering can be extended easily. In this paper, we employed the meliorated initial center algorithm proposed by Khan *et al* [2].

The rest of the paper is arranged as follows. In Section 2, a brief instruction to KKC is given. Section 3 elucidates the proposed optimizing algorithm of KKC, in which we will introduce in detail how to obtain the explicit form of the mapped data in the feature space. In Section 4, we demonstrate the effectiveness of out algorithm by some experiments. Finally, we give conclusions of our work in Section 5.

2. Kernel K-Means Clustering (KKC)

Let $X = \{x_1, x_2, \cdots, x_N\}$ denotes the input data set, where N denotes the number of all data. The main idea of KKC is all the samples $x_i (i = 1,2, \cdots, N)$ is firstly mapped in the feature space by a nonlinear mapping $\phi(\cdot)$ and then the K-means clustering algorithm based on the mapped samples $\phi(x_i)$ $(i = 1,2, \cdots, N)$ in the feature space. Obviously, where performing KKC means minimum Eq. (1).

$$J^{\phi} = \sum_{k=1}^{K} \sum_{i=1}^{N_k} \left\| \phi(x_i) - m_k^{\phi} \right\|^2 \tag{1}$$

Where K denotes the number of the class, $m_k^{\phi} = \frac{1}{N_k} \sum_{i=1}^{N_k} \phi(x_i)$ denotes the mean valve of the $k-th$ class and N_k is the number of the class k .

In feature space, the distance between any one mapped sample $\phi(x)$ and the mean m_k^{ϕ} can be computed in Eq. (2).

$$\left\| \phi(x) - m_k^{\phi} \right\|^2 = k(x, x) - \sum_{i=1}^{N_k} k(x, x_I) + \sum_{i,j=1}^{N_k} k(x_i, x_j) \tag{2}$$

Where $k(\cdot)$ denotes the kernel function.

Combing the kernel function and the K-means clustering, we can get the algorithm of KCC as followed.

Algorithm 1 The K-means clustering

Step1. Identifying the initial cluster centers $(m_1^{\phi}, m_2^{\phi}, \cdots, m_K^{\phi})$. In other words, K different samples are randomly selected from all mapped data $\phi(x_i)$ $(i = 1,2, \cdots, N)$ as the class centers.

Step2. According to Eq. (3), any one sample x from $X = \{x_1, x_2, \cdots, x_N\}$ is arranged respectively to $k-th$ class.

$$k \leftarrow \min_{k=1,2,\cdots K} \left\| \phi(x) - m_k^{\phi} \right\| \tag{3}$$

Step3. Updating the cluster centers $(m_1^{\phi}, m_2^{\phi}, \cdots, m_K^{\phi})$ and the J^{ϕ} according to Eq.(1). In fact, the cluster center $m_k^{\phi}(k = 1,2, \cdots, K)$ can't be obtained in feature space. So, $m_k^{\phi}(k = 1,2, \cdots, K)$ only be repeated by one mapped sample $\phi(x_k)$, where the distance sum of all the mapped samples belonging to $k-th$ class and $\phi(x_k)$ is minimum.

Step4. Repeating step2 and step3, until J^{ϕ} don't change or the change is small.

3. The Optimizing Algorithm of KKC (OKKC)

From the process of KCC presented in the second section, we known that the initial clustering center is chosen randomly from all the mapped data. Once the initial clustering center isn't suitable, the result may be a local minimum and easily cause the result unstable.

3.1. Second-Order Headings

In order to reduce the sensitive of KKC to the initial clustering centers, we presented a novel optimizing algorithm of KKC (OKKC). The main idea is as followed. Firstly, we get a standard orthogonal basis of the subspace spanned by all the mapped samples $\phi(x_i)\ (i = 1,2,\cdots,N)$. Secondly, we obtain the projection vector of each mapped samples. Lastly, we perform K-means clustering based all the projection vectors in the projection space. Because the projection vectors are expressed in the explicit form, it is very easy to extent the existing optimizing initial cluster centers algorithms of K-means clustering. An overview of the proposed optimizing algorithm is illustrated in Figure1. In Figure 1, $\{\beta_i\}_{i=1}^{r}$ denotes the standard orthogonal basis of the subspace spanned by all the mapped samples $\{\phi(x_i)\}_{i=1}^{N}$. Where r denotes the number of the basis. In the next part, we will give the detailed process about how to obtain $\{\beta_i\}_{i=1}^{r}$. Based the basis, we can get the projection vectors $Y = \{y_i\}_{i=1}^{N}$ by Eq. (4). Lastly, we perform the K-means clustering. Because y_i can be expressed in explicit form it will is very convenient to optimize the initial cluster centers using the existing optimizing initial cluster centers algorithms of K-means clustering.

$$y_i = (\beta_1, \beta_2, \cdots, \beta_r)^T \cdot \phi(x_i) \tag{4}$$

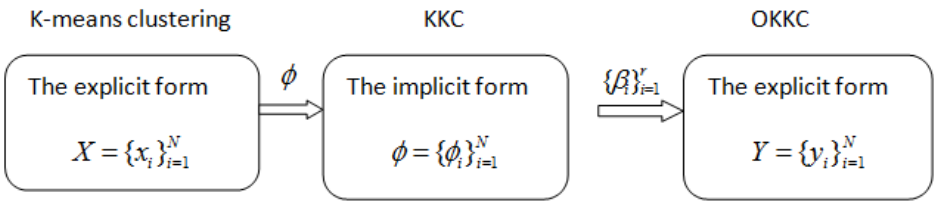

Figure 1. The Process of OKKC

In the process of OKKC, we may have noticed the question: is KKC equivalent to OKKC? The answer is apparently yes. The reason is that the distance of any two samples in the projection space is equal to that of the corresponding two samples in the feature space. This can be described in Eq. (5). Based on Eq.(4), it is easy to verify Eq.(5) is correct.

$$\|y_i - y_j\| = \|\phi(x_i) - \phi(x_j)\| \quad i, j = 1,2,\cdots,N \tag{5}$$

3.2 How to Get a Standard Orthogonal Basis

In order to get a standard orthogonal basis $\{\beta_i\}_{i=1}^{r}$ of the subspace spanned by all the mapped samples $\{\phi(x_i)\}_{i=1}^{N}$, we firstly designed a iterative algorithm to find a basis $\{\phi(x_{bi})\}_{i=1}^{r}$ and then make the orthogonalization of $\{\phi(x_{bi})\}_{i=1}^{r}$ for getting a standard

orthogonal basis $\{\beta_i\}_{i=1}^{r}$. The iterative algorithm to find a basis can be described as the follows [7].

Alogrithm2. The iterative algorithm to find a basis

Step1. Initialization Select randomly a sample x from the set $X = \{x_i\}_{i=1}^{N}$ and $k(x, x) \neq 0$. Where $k(\cdot)$ denotes the kernel function. Let $S = \{x\}, D = \{x\}, G = 1/k(x, x), t = 1.$

Step2. The condition of ending the procedure If $t = N$, then put out D and let the procedure end. Otherwise, next step.

Step3. Discrimination function Select randomly a sample $x^* \in X - S$. let $S = S \bigcup \{x^*\}$, $t = t + 1$ and verify Eq. (6) is correct or not.

$$k_{tt} - k_{st}^T G k_{st} = 0 \tag{6}$$

Where $k_{tt} = k(x^*, x^*), (k_{st})_i = k(\bar{x}_i, x^*), \bar{x}_i \in D.$

Step4. Update the procedure. If Eq. (6) is correct, go back step2. Otherwise, do the following and go to step2.

$$D = D \bigcup \{x^*\}, \quad G = \frac{1}{k_{st} - k_{st}^T G k_{st}} \begin{bmatrix} (k_{tt} - k_{st}^T G k_{st}) G + G k_{st} k_{st}^T G & -G k_{st} \\ -k_{st}^T & 1 \end{bmatrix}$$

When the above procedure stop, we get a samples set $D = \{x_{bi}\}_{i=1}^{r}$. Its mapped samples $\phi(D) = \{\phi(x_{bi})\}_{i=1}^{r}$ will be a basis of the subspace spanned by $\{\phi(x_i)\}_{i=1}^{N}$, which can be easily verified using the linear correlation theories. After that, we can orthogonalize the basis $\phi(D) = \{\phi(x_{bi})\}_{i=1}^{r}$ using Eq. (7).

$$(\beta_1, \beta_2, \cdots, \beta_r) = (\varphi(x_{b1}), \varphi(x_{b2}), \cdots \varphi(x_{br}))C \tag{7}$$

Where $C = \left(u1/\sqrt{\lambda_1}, u2/\sqrt{\lambda_2}, \cdots, ur/\sqrt{\lambda_r}\right)$ $u_i, \lambda_i (i, j = 1,2 \cdots, r)$ denotes respectively the eigenvector and the corresponding eigenvalue of the kernel matrix $K_{rr} = (k(x_{bj}, x_{bk}))_{1 \leq j,k \leq r}$.

Combined with Eq.(4) and Eq.(7), the projection vector of any mapped sample $\phi(x_i)$ can obtained by Eq.(8).

$$y_i = C^T (k(x_{b1}, x_i), k(x_{b2}, x_i), \cdots, k(x_{br}, x_i)) \quad (i = 1,2, \cdots, N) \tag{8}$$

Algorithm3. The optimizing algorithm of KKC (OKKC)

Step1. Based on algorithm 2, we get a basis $\phi(D) = \{\phi(x_{bi})\}_{i=1}^{r}$.

Step2. Computing the kernel matrix $K_{rr} = (k(x_{bj}, x_{bk}))_{1 \leq j,k \leq r}$ of the basis $\phi(D) = \{\phi(x_{bi})\}_{i=1}^{r}$ and eigenvalue decomposition of K_{rr} for getting the matrix C.

Step3. Using Eq. (8), we get the projection vectors $Y = \{y_i\}_{i=1}^{N}$.

Step4. Using the existing optimizing initial cluster centers algorithms of K-means clustering, we finish cluster process of $Y = \{y_i\}_{i=1}^{N}$. In this paper, we employed the meliorated initial center algorithm proposed by Khan *et al*.

4. Experiments

Use footnotes sparingly (or not at all!) and place them at the bottom of the column on the page on which they are referenced. Use Times New Roman 9-point type, single-spaced with 10-point interlining spacing. To help your readers, avoid using footnotes altogether and include necessary peripheral observations in the text.

Experiment1:the Bispiral dataset contains 150 data points from two classes, each data point is a two-dimensional real vector, the KKC and the proposed OKKC were applied , Figure(2) and Figure(3) show the clustering results, here, the maximum times of iteration is 100.

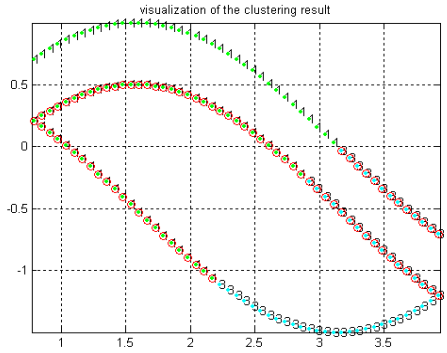

Figure 2. The Clustering Results for Bispiral Using Original KKC Algorithm

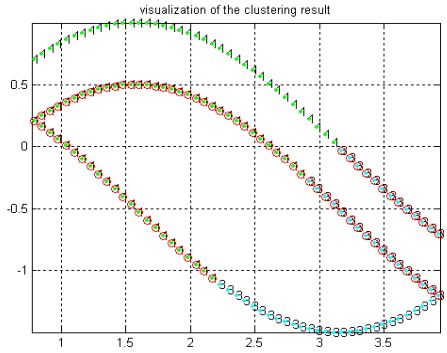

Figure 3.The Clustering Results for Bispiral Using OKKC Algorithm

From Figure(2) and Figure(3), we can see that the errors that we use the original KKC algorithm is 40, and the accuracy is only 60.784 percent; on the contrary, if the OKKC method is applied, the errors is only 3, and the relative accuracy increase to 96.0784 percent.

Experiment 2: The dataset contains 100 data points from three classes. Each class includes 50 data points. Each data point is a four-dimensional real vector, the KKC and the proposed OKKC were applied , Figure(4) and Figure(5) show the clustering results, here, the maximum times of iteration is 100.

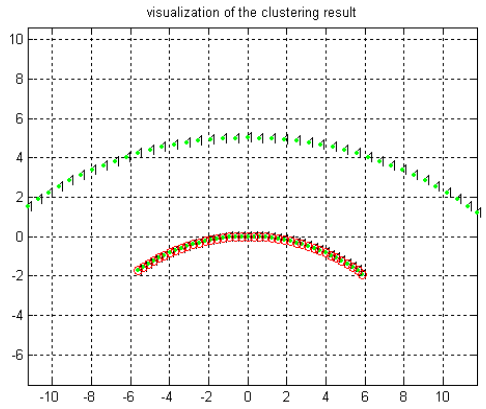

Figure 4. The Clustering Results for Dataset Using Original KKC Algorithm

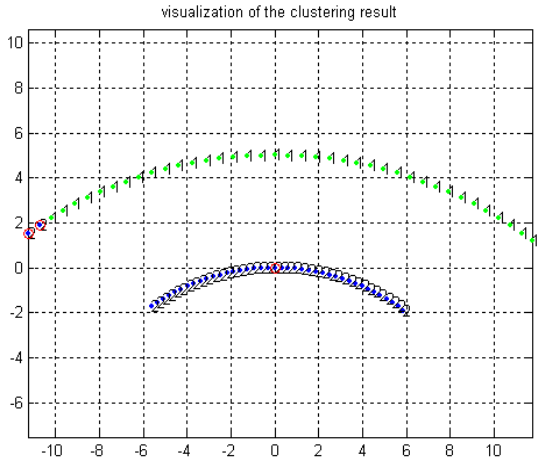

Figure 5. The Clustering Results for Dataset Using OKKC Algorithm

From Figure(4) and Figure(5), we can see that the errors that we use the original KKC algorithm is 41, and the accuracy is only 49.3827 percent; on the contrary, if the OKKC method is applied, the errors is only 3, and the relative accuracy increase to 96.2963 percent.

Experiment 3: The dataset contains 100 data points from three classes. Each class includes 50 data points. Each data point is a four-dimensional real vector, the KKC and the proposed OKKC were applied , Figure(6) and Figure(7) show the clustering results, here, the maximum times of iteration is 100.

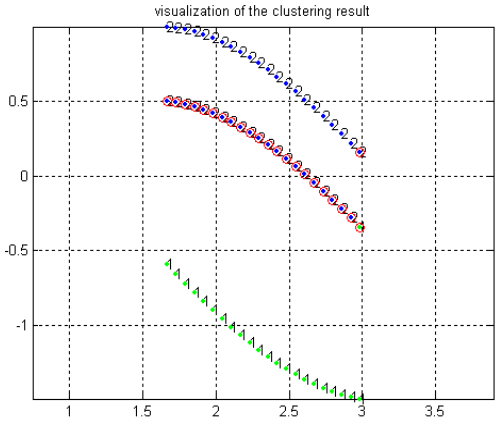

Figure 6. The Clustering Results for Bispiral Using the Original Algorithm

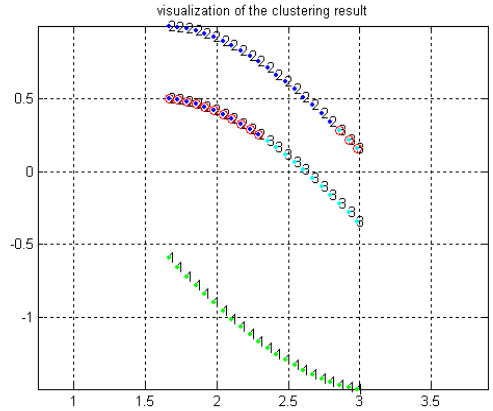

Figure 7. The Clustering Results for Bispiral Using the OKKC Algorithm

From Figure(6) and Figure(7), we can see that the errors that we use the original KKC algorithm is 23, on the contrary, if the OKKC method is applied, the errors is only 13.

Experiment4: To validate the performance of the presented algorithm---OKCC, three standard subsets (Iris, Balance-scale, and Wine) from UCI are employed in our experiments. To each dataset, we perform experimental operation for five times respectively. Table 1 shows the experimental results. In every experiment, the initial centers are randomly selected in KKC. On the other hand, the initial centers are optimized in OKCC (see algorithm3). In all

The experiments, the Gaussian kernel function $k(x, y) = \exp(\|x - y\|^2 / 0.1 \times N)$, where N denotes the number of all the samples. In fact, the optimized initial centers haven't any practical information, so we only list the number of a basis and skip the concrete value of the initial centers.

Table 1. The Cluster Results Based on KKC and OKKC

Algorithm		Iris		Balance-scale		Wine	
		Initial center	accuracy rate	Initial center	accuracy rate	Initial center	accuracy rate
KKC	1	35,6,21	93.2%	616,496,589	73%	35,191,34	61.2%
	2	47,5,42	84.1%	316,128,345	78.4%	15,89,131	79.2%
	3	12,49,134	85.3%	21,151,43	62.7%	26,112,169	84%
	4	37,71,84	73%	431,322,125	56%	110,46,96	76%
	5	24,9,43	90.5%	232,145,284	72%	89,172,59	89.1%
Average accuracy rate			85.22%		68.42%		77.9%
OKCC		r=4	94%	r=12	81.3%	r=21	87.3%

As can be seen from Table1, the accuracy rate of KKC is very unstable, the reason is that KKC is very sensitive to the initial centers. However, the proposed algorithm---OKKC is very stable and the accuracy rate is much higher than that of KKC, which verify KKC is very effective.

5. Conclusion

To reduce the sensitive to the initial cluster centers of kernel K-means clustering (KKC), an optimizing algorithm is proposed in this paper. Firstly , we designed a iterative algorithm to find a standard orthogonal basis of the subspace spanned by all the mapped data in the feature space, and then perform the linear K-means clustering algorithm based the projection vectors in the projection space. Because the projections vectors are expressed in the explicit form, it is easy to employ the existing optimizing cluster center algorithm of K-means clustering. In the experimental section, its performance of OKKC is verified based on the three subsets form UCI. Those experiments shown the accuracy rate of the proposed algorithm is much higher than that of KKC and its stability also be guaranteed.

Finally, as the extension of the kernel method, the accuracy of OKKC is still sensitive to the kernel parameter. How to reduce the sensitive of the kernel parameter still needs the further investigation.

References

[1] K. Tapas, M. M. David and S. N. Nathan, "An Efficient k-Means Clustering Algorithm: Analysis and Implementation", IEEE transactions on pattern analysis and machine intelligence, vol. 24, no. 7, **(2004)**, pp. 881-892.
[2] S. Z. Khan and A. Ahmad, "Cluster center initialization algorithm for K-means clustering", Pattern recognition letters, vol. 25, no. 11, **(2004)**, pp. 1293-1302.
[3] M. P. Cosmin, C. M. Marian and M. Mihai, "An optimized version of the K-means clustering algorithm", Proceedings of the 2014 Federated Conference on Computer Science and Information Systems, vol. 5, no. 2, **(2014)**, pp. 695-699.
[4] T. Grigorios and L. Aristidis, "The global kernel K-means clustering algorithm", IEEE World Congress on Computational Intelligence. Hong Kong, **(2008)**, pp. 1977-1984.
[5] L. Zhang, W. D. Zhou and L. C. Jiao, "Kernel Clustering Algorithm", Chinese Journal of Computers, vol. 25, no. 6, **(2002)**, pp. 587-590.
[6] F. Maurizio, C. Francesco and M. Francesco, "A survey of kernel and spectral methods for clustering", Pattern Recognition, vol. 41, no. 1, **(2008)**, pp. 176-190.
[7] F. Zhao and J. Y. Zhang, "Fast algorithm about KPCA. Control and Decision (Chinese)", vol. 22, no. 9, **(2007)**, pp. 1044-1057.

Authors

Chen Ning, Associate Professor in Mechanical and Energy Engineering College of Jimei University. His research focuses on the field of Robot Vision.

Zhang Hongyi, Professor in School of Optoelectronic & Communication Engineering, Xiamen University of Technology. Her research focuses on the field of pattern recognition.