Neurocomputing 349 (2019) 239–247

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Multivariate time series clustering based on common principal component analysis

Hailin Li[a,b,*]

[a] College of Business Administration, Huaqiao University, Quanzhou, China
[b] Research Center for Applied Statistics and Big Data, Huaqiao University, Xiamen, China

ARTICLE INFO

Article history:
Received 2 July 2018
Revised 19 March 2019
Accepted 19 March 2019
Available online 19 April 2019

Communicated by Wei Chiang Hong

Keywords:
Multivariate time series
Clustering analysis
Common principal component analysis
Data mining
Dimensionality reduction

ABSTRACT

Time series clustering is often applied to pattern recognition and also as the basis of the tasks in the field of time series data mining including dimensionality reduction, feature extraction, classification and visualization. Due to the high dimensionality of multivariate time series and most of the previous work concentrating on univariate time series clustering, a novel method which is based on common principal component analysis, is proposed to achieve multivariate time series clustering more fast and accurately. It is inspired by the traditional clustering method *K*-Means and can construct a common projection axes as prototype of each cluster. Moreover, the reconstruction error of each multivariate time series projected on the corresponding common projection axes are used to reassign the member of the cluster. The detailed algorithm of the proposed method Mc2PCA is given and the time complexity is analyzed, which shows that the proposed method is very fast and its time complexity is linear to the number of multivariate time series objects. Unlike the traditional methods, the proposed method considers the relationship among variables and the distribution of the original data values of multivariate time series. The experimental results in the various datasets demonstrate that Mc2PCA is superior to the traditional methods for multivariate time series clustering.

1. Introduction

Time series data can be seen everywhere including stock exchange, financial market, medicine and engineering, which is one kind of the important data needed to be mined for the valuable information and knowledge. It has two categories according to various number of the variables, they are univariate time series (UTS) and multivariate time series (MTS). Beside the variable(s), time series has the time order which is much different from other data. In the field of time series data mining [1–3], the most common useful tasks are dimensionality reduction, similarity measure, classification, clustering analysis, pattern discovery and visualization.

Most of relevant work had paid much attention on pattern recognition of univariate time series [4–6]. Especially, univariate time series clustering research based on dynamic time warping (DTW) has made a lot of progress [7,8]. It is well known that DTW is one of the popular methods used to measure the similarity between two time series [9–12]. However, its heavy time complexity has become a stumbling block to the development of time series clustering, and it is hard to combine with the classic clustering methods such as *K*-Means, Fuzzy *C*-Means and *K*-Medoids to finish the clustering. One of the key factors for time series clustering based on DTW is the computation of center sequence for each cluster. Some authors [10,13] had already proposed a global averaging method (DBA) for DTW with application to time series clustering. To some extent, DBA can compute the center sequence of a group of time series, but the initialization and the length of the center sequence will produce different results for the formation of the final center sequence. Moreover, each iteration must use DTW to compute the distance between the member series in a cluster and its center sequence, which possibly costs much time when large time series data need to be clustered. In addition, some methods such as *K*-shape [8] and *K*-MS [7] can be used for fast and accurate clustering and classification in univariate time series dataset, but they are still not suitable for multivariate time series data mining.

Multivariate time series with intrinsic features such as high dimensionality and similarity measure makes the clustering progress more complex than univariate time series. Principal component analysis (PCA) [14–18] is a common method to transform MTS into a new coordinate space to find the major features. In other words, the first K principal components are selected to represent

[*] Correspondence to: College of Business Administration, Huaqiao University, Quanzhou, China.
E-mail address: hailin@mail.dlut.edu.cn

https://doi.org/10.1016/j.neucom.2019.03.060
0925-2312/© 2019 Elsevier B.V. All rights reserved.

most of the information about the original MTS. Generally, the corresponding similarity measure methods are proposed to reflect the relationship between two time series in the new coordinate space. One of the earliest measure based on PCA was proposed by Krzanowski [19], and it compares the angles between all the combinations of any two retained components. However, it did not take the different weights of various components into consideration. Later, Johannesmeyer [20] proposed another method to modify the previous measure by weighting the angles according to the contribution of every component. Eros (*E*xtended F*ro*benius norm*s*) [21–23] is one of the most popular methods used to measure the similarity between two groups of the components after coordination transformation by PCA. Singhal and Seborg [24] extended the version of Johannesmeyer [20], which considers the influence of the original values and is based on mahalanobis distance and the Gaussian distribution. Besides PCA, some variants such as 2dSVD (two-dimensional singular value decomposition) [25], 2dPCA (two-dimensional principal component analysis) [26] and CPCA (common principal component analysis) [27,28] are able to reduce the dimensionality and also design the corresponding similarity measure applied to MTS data mining. In addition, some work based on wavelet [29] and hybrid distance [30] were proposed to cluster multivariate time series, but they were often designed for the special domain knowledge.

Through analyzing the main traditional methods [14,31] used to cluster multivariate time series dataset, we summarize some work that needs to study. (1) DTW can consider the difference of the shapes and values of time series, but the computation of similarity measure costs much time and a good center series of MTS is hard to obtained in the procedure of MTS clustering. (2) Most of the work for MTS data minig is based on PCA and its variants, which reflects the relationships between any two variables of MTS. However, they often ignore the comparison of the original values of MTS. (3) With the length and the volume of MTS and the number of variables increasing, the effectiveness and efficiency of the existed clustering methods should be improved. However, it is easy to find that the previous work respectively paid much attention on the original value and the relationship of variables in MTS, which also indicates that it is very important for MTS data mining in the two aspects, the original values and the relationships of variables.

In this paper, an improved method based on common principal component analysis is proposed to cluster multivariate time series data. The study motivation of the work can be summarized as follows. (1) Due to the high dimensionality of MTS, the dimensionality reduction is proposed to validly integrate into the clustering process. Moreover, a good clustering results can be obtained in the lower reduced dimensions. (2) When the length of MTS is very long and the volume of MTS dataset is very large, we hope to design a fast clustering method for MTS data whose computation speed is better than those methods based on DTW. (3) In the process of clustering analysis, the values and relationship among variables of MTS should be taken into consideration. (4) It is well known that *K*-Means is a simple and effective clustering method, of which the time complexity is linear to the number of MTS, and it is usually suitable for dynamic clustering and online clustering. The design of the proposed method is expected to reach the effects of *K*-Means.

In this work, multivariate time series clustering method based on common principal component analysis (MC_2PCA) is proposed, which is inspired by the principle of *K*-Means. Two main stages are included, constructing the projection coordination space for a cluster and reassigning MTS members to each cluster. In our work, the proposed method uses the common principal component analysis (CPCA) to construct a projection coordinate space for a cluster. Let every original MTS be projected to the new coordinate spaces and reconstruct them to compute the reconstruction error. According to the minimal one of the reconstruction errors of various clusters, the proposed method can finish assigning MTS to different cluster. The experimental evaluation shows that the new method is more effective and efficient.

The remainder of the paper is organized as follows. In Section 2, some preliminaries are introduced. In Section 3, the detailed illustration of the proposed method Mc2PCA is given. In Section 4, the comparisons of the clustering methods are arranged in the experimental evaluation. In the last section conclusion is presented.

2. Preliminaries

In this section, we introduce the common principal component analysis which is used to construct a projection axes. At the same time, we also describe the process of coordinate projection for an original MTS and the computation of the reconstruction.

2.1. Common principal component analysis

Common principal component analysis (CPCA) [27,28] is one of the variants of principal component analysis (PCA), which is often used for dimensionality reduction and feature representation for the complex data, such as multivariate time series and image pattern recognition. CPCA is based on PCA. They have the same principle to obtain the first k principle components to represent the original data.

Suppose there was a dataset X having N multivariate time series X_i. The size of a MTS X_i is $n_i \times m$, that is $X_i \in R^{n_i \times m}$, where n_i is the length of MTS X_i and m is the number of the variables. Generally, any pair of MTS X_i and X_j in the dataset X have the same number of the variables and different lengths. MTS X_i can be recorded as a matrix of size $n_i \times m$.

To obtain the common principal components of the dataset X, MTS X_i is transformed into a covariance matrix Σ_i, that is $\Sigma_i = cov(X_i)$. Formally, it is

$$\begin{aligned}\Sigma_i &= cov(X_i)\\ &= E[\bar{X}_i^T \bar{X}_i]. \qquad (1)\end{aligned}$$

where $\bar{X}_i$ is a normalized matrix of X_i, in which the elements had been eliminating the mean of the corresponding variable, that is $x_{ij} = \bar{x_{ij}} - \bar{x_{.j}}$, where $x_{.j}$ is the mean of the vector of the ith variable in X_i. It means that the expectation value of each column of the matrix $\bar{X}_i$ is equal to zero. Next, all the covariance matrixes can be averaged to a common covariance matrix $\bar{\Sigma}$ according to

$$\begin{aligned}\bar{\Sigma} &= covs(X)\\ &= \frac{1}{N}\sum_{i=1}^{N}\Sigma_i. \qquad (2)\end{aligned}$$

Singular value decomposition (SVD) can be used to decompose the common covariance matrix $\bar{\Sigma}$. It can obtain the eigenvalues $\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_m]$ and eigenvectors $U = \{U_1, U_2, \ldots, U_m\}$, where $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_m$ and $\bar{\Sigma} U_i = \lambda_i U_i$. According to the value of λ_i, the information contribution of the ith principal component and the order of the information contribution can be known. Therefore, it usually selects the first p eigenvectors to construct a common space $S = [U_1, U_2, \ldots, U_p]$, that is $S = U(:, 1:p) = [U_1, U_2, \ldots, U_p]$, where $p \leq m$. The common space based on covariance matrixes $\Sigma = \{\Sigma_1, \Sigma_2, \ldots, \Sigma_N\}$ can be obtained according to the Algorithm 1.

Thus, the first p principle components P_i of MTS X_i can be obtained, that is

$$P_i = X_i S. \qquad (3)$$

S is a common space constructed by CPCA executing on the MTS dataset, in which the features of MTS X_i can be described better

Algorithm 1 The algorithm of the common space, $S = CPCA\_S(\Sigma, p)$.

```
$\bar{\Sigma} \Leftarrow 0$ and $N \Leftarrow Getnumber(\Sigma)$.
for $i = 1$ to $i = N$ do
    $\bar{\Sigma} \Leftarrow \bar{\Sigma} + \Sigma_i$
end for
$\bar{\Sigma} \Leftarrow \frac{1}{N}\bar{\Sigma}$
$U = SVD(\bar{\Sigma})$
$S = U(:, 1 : p)$
```

than the original one. The reason is that the original MTS in the common space S has lower dimension but retains most of the data information.

2.2. Projection and reconstruction

Through CPCA, we can obtain the common space S, which is here called as common projection axes. The original MTS X_i can be projected onto the common projection axes S according to (3). In this way, the original MTS X_i and X_j can be transformed into the data in the new projection axes to better compare their difference in the lower dimension. In other words, the transformed data in the new projection axes are made of the first p principal components P_i. Some work [23,27,28] often used them to replace the original MTS to achieve the task of MTS data mining.

According to the principle of SVD, the projection axes S can be also used to reconstruct a new multivariate time series in the original space. The way is

$$\begin{aligned} Y_i &= P_i S^T \\ &= X_i S S^T. \end{aligned} \tag{4}$$

Y_i is the reconstruction one of the principal components P_i from the projection axes to the original time series. Since the first p principal components are previously retained and some information may be lost in the process of space coordinate transformation, the reconstruction series may be different from the original one. It means that there is some reconstruction error happened. Formally, the reconstruction error between Y_i and X_i is

$$\begin{aligned} E_i &= \| Y_i - X_i \|_2 \\ &= \sum_{j=1}^{n_i} \sum_{k=1}^{m} (y_{jk} - x_{jk})^2 \end{aligned} \tag{5}$$

where x_{ik} is the element of the matrix X at the ith row and the kth column. The construction error E_i is caused by two factors. One is the reduced dimensionality of the projection axes, another is the quality of the common space produced by CPCA. It means that the error is dependent on the retained number of the eigenvectors and the quality of the common space S constructed by the eigenvectors.

3. MTS clustering based on CPCA

Multivariate time series clustering is one of the most important tasks in the field of time series data mining. Recently, two kinds of MTS clustering have attracted much attention. One is the clustering methods based on PCA, the other is the ones based on DTW. The former analyzes the relationship among the variables of MTS and regards the ones with similar relationships as the members of a cluster. The latter measures the similarity between two MTS by DTW and divides the ones with the close values or similar shape trends into a group. In some cases, they can obtain good clustering results, but there exists some further work need to be researched, such as the improved quality of the methods based on PCA, the computation efficiency of the methods based on DTW, and the requirement of the clustering methods needed to consider the values and the relationships of the variables in MTS.

In this work, a novel method, multivariate time series clustering based on common principal component analysis (Mc2PCA), is proposed. It can simultaneously consider the relationships among different variables and the data value distribution of MTS. It is very important that the process of Mc2PCA seems to be the traditional K-Means, which indicates that the proposed method have some abilities of K-Means, such as fast computation, dynamic clustering and online clustering.

Let X denote a MTS dataset that has N MTS objects, that is $X = \{X_1, X_2, \ldots, X_N\}$. Moreover, X_i and X_j may have different lengths n_i and n_j respectively but must have the same number m of the variables, which means there is $n_i \neq n_j$ for any two MTS X_i and X_j, where $X_i \in R^{n_i \times m}$ and $X_j \in R^{n_j \times m}$. Now the clustering task is to divide the MTS dataset X into K clusters (or groups). MTS in a cluster are as similar as possible and the ones in various clusters are different from each other.

Mc2PCA like K-Means has two stages. One is to assign every MTS to a cluster and the other is to construct a prototype of a cluster. The assignment of MTS often relies on the distance from the MTS to the prototype. Therefore, the key work of Mc2PCA is to design a suitable prototype. In the previous work, there were often three prototypes [2], they are the medoid sequence, the averaged sequence and the local search prototype. In this work, we propose another prototype based on CPCA, which is the common projection axes of a cluster. In other word, the common projection axes S_i of the ith cluster is the prototype used to construct the reconstruction sequence of the MTS in the corresponding cluster.

Suppose there were n_k MTS objects in the kth cluster C_k, that is $C_k = \{X_{k1}, X_{k2}, \ldots, X_{kn_k}\}$. The MTS objects in a cluster C_k can be used to construct the common projection axes S_k according to Algorithm 1. It is easy to know that the more the MTS objects in a cluster are similar, the better the reconstruction quality of each MTS in C_k is according to (4) for a fixed number p of retained principal components. It means that a good common projection for a cluster can obtain high quality of the reconstruction for a MTS. So in this way, the common projection axes in a cluster can be seen as the prototype of the cluster.

K clusters have K prototypes to form construct the corresponding common projection axes. It means that clusters $C = \{C_1, C_2, \ldots, C_K\}$ can respectively produce the corresponding common projection axes $S = \{S_1, S_2, \ldots, S_K\}$. Thus, the common projection axes S_k is regarded as the prototype of the kth cluster C_k, where $k = 1, 2, \ldots, K$.

Besides the definition of the prototype, the other stage of Mc2PCA is to assign a MTS ($X_i \in X$) to a cluster. According to (4) and (5), the stage projects every MTS X_i in dataset X on the kth common projection axes and transform the group of the retained principal components to construct the corresponding reconstruction sequence Y_i^k. Y_i^k is the reconstruction sequence of MTS X_i projected and transformed from the kth common projection axes. That is

$$Y_i^k = X_i S_k S_k^T. \tag{6}$$

The reconstruction error E_{ik} between the original MTS X_i and Y_i^k can be used to assign the MTS X_i to the k' cluster that makes the E_{ik} be minimal for all k values, that is

$$\begin{aligned} k' &= \arg_k \min_{k=1}^{K} E_{ik} \\ &= \arg_k \min_{k=1}^{K} \| Y_i^k - X_i \|_2 \\ &= \arg_k \min_{k=1}^{K} \sum_{j=1}^{n_i} \sum_{l=1}^{m} (y_{jl}^k - x_{jl})^2. \end{aligned} \tag{7}$$

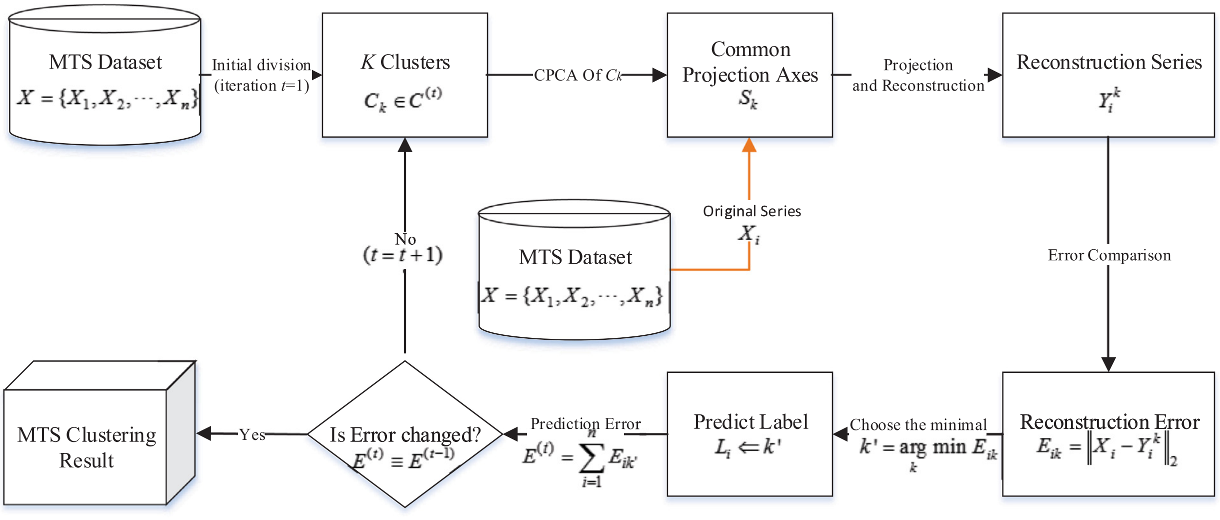


Fig. 1. The process of MTS clustering based on CPCA.

The proposed clustering method Mc2PCA can be illustrated as shown in Fig. 1. For a MTS dataset $X=\{X_1,X_2,\dots,X_n\}$, Mc2PCA initially divides the dataset into K clusters $C^{(t)}$, which is the same to the initial process of K-Means, where the iteration number t is equal to 1 ($t=1$) at the beginning of the process. In addition, in this paper we initially average the number L of the objects in the cluster C_k that is $L=n/K$, where $k=1,2,\dots,K$. Each cluster C_k can be transformed to construct the corresponding common projection axes S_k by CPCA, that is $S_k = CPCA(cov(C_k),p)$, where p is the number of the retained eigenvectors of the common covariance matrix $\bar{\Sigma}$. Let every MTS X_i project onto K common projection axes, then K reconstruction sequences can be obtained, which are respectively denoted as $Y_i^1, Y_i^2, \dots$ and Y_i^K. According to (7), the MTS X_i will be assigned to the cluster which constructs the common projection axes making MTS X_i have the minimal reconstruction error $E_{ik'}$. It means the predict label L_i of X_i is assigned with the value k' which is computed by (7). For all the MTS objects in the dataset X, the overall reconstruction error $E^{(t)}$ is made of every the minimal reconstruction error $E_{ik'}$, that is $E^{(t)}=\sum_i^n E_{ik'}$. Until the overall reconstruction errors $E^{(t)}$ and $E^{(t-1)}$ of the two adjacent iterations is not changeable or the iteration number exceeds the allowed value, then the procedure is stopped and returns the predict label of every MTS object in X as the clustering result. Otherwise, the iteration t is plus one ($t=t+1$), and the new clusters $C^{(t)}$ can be formed according to the predict label values of MTS objects in the dataset X. Thus, the process is redone again. The algorithm of Mc2PCA is shown in Algorithm 2.

In the Algorithm 2, the first line is used to get the number of MTS objects in dataset X and initializes some variables. Lines 2–5 normalize every MTS X_i by removing the mean of every column vector and compute the covariance matrix for each normalized MTS objects. Lines 6–9 equally divide the dataset into K initial clusters and construct K common projection axes, where $index_k$ is a vector and records the class labels that indicates that which MTS objects are in the kth cluster. Lines 12–21 are used to predict the label of every MTS object in dataset X. Line 15 computes the construction error between X_i and Y_i^k. Line 17 obtains the minimal value of construction errors with different common projection axes and the corresponding index the of clusters. Line 18 is used to accumulate the minial reconstruction errors for all MTS objects at iteration t. Line 19 records the members of the Ith cluster and line 20 records the labels of MTS objects. Lines 22–24 judges whether the reconstruction errors of two adjacent iterations are changed. If it is true, then stop the loop. Lines 25–27 recompute the K common projection axes according to the corresponding new clusters obtained at line 19.

Algorithm 2 The algorithm of Mc2PCA is described, $idx = Mc2PCA(X,K,p)$.

```
1:  $n \Leftarrow Getnumber(X)$, $L \Leftarrow n/K$, $t \Leftarrow 0$, $maxt \Leftarrow 100$, and $E^{(0)} \Leftarrow 0$.
2:  for $i = 1$ to $i = n$ do
3:      $X_i \Leftarrow Normalize(X_i)$
4:      $\Sigma_i \Leftarrow cov(X_i)$
5:  end for
6:  for $k = 1$ to $k = K$ do
7:      $index_k \Leftarrow (k-1)*L+1 : min(k*L, n)$
8:      $S_k \Leftarrow CPCA\_S(\Sigma_{index_k}, p)$
9:  end for
10: while $t \le maxt$ do
11:     $t \Leftarrow t+1$, $E^{(t)} \Leftarrow 0$, $index_{1,2,\cdots,K} \Leftarrow null$
12:     for $i = 1$ to $i = n$ do
13:         $Error(i) \Leftarrow 0$
14:         for $k = 1$ to $k = K$ do
15:             $Y_i^k \Leftarrow X_i S_k S_k^T$, $Error(k) \Leftarrow \| X_i - Y_i^k \|_2$
16:         end for
17:         $[v, I] \Leftarrow \min(Error)$
18:         $E^{(t)} \Leftarrow E^{(t)} + v$
19:         $index_I \Leftarrow Index_I \bigcup i$
20:         $idx(i) \Leftarrow I$
21:     end for
22:     if $E^{(t)} = E^{(t-1)}$ then
23:         Break
24:     end if
25:     for $k = 1$ to $k = K$ do
26:         $S_k \Leftarrow CPCA\_S(\Sigma_{index_k}, p)$
27:     end for
28: end while
29: Print $idx$
```

The time complexity of Mc2PCA as well as the algorithm K-Means is linear to the number of MTS objects in dataset X, that is $O(tKN)$, where t is the iteration number, K is the number of clusters and N is the number of MTS objects. The detailed analysis of the time complexity can be seen in Table 1.

Considering the size of MTS objects and the computation time of covariance matrixes, common projection and reconstruction, the overall time of Mc2PCA can be analyzed as follows. (1) The time

Table 1
The detailed analysis of time complexity of Mc2PCA.

Main Processes	Covariance matrix	Common projection	Reconstruction
Steps	2$\sim$ 3	6 $\sim$ 9 or 25 $\sim$ 27	12 $\sim$ 21
Execution time	nm^2	Km^3	$NKnmp$
Times(iterations)	N	t	t
Time complexity	Nnm^2	tKm^3	$tNKnmp$
Sum	$n(Nm^2+tNKmp)+tKm^3$		

consumption of covariance matrixes for all MTS objects is $O(Nnm^2)$, where n is the length of a MTS and m is the number of the variables existing in the MTS object. (2) The construction of K common projection axes S must cost $O(Km^3)$ at each iteration, which means that $O(tKm^3)$ should be cost to update the common projection axes t times in the process of Mc2PCA. (3) The assignment of MTS objects to the corresponding clusters from lines 12 to 21 of the Algorithm 2 costs $O(NKnmp)$. The t times of assignments will cost $O(tNKnmp)$. Therefore, the overall computation time of Mc2PCA is $O(Nnm^2+tKm^2+tNKnmp)$. However, it is usually $n \ll N$ and $p \leq m \ll N$, which makes the execution of Mc2PCA is very fast. Therefore, Mc2PCA is linear to the number of MTS objects in the dataset X and can be regarded as an effective clustering method for MTS dataset.

4. Experimental evaluation

In order to test the performance of the proposed method Mc2PCA, we arrange some experiments to compare the quality and the efficiency of the existing clustering methods for various multivariate time series datasets.

4.1. MTS datasets

We collected 13 various MTS datasets that are applied to the following experiments. They are shown in Table 2. They are from different applications such as speech recognition, activity recognition, medicine and etc. The characteristics of MTS in each dataset include name, variables number, the length of MTS, classes number and instances volume. In Table 2, the lengths of MTS in the first 6 datasets are unequal and the MTS in the other datasets (from No. 7 to No. 13) have the same length. For example, the length of MTS in the first dataset AUSLAN is from 45 to 136 and the length of MTS in dataset No. 7 is same to each other and is equal to 15. In addition, the number of variables in various datasets is from 2 to 62, which means that different dimensions had already been considered in the process of the following clustering experiments. At the same time, the volumes of the MTS datasets are also different. The minimal number is 29 and the maximal one is 10692.

4.2. MTS clustering

Since Mc2PCA is based on CPCA and also considers the data value comparison, two categories of the traditional MTS clustering methods are used to compare their performance. Two of them are K-Means and Spectral Clustering [7] based on DTW similarity measure [8] that are written as KMeans_DTW and Spectral_DTW respectively, and another is Permutation Distribution Clustering (PDC) [31] which is often used to cluster time series. The fourth compared method [14] based on PCA Similarity Factor and distance Similarity Factor are often used to cluster multivariate time series. It is necessary to point out that K-Means_DTW and Spectral_DTW use the distance matrix computed by DTW to cluster MTS objects by the traditional clustering methods K-Means and spectral clustering. Moreover, according to the experience of the previous research [7,32], *DTW* with a slide window of which the length is approximately 10 percent of the length of the compared time series can obtain good measure. So in this experiment, the length of the slide window in DTW is set to be $0.1 \times \max(n_i, n_j)$, where n_i and n_j are respectively the length of the two compared multivariate time series X_i and X_j.

In the proposed method, there are two parameters needed to be set. One is the number (K) of clusters, the other is the lower reduced dimension (p). In the clustering algorithm, it is often hard to set the number of clusters. Fortunately, Rodriguez and Laio [33] proposed a clustering method by fast search and find of density peaks which can easy to find the suitable number (K) of clusters. In the experiments of MTS data clustering, the number K of clusters is uniformly set to the corresponding number of classes as shown in fifth column of Table 2 for the datasets, respectively. The proposed method Mc2PCA has another parameter p, which is the number of the retained eigenvectors of the common covariance matrix for each cluster. It indicates the reduced dimension. Considering that the dimension of most multivariate time series dataset in Table 2 does not exceed 6, in order to observe the clustering effect after dimension reduction, we set the experimental parameter p from 1 to 4. It means that the reduced dimension p is less than or equal to 4 in the experiment of each dataset, that is $p=[1,2,3,4]$. Actually, the best value of the reduced dimension p is different for the clustering of different dataset. But in lower reduced dimension, we could often set it to be less than or equal to 4 in the datasets as shown in Table 2. In addition, Kmeans_DTW and Spectral_DTW are based on the distance matrix that are computed by DTW between multivariate time series. It means the the clustering function Kmeans_DTW and Spectral_DTW can be written as *idx*=Kmeans_DTW(*distM, K*) and *idx*=Spectral_DTW(*distM, K*), respectively. The distance matrix *distM* is obtained by DTW computing the distance between each pair of MTS in the dataset, that is *distM*=DTW(*D, r*), where D denotes the dataset and r is the length of the slide window and is often 0.1. Another two methods PDC and SF can be written as *idx*=PDC(*D, K*) and *idx*=SF(*D, K*), respectively. Through executing the above clustering functions in the experiment, the clustering results can be stored in the vector *idx*, which records the prediction label for each time series.

Let Mc2PCA perform on the various MTS datasets as shown in Table 2 according to the four values of p. Variables number m in some datasets is less than the max value of p, that is $m < p_{\max}$, which leads to Mc2PCA only perform on the corresponding MTS datasets m times. In addition, to compare the quality of the methods used to cluster MTS, clustering precision denoted as *Pre* had been regarded as a stable evaluation criterion. The criterion is

$$Pre = \sum_{j=1}^{K} \frac{|C_j|}{N} \times \max_{i=1,2,\cdots,g} \frac{|G_i \cap C_j|}{|C_j|}, \tag{8}$$

where C_j is the jth predicted cluster and $|C_j|$ represents the number of MTS in cluster C_j. G_i is the ith true group in which the MTS objects are similar and g is the number of classes.

According to the above analysis, Mc2PCA clusters MTS data in the 13 MTS datasets according to the different values of p as shown in Table 3. It is easy to find that the precision of Mc2PCA with different reduced dimension p for MTS clustering is not stable. It means that the larger p value will not necessarily make the clustering result better. In other word, the smallest p value in some cases such as CMU_MOCAP_S16 and Robot FailureLP4 can obtain the best clustering results. Therefore, the value p in the input parameters of Mc2PCA can be set to 1 by default, that is $p=1$. In Table 3, the symbol "-" indicates that Mc2PCA is not executed because of $m<p$.

The traditional MTS clustering methods including KMeans_DTW, Spectral_DTW, PDC and SF are used to perform on the 13 datasets and their clustering results are shown in Table 4, which is used to compare the quality of the five methods

Table 2
Multivariate time series datasets

No.	Name	Variables number	Length	Classes number	Volume
1	AUSLAN	22	[45–136]	95	1425
2	ArabicDigits	13	[4–93]	10	2200
3	CMU_MOCAP_S16	62	[127–580]	2	29
4	Character Trajectories	3	[109–205]	20	2558
5	ECG	2	[39–152]	2	100
6	Japanese Vowels	12	[7–29]	9	370
7	Robot Failure LP1	6	15	4	50
8	Robot FailureLP2	6	15	5	30
9	Robot FailureLP3	6	15	4	30
10	Robot FailureLP4	6	15	3	75
11	Robot FailureLP5	6	15	5	100
12	LIBRAS	2	45	15	180
13	Pendigits	2	8	10	10692

Table 3
The clustering precision of Mc2PCA in the lower reduced dimensions.

No.	Name	*Pre*			
		$p = 1$	$p = 2$	$p = 3$	$p = 4$
1	AUSLAN	0.4316	0.5067	0.5488	0.4835
2	ArabicDigits	0.4241	0.5323	0.4618	0.4123
3	CMU_MOCAP_S16	0.6552	0.6207	0.6207	0.6207
4	Character Trajectories	0.4605	0.4949	0.1263	–
5	ECG	0.6700	0.6700	–	–
6	Japanese Vowels	0.5676	0.5973	0.6108	0.6405
7	Robot Failure LP1	0.5000	0.5400	0.4600	0.5200
8	Robot FailureLP2	0.5667	0.6000	0.7000	0.6667
9	Robot FailureLP3	0.6333	0.6333	0.5667	0.7000
10	Robot FailureLP4	0.8000	0.6800	0.7067	0.6800
11	Robot FailureLP5	0.4700	0.4100	0.4000	0.3900
12	LIBRAS	0.3278	0.0667	–	–
13	Pendigits	0.2908	0.1042	–	–

for MTS clustering analysis. In the propose method Mc2PCA, the precision values under $p = 1$ are used to compare with other traditional methods. The value in bracket indicate the optimal (best) result of MTS clustering analysis by Mc2PCA under the four reduced dimensions $p = 1, 2, 3, 4$. The original precision values of the Table 4 make the clustering results difficult to identify which methods have better performance in the datasets.

To better compare the precision values of the five clustering methods in the 13 MTS datasets, we normalize the each row precision values of Table 4. The *min_max* normalization equation is used, that is

$$\overline{Pre}(i, j) = \frac{Pre(i, j) - \min(Pre(i, :))}{\max(Pre(i, :)) - \min(Pre(i, :))}. \tag{9}$$

$Pre(i, j)$ denotes the original precision of the jth clustering method performed on the ith dataset. According to the column order of Table 4, the first clustering method is Mc2PCA, and the second one is *K*Means_DTW, and so on. After normalization, Table 4 can be transformed as Table 5. Moreover, the average of each column values are added to the last row of Table 5.

In Table 5, $\overline{Pre}$ is from 0 to 1, and the bigger the value is, the better the clustering quality is. Compared to other traditional methods, the proposed method Mc2PCA can get the most number of the values that are equal to 1, which means that Mc2PCA can retrieve the best clustering results in most of MTS datasets. In particular, the average of each column values shows that the clustering precision of the proposed method Mc2PCA is the best in all the methods. However, *K*Means_DTW is also able to get good results, but its average value (0.5755) is still lower than the optimal one (0.7515) and even lower than that (0.6245) of Mc2PCA with p value set by default. In this case, the same distance matrix measured by DTW integrated into *K*-Means and Spectral clustering produces various averaged results, 0.5755 and 0.4826 respectively. However, the two traditional methods SF and PDC are less effective than the first three ones. Therefore, the proposed method McPCA compared to other four traditional ones is the most effective for MTS data clustering analysis.

From the comparison of MTS clustering results, it is well known that Mc2PCA can obtain the good effects in most cases. However, the clustering results of Mc2PCA in some datasets such as ECG, LIBRAS and Pendigits is unsatisfactory. The reason is that the operation of the dimensionality reduction and data transformation for the datasets with 2 dimensions are prone to information lossy and information disturbance. In addition, if MTS with small length and low dimension in the datasets are projected into the common subspace by data dimension reduction, it possibly cause too much data

Table 4
The clustering precision of five methods for MTS data clustering.

No.	Name	*Pre*				
		Mc2PCA(*best*)	*K*Means_DTW	Spectral_DTW	SF	PDC
1	AUSLAN	0.4316(0.5488)	0.3200	0.3200	0.1235	0.0884
2	ArabicDigits	0.4241(0.5323)	0.3873	0.3873	0.3414	0.1041
3	CMU_MOCAP_S16	0.6552(0.6552)	0.6207	0.6207	0.5517	0.5862
4	Character Trajectories	0.4605(0.4949)	0.2467	0.2467	0.3374	0.2076
5	ECG	0.6700(0.6700)	0.7800	0.7800	0.6700	0.6800
6	Japanese Vowels	0.5676(0.6405)	0.3919	0.3919	0.3108	0.3081
7	Robot Failure LP1	0.5000(0.5400)	0.5000	0.5000	0.6000	0.5000
8	Robot FailureLP2	0.5667(0.7000)	0.6333	0.6333	0.5667	0.6000
9	Robot FailureLP3	0.6333(0.7000)	0.5333	0.5333	0.5333	0.6667
10	Robot FailureLP4	0.8000(0.8000)	0.6800	0.6800	0.8267	0.6800
11	Robot FailureLP5	0.4700(0.4700)	0.4300	0.2900	0.4600	0.3800
12	LIBRAS	0.3278(0.3278)	0.5944	0.4833	0.2389	0.3889
13	Pendigits	0.2908(0.2908)	0.7198	0.6474	0.3471	0.1062

Table 5
The normalized clustering precision of five methods for MTS data clustering.

No.	Name	$\overline{Pre}$				
		Mc2PCA(*best*)	*K*Means_DTW	Spectral_DTW	SF	PDC
1	AUSLAN	1.0000(1.0000)	0.6748	0.6748	0.1022	0.0000
2	ArabicDigits	1.0000(1.0000)	0.8849	0.8849	0.7415	0.0000
3	CMU_MOCAP_S16	1.0000(1.0000)	0.6667	0.6667	0.0000	0.3333
4	Character Trajectories	1.0000(1.0000)	0.1546	0.1546	0.5131	0.0000
5	ECG	0.0000(0.0000)	1.0000	1.0000	0.0000	0.0909
6	Japanese Vowels	1.0000(1.0000)	0.3229	0.3229	0.0104	0.0000
7	Robot Failure LP1	0.0000(0.4000)	0.0000	0.0000	1.0000	0.0000
8	Robot Failure LP2	0.0000(1.0000)	1.0000	1.0000	0.0000	0.5000
9	Robot Failure LP3	0.7501(1.0000)	0.0000	0.0000	0.0000	1.0000
10	Robot Failure LP4	0.8182(0.8182)	0.0000	0.0000	1.0000	0.0000
11	Robot Failure LP5	1.0000(1.0000)	0.7778	0.0000	0.9444	0.5000
12	LIBRAS	0.2500(0.2500)	1.0000	0.6875	0.0000	0.4219
13	Pendigits	0.3008(0.3008)	1.0000	0.8820	0.3925	0.0000
	Average	**0.6245**(**0.7515**)	0.5755	0.4826	0.3619	0.2189

Table 6
The CPU runtime of five methods for MTS data clustering.

No.	Name	$T/10^4$				
		Mc2PCA	*K*Means_DTW	Spectral_DTW	SF	PDC
1	AUSLAN	0.0038	0.1978	0.1977	3.8746	1.2838
2	ArabicDigits	0.0013	0.2209	0.2214	2.7809	1.7900
3	CMU_MOCAP_S16	0.0000	0.0016	0.0016	0.0018	0.0023
4	Character Trajectories	0.0022	4.2017	4.2033	7.9222	0.7600
5	ECG	0.0000	0.0016	0.0016	0.0001	0.0002
6	Japanese Vowels	0.0000	0.0013	0.0013	0.0802	0.0663
7	Robot Failure LP1	0.0000	0.0000	0.0000	0.0008	0.0001
8	Robot FailureLP2	0.0000	0.0000	0.0000	0.0006	0.0000
9	Robot FailureLP3	0.0000	0.0000	0.0000	0.0005	0.0000
10	Robot FailureLP4	0.0000	0.0001	0.0001	0.0011	0.0002
11	Robot FailureLP5	0.0000	0.0001	0.0001	0.0022	0.0004
12	LIBRAS	0.0000	0.0019	0.0019	0.0083	0.0003
13	Pendigits	0.0078	0.7672	0.9703	0.9578	1.6434

information loss or disturbs the true relationship between the original time series data. Overall, the proposed method Mc2PCA can obtain the acceptable clustering results in the dataset with higher dimension or bigger length. In fact, only high-dimensional time series need dimensionality reduction, in this case our method can achieve good clustering results.

4.3. Time consumption comparison

To test the efficiency of the five methods used for MTS data clustering, we recorded the runtime of the corresponding programs to finish clustering the MTS data in the 13 datasets. The time consumption of the clustering tasks to obtain the Table 4 is recorded as shown in Table 6. It should be noted that the recorded time of the proposed method Mc2PCA executing at $p = 1$. From the Table 6, we know that the consumption time of Mc2PCA is far less than other methods. Moreover, due to the huge variability of the values in Table 6, the specific runtime with preserving the four significant digits can not be displayed well for most of the datasets clustered by Mc2PCA. However, from analyzing the magnitude of the runtime values, we know that the proposed method Mc2PCA is much more efficient than other five methods.

All the algorithms in the experiment is in the same platform and the CPU time is recorded in the same way for different datasets and different methods. The computer environment settings is 64 bit Windows operation system with Intel(R) Core(TM) i5-4258U CPU @ 2.40GHz and 8G RAM. Moreover, the programming is executed in MATLAB R2010a. In order to eliminate the uncertainty of time record caused by extra factors occupying CPU resources, the average running time of CPU is taken as the running time of the corresponding method through repeated experiments. This also eliminates the problem of inconsistent CPU running time caused by time changes. For detailed analysis the runtime of the five methods, we do a logarithmic operation on these values in Table 6. At the same time, to ensure that the operated values is positive after taking the logarithm, we let the original runtime value plus 1 and then do the logarithmic operation. That is

$$T' = Log(T + 1). \tag{10}$$

After the logarithm operation, the runtime values are as shown in Table 7, which is easy to observe the runtime difference of the five methods. The average runtime is also added to compare the efficiency of the five methods for data MTS clustering. In addition, for visually observing the comparison of the operated CPU runtime, the visualization of the operated runtime is shown in Fig. 2.

In Table 7 and in Fig. 2, it is easy to find that Mc2PCA is very efficient to clustering MTS data. In the analysis of time complexity, we know that Mc2PCA is linear to the number of MTS objects in the dataset. *K*-Means and spectral clustering should have been computed efficiently, but in this case they are much less efficient than the proposed method Mc2PCA. The reason is that the two methods use DTW with high time cost to measure the similarity. In addition, SF and PDC cost much more time to cluster MTS data than the first three methods. Through comparing the efficiency of the five methods, we conclude that Mc2PCA is a fast method applied to MTS data clustering. Combining with the effectiveness, the experimental results demonstrate that Mc2PCA is an accurate and fast method for MTS data clustering.

Table 7
The logarithm operated CPU runtime of five methods for MTS data clustering.

No.	Name	T'				
		Mc2PCA	*K*Means_DTW	Spectral_DTW	SF	PDC
1	AUSLAN	3.6528	7.5904	7.5899	10.5648	9.4603
2	ArabicDigits	2.6586	7.7009	7.7032	10.2331	9.7926
3	CMU_MOCAP_S16	0.2228	2.8381	2.8345	2.9421	3.1733
4	Character Trajectories	3.1155	10.6458	10.6462	11.2800	8.9360
5	ECG	0.0457	2.8454	2.8418	0.7675	1.1409
6	Japanese Vowels	0.3839	2.6662	2.6575	6.6880	6.4989
7	Robot Failure LP1	0.0457	0.2228	0.2228	2.2399	0.6211
8	Robot FailureLP2	0.0155	0.1176	0.1314	1.9174	0.2475
9	Robot FailureLP3	0.0056	0.1167	0.1352	1.7506	0.2475
10	Robot FailureLP4	0.0307	0.4357	0.4357	2.4571	0.9910
11	Robot FailureLP5	0.0751	0.7231	0.7306	3.1475	1.5154
12	LIBRAS	0.3180	2.9872	2.9950	4.4320	1.4218
13	Pendigits	4.3756	8.9455	9.1803	9.1673	9.7071
	Average	**1.1496**	3.6796	3.7003	5.1990	4.1349

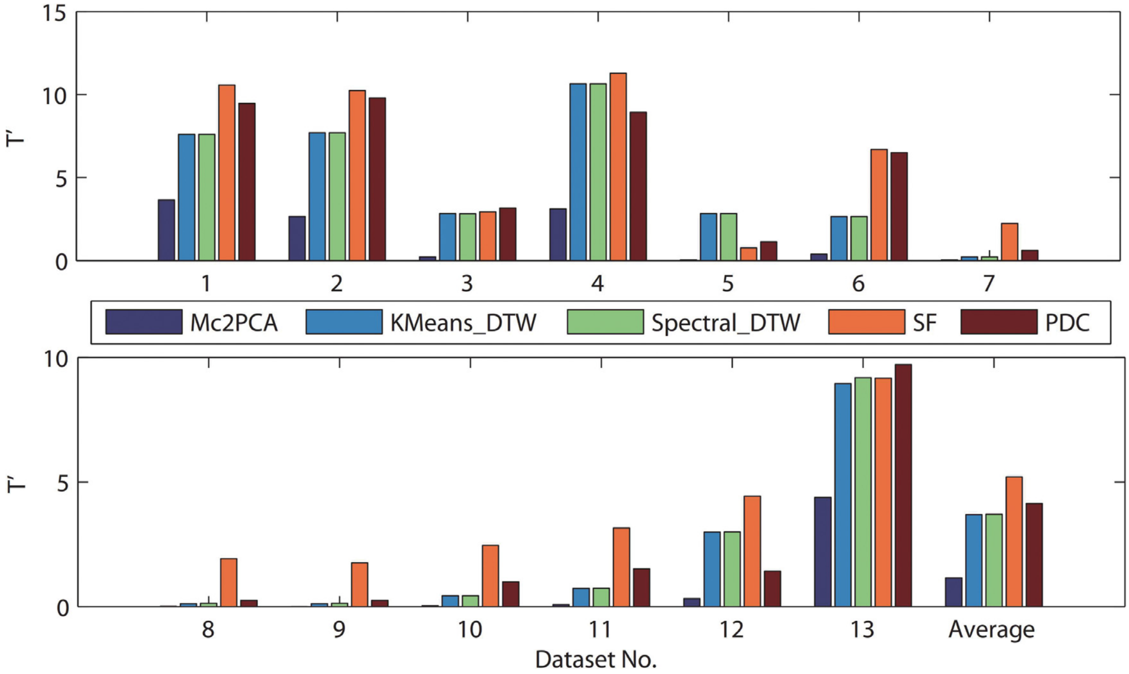


Fig. 2. The operated runtime of the five methods is visualized for efficiency comparison.

5. Conclusions

Time series clustering is one of the important tasks in the field of time series data mining. Due to the existed methods having the requirement of the improvement for multivariate time series clustering, we proposed an accurate and fast method to achieve the work. The proposed method Mc2PCA is based on common principal component analysis, which considers the relationship among various variables and the distribution of the original values. The relationship can be reflected by the covariance matrix and the distribution of the original values can be described by the the common projection axes. Mc2PCA like the tradition clustering method *K*-Means is a fast one to update the members and prototypes in every cluster with repeated iteration. In addition, the reconstruction error based on the common projection axes are used to evaluate the quality of the assignment of MTS objects to the corresponding cluster. Through comparing to other four traditional methods, the experimental results demonstrate that Mc2PCA is more effective and efficient.

In our work, there are some advantages as follows. (1) Mc2PCA simultaneously takes the variable relationship and the data values into consideration, which overcomes the shortcomings of the methods based on variable relationships or numerical values. (2) The prototype is the common projection axes of a cluster. It is based on the covariance matrix of MTS object in the corresponding cluster, and the covariance matrixes of all MTS objects are only computed once, which makes the prototype update fast. (3) The process of Mc2PCA is approximately same to the classic clustering method *K*-Means, of which the time complexity is linear to the number of MTS objects. All those advantages make the proposed method Mc2PCA have a good ability for multivariate time series clustering.

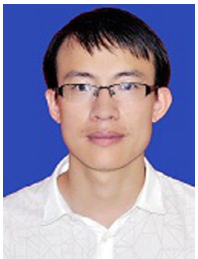
Acknowledgments

This work has been supported by the National Natural Science Foundation of China (71771094, 61300139) and the Natural Science Foundation of Fujian Province (FJ2017B065).

References

[1] N. Mishra, H.K. Soni, S. Sharma, A.K. Upadhyay, A comprehensive survey of data mining techniques on time series data for rainfall prediction, J. ICT Res. Appl. 11 (2) (2017) 167–183.

[2] S. Aghabozorgi, A.S. Shirkhorshidi, T.Y. Wah, Time-series clustering–a decade review, Inf. Syst. 53 (2015) 16–38.
[3] P. Esling, C. Agon, Time-series data mining, ACM Comput. Surv. (CSUR) 45 (1) (2012) 12.
[4] O. Lauwers, B.D. Moor, A time series distance measure for efficient clustering of input/output signals by their underlying dynamics, IEEE Control Syst. Lett. 1 (2) (2017) 286–291.
[5] B.B. Nair, P.K.S. Kumar, N.R. Sakthivel, U. Vipin, Clustering stock price time series data to generate stock trading recommendations: an empirical study, Expert Syst. Appl. 70 (2017) 20–36.
[6] E. Otranto, M. Mucciardi, Clustering space-time series: Fstar as a flexible star approach, Adv. Data Anal. Classif. (2018) 1–25.
[7] J. Paparrizos, L. Gravano, Fast and accurate time-series clustering, ACM Trans. Database Syst. 42 (2) (2017) 1–49.
[8] Z.G. Ives, Technical perspective: k-shape: Efficient and accurate clustering of time series, ACM SIGMOD Rec. 45 (1) (2016). 68–68
[9] D.J. Berndt, J. Clifford, Using dynamic time warping to find patterns in time series., in: KDD workshop, 10, Seattle, WA, 1994, pp. 359–370.
[10] F. Petitjean, G. Forestier, G.I. Webb, A.E. Nicholson, Y. Chen, E. Keogh, Dynamic time warping averaging of time series allows faster and more accurate classification, in: Proceedings of the IEEE International Conference on Data Mining (ICDM), 2014, IEEE, 2014, pp. 470–479.
[11] H. Li, On-line and dynamic time warping for time series data mining, Int. J. Mach. Learn. Cybern. 6 (1) (2015) 145–153.
[12] P.E. Tsinaslanidis, Subsequence dynamic time warping for charting: bullish and bearish class predictions for NYSE stocks, Expert Syst. Appl. 94 (2018) 193–204.
[13] F. Petitjean, A. Ketterlin, P. Gan?arski, A global averaging method for dynamic time warping, with applications to clustering, Pattern Recognit. 44 (3) (2011) 678–693.
[14] A. Singhal, D.E. Seborg, Clustering multivariate time-series data, J. Chemom. 19 (8) (2005) 427–438.
[15] H.L. Shang, A survey of functional principal component analysis, AStA Adv. Stat. Anal. 98 (2) (2014) 121–142.
[16] H. Li, Asynchronism-based principal component analysis for time series data mining, Expert Syst. Appl. 41 (6) (2014) 2842–2850.
[17] S.Y. Samadi, L. Billard, M.R. Meshkani, A. Khodadadi, Canonical correlation for principal components of time series, Comput. Stat. 32 (3) (2017) 1–22.
[18] L. Cai, N.F. Thornhill, S. Kuenzel, B.C. Pal, Wide-area monitoring of power systems using principal component analysis and *k*-nearest neighbor analysis, IEEE Trans. Power Syst. PP (99) (2018) 1–11.
[19] W. Krzanowski, Between-groups comparison of principal components, J. Am. Stat. Assoc. 74 (367) (1979) 703–707.
[20] M.C. Johannesmeyer, Abnormal situation analysis using pattern recognition techniques and historical data, Ph.D. thesis, University of California, Santa Barbara, 1999.
[21] L. Karamitopoulos, G. Evangelidis, D. Dervos, Pca-based time series similarity search, Data Mining, Springer, 2010, pp. 255–276.
[22] K. Yang, C. Shahabi, A pca-based similarity measure for multivariate time series, in: Proceedings of the 2nd ACM International Workshop on Multimedia databases, ACM, 2004, pp. 65–74.
[23] L. Karamitopoulos, G. Evangelidis, D. Dervos, Multivariate time series data mining: Pca-based measures for similarity search., in: DMIN, 2008, pp. 253–259.
[24] A. Singhal, D.E. Seborg, Pattern matching in multivariate time series databases using a moving-window approach, Industrial & engineering chemistry research 41 (16) (2002) 3822–3838.
[25] X. Weng, J. Shen, Classification of multivariate time series using two-dimensional singular value decomposition, Knowl. Syst. 21 (7) (2008) 535–539.
[26] J. Yang, D. Zhang, A.F. Frangi, J.-y. Yang, Two-dimensional pca: a new approach to appearance-based face representation and recognition, IEEE Trans. Pattern Anal. Mach. Intell. 26 (1) (2004) 131–137.
[27] Z.-X. Li, J.-S. Guo, X.-B. Hui, F.-F. Song, Dimension reduction method for multivariate time series based on common principal component, Control Dec. 28 (4) (2013) 531–536.
[28] H. Li, Accurate and efficient classification based on common principal components analysis for multivariate time series, Neurocomputing 171 (2016) 744–753.
[29] P. D'Urso, E.A. Maharaj, Wavelets-based clustering of multivariate time series, Fuzzy Sets Syst. 193 (Supplement C) (2012) 33–61.
[30] C.H. Fontes, H. Budman, A hybrid clustering approach for multivariate time series-a case study applied to failure analysis in a gas turbine, ISA Trans. 71 (Part 2) (2017) 513–529.
[31] A.M. Brandmaier, pdc: Permutation distribution clustering, Psychol. Methods 18 (1) (2015) 71–86.
[32] E. Keogh, C.A. Ratanamahatana, Exact indexing of dynamic time warping, Knowl. Inf. Syst. 7 (3) (2005) 358–386.
[33] A. Rodriguez, A. Laio, Clustering by fast search and find of density peaks, Science 344 (6191) (2014) 1492–1496.



Hailin Li received the B.S. degrees from information and computing science of Jingdezhen Ceramic University, Jingdezhen, China, in 2006 and received the Ph.D. degree in management science and engineering from Dalian University of Technology, China in 2012 respectively. From 2015 to 2018, he is an associate professor in the school of business administration, Huaqiao University, Quanzhou, China. Since 2018, he is a professor and His research interests include time series data mining and decision making. He is a leader of some Natural Science Fund Projects.