

## Time Series Clustering Ensemble Algorithm Based on Locality Preserving Projection

---

**Xue Liu**<sup>12</sup>

*Information & Technology College, Hebei University of Economics & Business, 050061 Shijiazhuang, China*

*E-mail: 924755771@qq.com*

**Xiaoqing Weng**

*Information & Technology College, Hebei University of Economics & Business, 050061 Shijiazhuang, China*

*E-mail: xqweng@heuet.edu.cn*

The time series clustering is one of the important research contents in the time series data mining. Since the dimension of time series is common high, the performance of direct raw time series data clustering is not ideal. How to improve the clustering performance of time series is the main research point of this paper. Firstly, use Locality Preserving Projection (LPP) to time series samples for dimensionality reduction; secondly, carry out clustering ensemble to the low-dimension data; finally, compare the clustering performance with the existing methods such as Principal Component Analysis (PCA) and Piecewise Aggregate Approximation (PAA). Experiment results show that the proposed method is superior to the compared methods.

*CENet2015*

*12-13 September 2015*

*Shanghai, China*

---

<sup>1</sup>Speaker

<sup>2</sup>Corresponding Author

## 1. Introduction

Time series [1], usually a kind of data, is of high dimensional and changes with time. It is very easy to be influenced by the environmental factors and there is a certain noise. The time series clustering is one of the important tasks in the time series mining.

Su et al. showed that it would be feasible to reduce the dimension of a single variable time series by using PCA and proposed the method of time series clustering based on PCA [2]. Li et al. used PAA to achieve the feature extraction and data dimension reduction of time series [3], and used dynamic time warping (DTW) to calculate the distance; however, the time complexity of DTW was relatively high. These methods are only individual sample concerned without considering the intrinsic relationship between samples. PCA considers the intrinsic relationship, but it belongs to the linear dimension reduction method, and the real datasets often have the nonlinear characteristics. LPP is a linear dimension reduction method, which is the linear approximation of Laplacian feature maps [4]. It clearly considers the manifold structure of the data, which, upon dimension reduction can be the best to keep the local neighborhood information of the raw dataset and considers the intrinsic relationship between each time series samples.

As the single algorithm brings about the problem of instability, it now tends to ensemble multiple results. AdaBoost is designed for supervised learning where the labels are known; different classifiers are trained on the same training set, and then combine the classifiers together to form a stronger final classifier. Because of lack of training labels, the design of ensemble methods for unsupervised learning is more difficult than the supervised learning. In this paper, the ensemble method is designed for unsupervised learning where the labels are unknown. Use the mutual information to represent the weights of each member of the cluster and the ensemble method uses the weighted voting; therefore, the clustering performance has been improved.

## 2. Relevant Knowledge

### 2.1 Locality Preserving Projection

As to the LPP algorithm, this paper gives a simple introduction [5]. Let the raw dataset  $X = \{x_1, x_2, \dots, x_n\} \subset \mathbb{R}^l$ . LPP's purpose is to find a transformation matrix  $A$ , which uses  $A$  to  $X$ . when  $X$  is mapped into a low dimensional dataset  $Y = \{y_1, y_2, \dots, y_n\} \subset \mathbb{R}^d$ , ( $d \ll l$ ), it can preserve the local structure of the raw dataset.

The variable  $a$  can be obtained by solving the minimization problem:

$$\min_a \sum_{i,j=1}^N (a^T x_i - a^T x_j)^2 S_{ij} \quad (2.1)$$

The similarity matrix  $S$  is defined as follows: if  $x_i$  is in the  $k$  nearest neighbor of  $x_j$  or  $x_j$  is in the  $k$  nearest neighbor of  $x_i$ ,  $S_{ij} = \exp(-\|x_i - x_j\|^2/t)$ , under other conditions,  $S_{ij} = 0$ ,  $t$  is an appropriate constant and  $S_{ij}$  is a local structure to evaluate  $X$ .

The transformation vector  $a$  can be obtained by solving the general eigenvalues:

$$XLX^T a = \lambda XDX^T a \quad (2.2)$$

Where the matrix  $D$  is a diagonal matrix,  $D_{ii} = \sum_j S_{ji}$ ,  $L = D - S$  is the Laplacian matrix. Let  $A = [a_0, a_1, \dots, a_{d-1}]$ , the eigenvalues of sort are  $0 \leq \lambda_0 \leq \dots \leq \lambda_{d-1}$ .

Thus, the vector  $y$  is obtained from the dimension reduction:  $y_i = A^T x_i$ ; where  $y_i$  is the  $d$  dimension vector and  $A$  is the  $l \times d$  transform matrix.

### 2.2 Weights of Clustering Members based on Mutual Information

Let sample set  $X = \{X_1, X_2, X_3, \dots, X_n\}$ , the elements in  $X$  are classified into  $k$  clusters, which can be represented by a cluster member  $p = \{p_1, p_2, \dots, p_n\}$ , and  $p_i \in \{1, 2, \dots, k\}$  is the cluster label. In this paper, mutual information is used to represent the weight of each cluster member [6].  $X$  is

classified into  $k$  clusters, getting cluster members  $P^a, P^b$ , respectively represented by  $\{P_1^a, P_2^a, \dots, P_k^a\}$  and  $\{P_1^b, P_2^b, \dots, P_k^b\}$ . Suppose there be  $n$  objects, where  $n_i$  are in cluster  $P_i^a$ ,  $n_j$  are in cluster  $P_j^b$ , and  $n_{ij}$  are in both  $P_i^a$  and  $P_j^b$ , thus the mutual information  $\Phi^{MI}$  can be defined as:

$$\Phi^{MI}(P^a, P^b) = \frac{2}{n} \sum_{i=1}^k \sum_{j=1}^k n_{ij} \log_k \left( \frac{n_{ij} n}{n_i n_j} \right) \quad (2.3)$$

The average mutual information for each cluster member of the dataset is:

$$\beta_m = \frac{1}{t-1} \sum_{l=1, l \neq m}^t \Phi^{MI}(P^m, P^l) \quad (m=1, 2, \dots, t) \quad (2.4)$$

The bigger the  $\beta_m$ , the less unique information contained by the  $P^m$ ; therefore, the weight can be defined as this:

$$w_m = \frac{1}{\beta_m Z} \quad (m=1, 2, \dots, t) \quad (2.5)$$

Where  $Z$  is used to normalize the weights so that  $w_m > 0 (m=1, 2, \dots, t)$ , and  $\sum_{m=1}^t w_m = 1$ .

### 3. Time Series Clustering Ensemble Algorithm

In this paper, the time series clustering ensemble algorithm based on LPP consists of three steps: firstly, use the LPP algorithm to reduce the dimension of the raw dataset; secondly, run K-means algorithm to the data of dimension reduction in order to get the clustering members; finally, use weighted voting for cluster members to the clustering ensemble. In this paper, we use the classical K-means algorithm to generate the clustering members, and the ensemble function uses the weighted voting method.

The size of raw dataset is  $n$ . In this paper, each sample is considered as a point in the  $l$  dimensional space. Since  $n$  is sometimes far less than  $l$ , it will cause the matrix  $XDX^T$  to become the singular matrix [7]; therefore, to make  $XDX^T$  become a non-singular matrix, first of all, use PCA to project the dataset to the PCA subspace.

The clustering ensemble algorithm is described as follows:

Input: Data set, Data, Nearest neighbor number,  $k$ , Embedding dimension,  $d$ , Clustering number,  $M$ , The number of cluster members,  $H$

Output: clustering results

Step 1: use PCA to preprocess the dataset; remove the noise and retain the main information of the raw dataset.

Step 2: construct the adjacency graph  $G$ .

Step 3: transformation matrix  $A_{LPP}$  is  $[a_0, a_1, \dots, a_{d-1}]$ , and the order of eigenvalues are  $0 \leq \lambda_0 \leq \dots \leq \lambda_{d-1}$ .

Step 4:  $y_i = A_{LPP}^T x_i$ , get the low-dimensional matrix  $Y$ .

Step 5: use K-means to  $Y$  to cluster into  $M$  classes. Loop  $H$  times, so the number of cluster members is  $H$ .

Step 6: convert cluster labels of the cluster members.

Step 7: calculate weights for each cluster member.

Step 8: carry out clustering ensemble for  $H$  clustering members.

Algorithm analysis:

Using  $m$  to represent the number of samples, and the length of each sample is  $n$ . In Step 1, the time complexity of preprocess is  $O(m \times n^2)$  [8]. The complexity of LPP is divided into two parts: one is  $k$ , the nearest neighbor search; the other is eigenvalue problem. In  $k$  nearest neighbor search, the complexity is  $O((n+k)m^2)$ ; if the initial population about the sample with large size, we can establish index in order to speed up search of the  $k$  nearest neighbor, such as R-tree. In the eigenvalue problem, assume that the dimension is reduced to  $d$  and the complexity is  $O((n+d)n^2)$ ; therefore, the complexity of LPP is  $O((n+k)m^2 + (n+d)n^2)$  [9]. The complexity of K-means clustering is  $O(hmM)$  [10]; among them,  $h$  is the number of iterations and  $M$  is the

number of clusters. In the weighted voting, the computation time cost of the mutual information weight is negligible compared with the K-means clustering; in this sense, the complexity of clustering ensemble is  $O(HhmM)$  [6], where,  $H$  is the number of cluster members. So the complexity of this experiment is  $O((n+k)m^2+(n+d+m)n^2+HhmM)$ .

## 4. Experiment

### 4.1 Dataset Description

No.	Dataset name	Number of classes	Total number of samples	Length of the time series
1	Synthetic Control	6	600	60
2	Gun Point	2	200	150
3	CBF	3	930	128
4	Face Four	4	112	350
5	Lighting2	2	121	637
6	Lighting7	7	143	319
7	ECG	2	200	96
8	Beef	5	60	470
9	Coffee	2	56	286
10	Olive Oil	4	60	570

**Table 1:** Dataset Description

In the dimension reduction of time series, the number of input variables is the length of  $n$ , and the number of output variables is low dimensional space dimension  $d$ . In this experiment, 10 times series datasets are tested [11]. These datasets come from medical, gesture recognition, chemical, biological information and other fields. Basically, all of them have nonlinear features and are non-stationary series. Table 1 lists the main features of the 10 datasets.

### 4.2 Evaluation Scheme

The clustering performance can be measured by *micro-p* (micro-precision) [6]. In a dataset that already has known classes, suppose there be  $c$  classes,  $\{C_1, C_2, \dots, C_c\}$ . The cluster label of each sample is obtained by clustering experiment;  $a_h$  represent of the number of samples to be correctly classified into the class  $C_h$ , where,  $n$  is the number of samples of the dataset. The higher the *micro-p* value is, the better the clustering effect will be.

$$Micro-p = \frac{1}{n} \sum_{h=1}^c a_h \quad (4.1)$$

### 4.3 Performance Comparison

In Table 2, repeat each test for 10 times, and then record the average *micro-p* values. The second column gives the *micro-p* value of the K-means clustering on the raw data; the third column, the fourth column, and the fifth column give the highest *micro-p* value and the corresponding embedding space dimension of PCA, PAA and LPP directly of the K-means clustering. How to select the optimal embedding dimension is worth of further study in the future. In the experiment, we refer to the dataset sample number to select the nearest neighbor number  $k$ ; if the number of sample data is high, the number of  $k$  can be larger; if the number of dataset is small, the number of smaller neighbor is selected. When the number of cluster members is different, the *micro-p* value may be different; when compared with other methods, the value of cluster members  $H$  is from 5 to 200; In the sixth column, the highest *micro-p* value that clustering ensemble of LPP dimension reduction data and the corresponding cluster members are given, and the clustering ensemble algorithm is represented by LPPC.

In order to judge the significant difference of the average value of the results of the clustering, this paper uses the paired sample  $t$  test [12]. Use Matlab to make the experiment. The  $t$  statistic value and the probability  $p$  value are shown in Table 3.

Dataset name	Raw data	PCA	PAA	LPP	LPPC
Synthetic Control	0.643	0.679(38)	0.680(28)	0.740(26)	0.755(95)
Gun Point	0.500	0.500(1)	0.574(6)	0.745(2)	0.755(5)
CBF	0.645	0.693(1)	0.816(79)	0.713(52)	0.863(105)
Face Four	0.581	0.629(2)	0.696(192)	0.646(14)	0.764 (185)
Lighting2	0.621	0.658(21)	0.669(160)	0.670(92)	1.000(65)
Lighting7	0.509	0.523(31)	0.569(5)	0.544(5)	0.597(165)
ECG	0.727	0.745(2)	0.745(18)	0.805(1)	0.801(5)
Beef	0.498	0.515(66)	0.512(148)	0.560(1)	0.592(160)
Coffee	0.609	0.625(72)	0.629(77)	0.839(67)	1.000(45)
Olive Oil	0.738	0.808(36)	0.833(96)	0.867(2)	0.873 (40)
Mean value	0.607	0.638	0.672	0.713	0.800

**Table 2:** Comparison of Clustering Result

Pair	$t$	$p$
Raw data and LPP	-4.5140	0.0015
PCA and LPP	-2.8483	0.0191
PAA and LPP	-1.3512	0.2096
Raw data and LPPC	-5.2120	0.0006
PCA and LPPC	-4.2721	0.0021
PAA and LPPC	-3.2232	0.0104

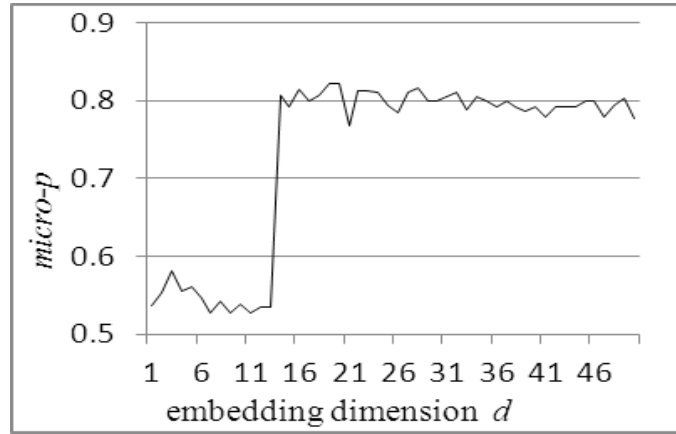
**Table 3:** T Test of Paired Sample

From Table 2 and Table 3, we can see, raw data and PCA respective with LPP's probability  $p$  value are less than 0.05, which shows that the clustering performance of LPP is significantly better than that of the two methods. The average *micro-p* value of LPP is 0.713, and the average *micro-p* value of PAA is 0.672, and the probability  $p$  value of PAA and LPP is 0.2096, more than 0.05, which shows that the clustering performance of LPP is better than that of PAA, but it is not significant.

Raw data, PCA, PAA respective with LPPC's probability  $p$  value are less than 0.05, which shows that the clustering performance of LPPC is significantly better than that of the three methods; the clustering ensemble algorithm based on LPP can improve the clustering performance.

#### 4.4 Effect of Parameters on The Performance of The Algorithm

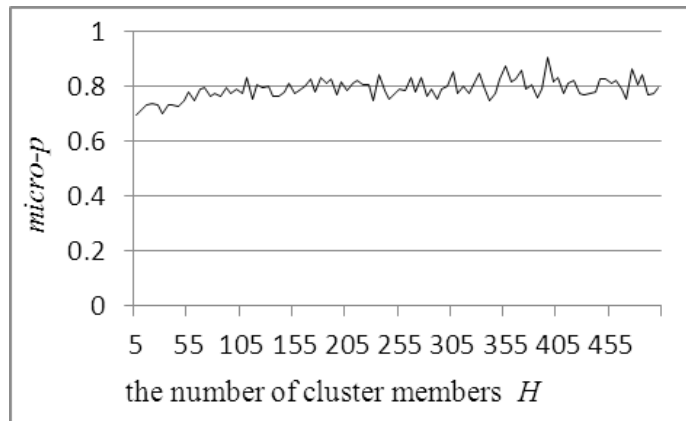
The clustering ensemble algorithm has three parameters, the embedding dimension  $d$ , the nearest neighbor number  $k$  and the number of cluster members  $H$ . Fig. 1 shows  $k$  is fixed at 10 and *micro-p* changes with  $d$  in the Coffee dataset. We can see when the  $d$  is relatively low, the *micro-p* value is relatively low and the clustering performance is poor. A possible explanation is that the different time series samples through LPP mapping are overlap together in low-dimensional embedding space. With  $d$  gradually increased the *micro-p* value increases rapidly. The clustering ensemble method is thus proposed to obtain good clustering results without high embedding dimension.



**Figure 1:** Change of *micro-p* with Embedding Dimension  $d$  in Coffee Dataset



**Figure 2:** Change of *micro-p* with Nearest Neighbor Number  $k$  in Face Four Dataset



**Figure 3:** Change of *micro-p* with the Number of Cluster Members  $H$  in CBF Dataset

Fig. 2 shows the  $d$  is fixed at 14, *micro-p* changes with the  $k$  in Face Four dataset. We can see, with the increase of  $k$ , *micro-p* in a certain range of fluctuations, relatively stable. The effect of  $k$  on the performance of the cluster is small. In order to observe the effect of clustering members on the performance of the algorithm, the number of members  $H$  is changed from 5 to 500 and the interval is 5. Fig. 3 shows that the *micro-p* changes with the number of clustering members  $H$  in the CBF dataset. We can see when  $H$  grows from 5 to 100, the clustering performance is gradually improved; when  $H$  continues to increase, the clustering performance in a certain range is stable.

## 5. Conclusion

In this paper, a time series clustering ensemble algorithm based on LPP is proposed. Firstly, use LPP to time series samples for dimensionality reduction; secondly, carry out the clustering ensemble to the low-dimension data; finally, the clustering performance is compared with the existing methods such as PCA and PAA. Experiment results show that the proposed method can improve the clustering performance. How to select the optimal embedding dimension, improve the quality of the clustering members and the design of the ensemble function, is worth of further research in the future.

## References

- [1] H. L. Li, C. H. Guo. *Research on feature representation and similarity measurement in time series data mining*[D]. Dalian University of Technology, Dalian City, 2012 (In Chinese).
- [2] M.Y. Su, C. H. Guo. *Single variable time series clustering method based on principal component analysis*[J]. Operations and management. 2011, (6):66-72. (In Chinese).
- [3] H. L. Li, C. H. Guo, L. B. Yang. *Time series mining based on the time of block polymerization*[J]. Journal of Shandong University. 2011, 41(5):57-62. (In Chinese).
- [4] X. F. He, P. Niyogi, *Locality Preserving Projections*[C]// Advances in Neural Information Processing systems , MIT Press, Cambridge, 2003, pp:16-24.
- [5] X. Q. Weng, J. Y. Shen. *Classification of multivariate time series using locality preserving projections*[J]. Knowledge-Based Systems, 2008, 21(7):581–587.
- [6] Z. H. Zhou, W. Tang. *Clusterer ensemble*[J]. Knowledge-Based Systems, 2006, 19(1):77–83.
- [7] X. F. He, S. Yan, Y. Hu, P. Niyogi, H. Zhang. *Face recognition using laplacianfaces*[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, 27(3):328-340.
- [8] Q. Li, Z. Qiu, D. Sun, L. Liu. *Online palmprint recognition based on improved two dimensional principal component analysis*[J]. Electronic journal. 2005, 33(10):1886-1889. (In Chinese).
- [9] X. F. He, *Locality preserving projections*[D]. Computer Science Department, Chicago, 2005.
- [10] J. W. Han, Kamber M. *Concept and technology of data mining*[M]. China Machine Press, Beijing, pp:293-295. 2001. (In Chinese).
- [11] Y. Chen, E. Keogh(2015). *The UCR Time Series Classification Archive*. URL [www.cs.ucr.edu/~eamonn/time\\_series\\_data/](http://www.cs.ucr.edu/~eamonn/time_series_data/)
- [12] Z. Sheng, S. Q. Xie, *Probability theory and mathematical statistics*[M]. Higher Education Press, Beijing, pp:213-225, 2001. (In Chinese).