ORIGINAL PAPER

Moving dynamic principal component analysis for non-stationary multivariate time series

Fayed Alshammri[1,2] · Jiazhu Pan[1,3]

Abstract

This paper proposes an extension of principal component analysis to non-stationary multivariate time series data. A criterion for determining the number of final retained components is proposed. An advance correlation matrix is developed to evaluate dynamic relationships among the chosen components. The theoretical properties of the proposed method are given. Many simulation experiments show our approach performs well on both stationary and non-stationary data. Real data examples are also presented as illustrations. We develop four packages using the statistical software R that contain the needed functions to obtain and assess the results of the proposed method.

1 Introduction

Multivariate time series analysis has many applications, as it can account for inter-relations between variables. Advanced technology nowadays allows for the collection of multivariate natured data in a wide range of fields, such as economics, industry, healthcare, and social networks. Many of the existing models, such as VARIMA models, face the challenge of complexity in their structures, even when modelling series

✉ Fayed Alshammri
  fayed.alshammri@strath.ac.uk

✉ Jiazhu Pan
  jiazhu.pan@strath.ac.uk

1   Department of Mathematics and Statistics, University of Strathclyde, Glasgow G1 1XH, UK

2   Department of Basic Sciences, Saudi Electronic University, Riyadh, Kingdom of Saudi Arabia

3   School of Mathematics and Statistics, Yangze Normal University, Chongqing 408100, China

🖄 Springer

with large dimensions. This complexity occurs because the number of parameters expands enormously fast as the dimension increases. Therefore reducing the dimension of the series becomes critical to manage such data.

Many approaches have been proposed in literature for dimension reduction. Factor models are widely used tools to reduce the dimension of a vector time series by applying eigenanalysis on the covariance matrix of the data. See, for example, Peña and Box (1987), Stock and Watson (1988, 2002), Bai and Ng (2002), Forni et al. (2005), Peña and Poncela (2006), Pan and Yao (2008), Lam and Yao (2012), and many others. These models treat the variables of an observed series as linear combinations of some hidden factors that could be interpreted subjectively. Another approach of interest is the principal component analysis (PCA), where it also applies eigenanalysis on the covariance matrix of the data. PCA seeks dimension reduction by retaining a small number of principal components that are linear combinations of the original variables. PCA is, in fact, a commonly used technique to perform dimension reduction for static and independent multivariate data. However, because of the dynamic nature of multivariate time series data, the classical PCA technique will not be applicable. The reason is that PCA is static, therefore, will not be able to capture the dynamic dependence between the variables of a multivariate time series. Dimension reduction for time series data can also be achieved using canonical correlation analysis by Box and Tiao (1977) and scalar component analysis by Tiao and Tsay (1989).

Ku et al. (1995) introduced dynamic principal component analysis (DPCA) by including lagged series into the analysis. Without losing a valuable amount of information, the results of projected components are linear combinations of both current and lagged values of the data. However, DPCA assumes a stationary series. Therefore, it is not suitable for non-stationary series.

Chang et al. (2018) extended PCA by transforming the original series into uncorrelated subseries with lower dimensions. This method is called the principal component analysis for time series (TS-PCA). The resulted subseries can be separately analyzed as they are uncorrelated. However, this method is also limited to stationary series.

Many PCA-based methods were proposed to account for non-stationarity such as moving window principal component analysis (MWPCA) by Lennox et al. (2001) and variable MWPCA by He and Yang (2008). These methods were mostly developed for process monitoring, where PCA is performed separately on each window. By including data from the next time point and excluding those from the oldest time point, new results are obtained based on the new window and so on. However, excluding a large amount of observation by using one widow at a time would lead to the loss of a valuable amount of information.

Brillinger (1981) proposed another related approach where the reduction is produced based on a reconstruction criterion. The resulted dynamic components are linear combinations of the original series. Peña and Yohai (2016) proposed the generalized dynamic principal component analysis (GDPCA), where the original data is reconstructed based on a loss function. This method accounts for non-stationary series and produces dynamic principal components that could be non-linear combinations of the original data with nearly zero reconstruction error. This precision is a result of using an iteration method that minimizes the reconstruction error. However, using this iteration

method reduces the accuracy of forecasting by using GDPCA's results; See Peña and Yohai (2016).

In this paper, we extend DPCA to non-stationary vector time series. The main difference between DPCA and the method we propose is that the former uses the classical covariance matrix of the data, where the latter uses a new form of the covariance matrix called the moving cross-covariance matrix of the data. This new matrix updates itself at each time point and consists of static and dynamic information of the whole series. The method we propose is different from MWPCA approaches mentioned earlier, where our method uses all observation to calculate one moving cross-covariance matrix. Using the moving cross-covariance matrix enables our method to extract static and dynamic information from series that are allowed to be non-stationarity. Therefore, it will be called the moving dynamic principal component analysis (MDPCA).

There are other methods that divide analyze time series into windows in order to analyze the data such as Multivariate Singular Spectrum Analysis (MSSA). Even though both MSSA and MDPCA aim to produce complex components that consist of dynamic dependence in the data, the two methods have different purposes. MDPCA aims to reduce the dimension of a multivariate time series by seeking fewer uncorrelated principal components with directions which can explain most of variation of the original data. MSSA on the other hand decomposes a time series into a number of components (i.e. elements such as trend, periodic and random noise) then reconstructs data by selecting important elements that contain the dynamic information of the original series; See for example, Golyandina et al. (2001), Broomhead and King (1986) and Hossein and Rahim (2013).

This paper is arranged as follows. Section 2 reveals the building-structure of MDPCA, along with a new proposed diagnostic tool to evaluate the relationship between the retained components. Additionally, a criterion for determining the number of final retained components is proposed. Section 3 shows the theoretical properties of our estimators. In Sect. 4, the ability of MDPCA to dimension reduction is examined on both simulated and real data. We also reveal the R packages that consist of the necessary functions used to produce and assess MDPCA's results. Section 5 states concluding remarks and suggested problems for further research.

2 Methodology

Consider an m-dimensional time series $\mathbf{z}_t = (z_{1,t}, z_{2,t}, \ldots, z_{m,t})'$, which is allowed to be non-stationary. The initial step in the MDPCA method is to build an $m(l+1)$-dimensional extended data vector, denoted by $\mathbf{y}_t$, which consists of the series $\mathbf{z}_t$ and its lagged series up to a pre-specified lag l. Then the extended data vector $\mathbf{y}_t$ is going to have the following structure

$$\begin{aligned}
\mathbf{y}_t &= (\mathbf{z}'_{t+l}, \mathbf{z}'_{t+l-1}, \ldots \mathbf{z}'_t)' \\
&= (z_{1,t+l}, \ldots z_{m,t+l}, z_{1,t+l-1}, \ldots z_{m,t+l-1}, \ldots, z_{1,t}, \ldots z_{m,t})'.
\end{aligned}$$

The rest of the analyses will be performed on $\mathbf{y}_t$ instead of $\mathbf{z}_t$. Assume the series $\mathbf{z}_t$ is observed at T time points. Let $m(l+1) = M$ and $T - l = N$. Let $\mathbf{Y}$ be an $M \times N$

extended data matrix whose columns are $\mathbf{y}_1, \ldots, \mathbf{y}_N$. A critical feature of the extended data vector $\mathbf{y}_t$ is that its cross-covariance matrix will account for the dynamic relations that exist among the components (i.e., variables) of $\mathbf{z}_t$. This idea was first introduced to PCA by Ku et al. (1995) to reduce the dimension of dynamic data, while PCA is limited to static data.

For a stationary series, the DPCA of Ku et al. (1995) applies its analysis to the cross-covariance matrix of $\mathbf{y}_t$ to reduce the dimension of $\mathbf{z}_t$. However, for a non-stationary series, the results of the DPCA would not be valid as it assumes the first two moments to be constant for all time points. Furthermore, if DPCA is applied to a non-stationary series, it could produce correlated dynamic principal components (i.e., DPCs). This is mainly because the cross-covariance matrix will not be able to measure the dynamic dependence between the variables of non-stationary series. One solution we propose is to use moving cross-covariance matrices instead. These matrices will allow the capture of dynamic relations among the components of non-stationary time series because they can be updated as we move in time. Define the cross-covariance matrix to be

$$\Gamma_t = \text{Var}(\mathbf{y}_t). \tag{1}$$

Once $\mathbf{z}_t$ is observed, the sample cross-covariance matrix defined over window i with a pre-specified size $W = 2w + 1$ can be calculated as follows

$$\hat{\Gamma}_i = \frac{1}{2w+1} \sum_{t=i-w}^{i+w} (\mathbf{y}_t - \bar{\mathbf{y}}_i)(\mathbf{y}_t - \bar{\mathbf{y}}_i)' \tag{2}$$

where

$$\bar{\mathbf{y}}_i = \frac{1}{2w+1} \sum_{t=i-w}^{i+w} \mathbf{y}_t$$

where w is a positive integer. Then, the moving cross-covariance matrix is defined as

$$\mathbf{M}\Gamma = \frac{1}{N-2w} \sum_{i=w+1}^{N-w} \Gamma_i \tag{3}$$

where Γ_i is the cross-covariance matrix defined over window i of $\mathbf{y}_t$. The building structure of $\mathbf{M}\Gamma$ will make it more suitable to measure the dynamic dependence between non-stationary series' components as it collects its information from the cross-covariance matrices defined over the updated local windows of $\mathbf{y}_t$. In specific, the first cross-covariance matrix is calculated over the first window, then the second cross-covariance matrix is calculated over the second window (i.e. by including the next time point and excluding the oldest one), and so on. Then the moving cross-covariance matrix uses all these cross-covariance matrices to extract the dynamic

dependence from $\mathbf{y}_t$ as a whole. Based on sample data, $\mathbf{M}\Gamma$ can be estimated bulging in $\hat{\Gamma}_i$ into Eq. (3) as

$$\hat{\mathbf{M}\Gamma} = \frac{1}{N - 2w} \sum_{i=w+1}^{N-w} \hat{\Gamma}_i. \tag{4}$$

Note that the moving cross-covariance matrix $\mathbf{M}\Gamma$ is an $M \times M$ symmetric matrix which has a spectral decomposition as follows:

$$\mathbf{M}\Gamma = \mathbf{U}\Lambda\mathbf{U}'.$$

Correspondingly, the sample moving cross-covariance matrix $\hat{\mathbf{M}\Gamma}$ has the following spectral decomposition:

$$\hat{\mathbf{M}\Gamma} = \hat{\mathbf{U}}\hat{\Lambda}\hat{\mathbf{U}}' \tag{5}$$

where $\hat{\mathbf{U}}$ is an $M \times M$ orthogonal matrix whose columns are the eigenvectors of $\hat{\mathbf{M}\Gamma}$ and $\hat{\Lambda}$ is an $M \times M$ diagonal matrix consists of the eigenvalues of $\hat{\mathbf{M}\Gamma}$ along its diagonal. Let $\hat{\lambda}_j$, $1 \leq j \leq M$, be the jth eigenvalue of $\hat{\mathbf{M}\Gamma}$ (i.e., $\hat{\lambda}_j$ is the (j, j)th element of $\hat{\Lambda}$), where $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \cdots \geq \hat{\lambda}_M$. Let $\hat{\mathbf{u}}_j$ be the corresponding eigenvector (i.e., $\hat{\mathbf{u}}_j$ is the jth column of $\hat{\mathbf{U}}$). MDPCA reduces the dimension of $\mathbf{z}_t$ by producing M uncorrelated moving dynamic principal components (MDPCs) and transform $\mathbf{y}_t$ into a space with dimension $k < m$ such that

$$\frac{\hat{\lambda}_1 + \cdots + \hat{\lambda}_k}{\hat{\lambda}_1 + \cdots + \hat{\lambda}_M} \simeq 1. \tag{6}$$

Here, the value of k also indicates the number of MDPCs being used to reconstruct the data. The optimal value for k is the minimum number of MDPCs that consist of the maximum variation of the data and the minimum reconstruction error. More details about determining the optimal choice of k will be provided in the next section.

Remark 1 Averaging the local sample variance–covariance matrices in Eq. (2) will formulate the estimation of the global average variance–covariance matrix (i.e. the sample moving cross-covariance matrix) in Eq. (4). The aim of this procedure is to allow non-stationarity while measuring variation and cross-covariation. This procedure can be carried out by following a few steps. First, we determine the window size (i.e. W) based on the degree of stationarity of the data. Then calculate the first local sample variance–covariance matrix based on the observations from times 1 to W. Then calculate the second local sample variance–covariance matrix based on the observations from times 2 to $W + 1$, and so on. Averaging these local sample variance–covariance matrices produce a global smoothed covariance matrix that consists of the dynamic dependence of the original series.

2.1 Optimizing MDPCA's results

In order to improve the results of MDPCA, one would choose optimal values for the window size W, the number of lags l to include in the extended data vector, and the number of retained MDPCs.

Choosing a size for W is vital to enhance the results of MDPCA and extract accurate information from the data. The size of W depends on the degree of stationarity of the data. Small window sizes will be more suitable for data that exhibit strong non-stationarity. Hence, determining a size for W can be done by looking at the time series plot and assessing the stationarity of the data. More analyses on determining the size of W will be conducted in the simulations section of this article. Notice that MDPCA can be applied on both stationary and non-stationary series by adjusting the window size, as mention earlier. Therefore, DPCA is a special case of MDPCA where $W = N$.

In the following, we are going to provide a procedure to determine the optimal size for l. Additionally, a new criterion will be proposed in order to objectively determine the number of optimal components (i.e. MDPCs) to retain.

2.1.1 Choosing optimal number of lags

Including more lagged series can provide more dynamic information to the analysis; however, it would also increase the dimension of $\mathbf{y}_t$, which makes the analysis more complicated. Therefore, one would include only lagged series that provide more dynamic information related to the original series in order to gain accurate results with the lowest dimension possible for $\mathbf{y}_t$. In order to choose an optimal value for l, we are going to adapt the procedure suggested by Ku et al. (1995), which can be summarized as follows:

1. Start with $l = 0$.
2. Build the extended data vector $\mathbf{y}_t$ by including l lagged series.
3. Apply MDPCA to $\mathbf{y}_t$ and obtain all MDPCs.
4. Set $j = m(l + 1)$ and $r(l) = 0$ where r is the number of relations.
5. Determine if the jth MDPC provides a linear relation. If yes, go to next step, otherwise go to step 7.
6. Set $j = j - 1$ and $r = r(l) + 1$, then repeat step 5.
7. Calculate the number of new relations by:

$$r_{\text{new}}(l) = r(l) - \sum_{k=0}^{l-1}(l - k + 1)r_{\text{new}}(k)$$

if $r_{\text{new}}(l) \leq 0$, go to step 9, otherwise go to next step.
8. Set $l = l + 1$, go to step 2.
9. Stop.

The above steps assumed the size of W to be given or already determined. The number of significant MDPCs can be determined by examining the plot of the eigenvalues of $\mathbf{\hat{M}\Gamma}$. Then, the number of relations r can be obtained by subtracting the number
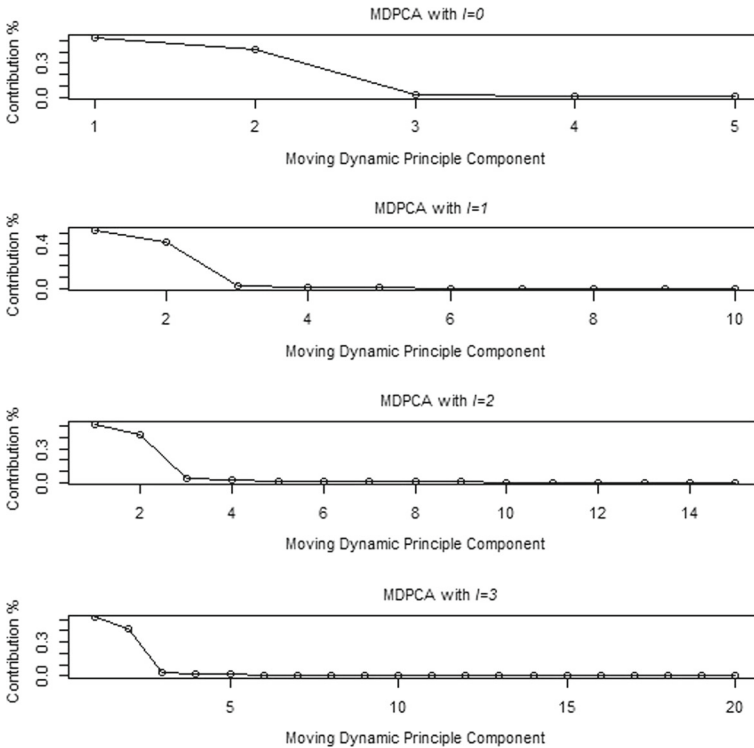
Fig. 1 Eigenvalues' contribution rate plots after applying MDPCA to a series of five variables with three static and two dynamic relations

of significant MDPCs from the total number of variables (i.e. $r = M-$ number of significant MDPCs).

Here we provide a diagram example to clarify the idea behind the above procedure. Suppose we consider a series $\mathbf{z}_t$ which consists of five variables that have some relationships among them. Assume the contribution rate plot of eigenvalues after applying MDPCA with 0, 1, 2, and 3 lags are shown in Fig. 1. Then three static relations are found when MDPCA with $l = 0$ is applied because only two MDPCs have significantly high contribution rates. Notice that by including each lag, new relations might be detected, and the previous relations will be repeated $(l + 1)$ times. By applying MDPCA with $l = 1$, eight relations are found, which are the three static relations repeated twice and two new dynamic relations that are exposed by including the first lagged series. By using MDPCA with $l = 2$, 13 relations are found, which are the three static relations repeated three times, and two dynamic relations repeated twice. Hence, no new relations are found, and the procedure would suggest not including more lags as $l = 1$ is the optimal choice.

Notice that in the above example, if these five variables are independent and do not have any relationship between them, then all MDPCs resulted from applying MDPCA will be significant. Therefore, no relation will be detected in this case (i.e. $r = 0$).

2.1.2 Retained component criterion (RCC)

Once W and l are already determined, and MDPCA is applied to $\mathbf{y}_t$, then the next task is to choose the optimal number of MDPCs to retain, k. This can be done by balancing between the following desires: maximizing the percentage of explained variance, minimizing the MSE (i.e., mean of squared error) of reconstructing the original data, and reducing the dimension of the series. The percentage of explained variance can be measured as given in Eq. (6). The following equation calculates the MSE of reconstructed data by the first k MDPCs:

$$\text{MSE}_k \left(\mathbf{y}_t \ \mathbf{y}_t^{\text{recon}} \right) = \frac{1}{MN} \sum_{j=1}^{M} \sum_{t=1}^{N} \left(y_{j,t} - \sum_{v=1}^{k} \hat{u}_{j,v} C_{v,t} \right)^2 \tag{7}$$

where $\mathbf{y}_t^{\text{recon}}$ is the reconstructed data by the first k MDPCs, $\hat{u}_{j,v}$ is the vth element of the jth eigenvector of $\hat{\mathbf{M}}\Gamma$ and $C_{v,t}$ is the tth observation of the vth MDPC (C_v) which can be obtained by left multiplying $\mathbf{y}_t$ by the transpose of the first v columns of $\hat{\mathbf{U}}$. Notice that choosing more MDPCs will increase the percentage of explained variance and reduce the MSE of reconstructed data; however, it will also increase the final dimension. Therefore, our goal here is to retain the minimum number of MDPCs that explain most of the variation and have minimum reconstruction error. In literature, this is usually done subjectively by balancing between the above desires. To this end, we are going to propose a criterion that can balance between the above desires and objectively suggests the optimal number of MDPCs to retain. This criterion will be called the retained component criterion (RCC).

In order to determine the optimal number of MDPCs, we need to measure the effect of adding each MDPC on the accuracy of the final results of MDPCA, where maximum accuracy can be achieved by explaining all variations in the original data and reducing the MSE of reconstructed data to zero. Notice, we are going to assume that both the percentage of explained variance and the MSE of reconstructed data are equally important to measure the accuracy of MDPCA's results.

Consider the case where a time series $\mathbf{y}_t$ with dimension M is observed. Assume an ideal case where all M variables are independent and equally important to explain the variability in the data. In particular, all variables consist of equally important information and contribute equally to the variation in $\mathbf{y}_t$. Then after applying MDPCA to $\mathbf{y}_t$, we expect that each MDPC will equally explain $1/M\%$ of the total variation of $\mathbf{y}_t$ and reduce the MSE of reconstructing the data by an equal amount of $1/M\%$. Therefore, each MDPC will improve the accuracy of the final results of MDPCA by $2/M\%$. The reason behind assuming an ideal case and giving the components of $\mathbf{y}_t$ equal weights is to include an objective penalty term in our criterion for retaining an extra MDPC in the final results. Before we move further, consider the following definition. Let MaxMSE be the maximum MSE of reconstructing data defined by

$$\text{MaxMSE} = \frac{1}{MN} \sum_{j=1}^{M} \sum_{t=1}^{N} (y_{j,t})^2. \tag{8}$$

Notice that MaxMSE is equivalent to the MSE of reconstructing data with no MDPCs available and replacing elements of $\mathbf{y}_t^{\text{recon}}$ in (7) by zeros.

Then, the RCC criterion of the first k MDPCs is defined as

$$\text{RCC}_k = 2 - \left(\sum_{j=1}^{k} \hat{\lambda}_j / \sum_{j=1}^{M} \hat{\lambda}_j \right) - ((\text{MaxMSE} - \text{MSE}_k)/\text{MaxMSE}) + (2k/M) \quad (9)$$

where $\hat{\lambda}_j$ is the largest jth eigenvalue of the matrix $\hat{\mathbf{M}}\Gamma$ defined in (3), MaxMSE is defined in (8), and MSE_k is given in (7). The RCC criterion consists of three main terms: the term $(\sum_{j=1}^{k} \hat{\lambda}_j / \sum_{j=1}^{M} \hat{\lambda}_j)$, which represents the percentage of explained variance by first k MDPCs, the term $((\text{MaxMSE} - \text{MSE}_k)/\text{MaxMSE})$, which represents the percentage of reduced MSE by the first k MDPCs, and the term $(2k/M)$, which is a penalty for retaining k MDPCs. The constant "2" is included in the calculation in (9) to retain positive values for the RCC criterion. This is a technical reason as the constant value will not change the final decision of the RCC criterion. The optimal number of MDPCs to retain is the number corresponding to the minimum RCC value in (9). Furthermore, the RCC criterion can be used to determine the optimal number of components in most of the PCA-based reduction methods (e.g., classical PCA and DPCA).

For example, consider a series $\mathbf{y}_t$ with dimension $M = 8$. After applying MDPCA, if the first MDPC explains 50% of the total variation of $\mathbf{y}_t$ (i.e., $\sum_{j=1}^{k} \hat{\lambda}_j / \sum_{j=1}^{M} \hat{\lambda}_j = 0.5$) and reduces MaxMSE by 85% (i.e., $(\text{MaxMSE} - \text{MSE}_k)/\text{MaxMSE} = 0.85$), then the RCC criterion will have a value of $2 - 0.5 - 0.85 + 0.25 = 0.9$. Now, let the second MDPC explains 40% of the total variation of $\mathbf{y}_t$ and reduces MaxMSE by 10%, then the RCC criterion of the first two MDPCs will have a value of $2 - (0.5 + 0.4) - (0.85 + 0.1) + (0.25 + 0.25) = 0.65$. Hence, adding the second component will contribute significantly to increase the accuracy of MDPCA's results. Additionally, if the third MDPC explains 5% of the total variation of $\mathbf{y}_t$ and reduces MaxMSE by 3%, then the RCC criterion of the first three MDPCs will have a value of $2 - (0.5 + 0.4 + 0.05) - (0.85 + 0.1 + 0.03) + (0.25 + 0.25 + 0.25) = 0.87$, which means that adding the third MDPC will increase the accuracy by a non-significant amount. This can be explained as the penalty of using the third MDPC is larger than the amount of accuracy added to MDPCA's results. Hence, for this example, the optimal number of retained MDPCs will be 2, as it has the lowest RCC of 0.65.

2.2 MDPCA calculation procedure

The following is a summary of the steps of MDPCA:

1. Create the extended data vector $\mathbf{y}_t$ by including lagged series up to lag l.
2. Calculate the moving cross-covariance matrix $\hat{\mathbf{M}}\Gamma$ based on $\mathbf{y}_t$.
3. Calculate the eigenvalues and the corresponding eigenvectors of $\hat{\mathbf{M}}\Gamma$.
4. Use the RCC criterion to determine k, the optimal number of MDPCs to retain.
5. Left multiplying $\mathbf{y}_t$ by the first kth columns of the matrix $\hat{\mathbf{U}}'$ defined in (5) produces the transformed data with reduced dimension.

2.3 Evaluating dynamic relationships between MDPCs

For stationary series, examining for a significant correlation between the variables of a multivariate time series can be done by visualizing tools such as the cross-correlation plots, which is a generalization of the autocorrelation function plot (ACF) of Box and Jenkins (1976) to the multivariate time series. Methods involving testing the significance of the multiple null hypotheses exist in literature such as the multivariate portmanteau statistic; See Hosking (1980). However, the methods mentioned above were developed to capture the dynamic dependence of stationary series and would not be meaningful for non-stationary series because they use the classical correlation function with a fixed mean throughout the calculations. Methods such as co-integration searches for stationary linear combinations of non-stationary series. However, co-integration is concerned with the long-run relationships between non-stationary variables; See Engle and Granger (1987), Johansen (1995). To this end, we need to extend some of the methods mentioned above to find correlated components or variables of non-stationary series by using a bit different measurement of correlation that can be updated as we move forward or backward in time. Hence, we propose the use of a moving cross-correlation function. This function will be used to check whether two non-stationary variables are correlated. It will also be used to evaluate the relationship between MDPCs. Before we proceed further, the following definitions are needed. Define the lag l cross-covariance matrix of $\mathbf{y}_i$ as

$$\Gamma_i(l) = \text{Cov}(\mathbf{y}_i, \mathbf{y}_{i-l}). \tag{10}$$

Also, define the lag l cross-correlation matrix of $\mathbf{y}_i$ to be

$$\rho_i(l) = \text{Corr}(\mathbf{y}_i, \mathbf{y}_{i-l}) = \mathbf{S}_i^{-1} \, \Gamma_i(l) \, \mathbf{S}_{i-l}^{-1} \tag{11}$$

where l is a non-negative integer, $\Gamma_i(l)$ is defined in (10) and $\mathbf{S}_i$ is the diagonal matrix of the standard deviations of $\mathbf{y}_i$. The (j, j)th element of $\mathbf{S}_i$ is the square root of the (j, j)th element of $\Gamma_i(0)$ defined over $\mathbf{y}_i$. The above functions can be estimated using the following formulas as follows. The sample lag l cross-covariance matrix over window $\mathbf{y}_i$ with a pre-specified size of $W = 2w + 1$ will be calculated using

$$\hat{\Gamma}_i(l) = \frac{1}{2w + 1} \sum_{t=i-w}^{i+w} (\mathbf{y}_t - \bar{\mathbf{y}}_i)(\mathbf{y}_{t-l} - \bar{\mathbf{y}}_i)' \tag{12}$$

where

$$\bar{\mathbf{y}}_i = \frac{1}{2w + 1} \sum_{t=i-w}^{i+w} \mathbf{y}_t.$$

Then, $\hat{\Gamma}_i(l)$ defined in (12) can be used to calculate the sample lag l moving cross-correlation matrix over the window $\mathbf{y}_i$, $\hat{\rho}_i(l)$, to estimate $\rho_i(l)$ as

$$\hat{\rho}_i(l) = \hat{\mathbf{S}}_i^{-1} \, \hat{\Gamma}_i(l) \, \hat{\mathbf{S}}_{i-l}^{-1} \tag{13}$$

where the (j, j)th element of $\hat{\mathbf{S}}_i$ is the square root of the (j, j)th element of $\widehat{\Gamma}_i(0)$ defined over the same window $\mathbf{y}_i$. Further, define the lag l moving cross-correlation matrix of the series $\mathbf{y}_t$ to be

$$\mathbf{M}\rho(l) = \frac{1}{N - 2\max(l, w)} \sum_{i=\max(l,w)+1}^{N-\max(l,w)} \rho_i(l). \tag{14}$$

Based on sample data, we can estimate $\mathbf{M}\rho(l)$ using the sample lag l moving cross-correlation matrix as follows

$$\hat{\mathbf{M}}\rho(l) = \frac{1}{N - 2\max(l, w)} \sum_{i=\max(l,w)+1}^{N-\max(l,w)} \hat{\rho}_i(l). \tag{15}$$

Notice that $\hat{\mathbf{M}}\rho(l)$ will be updated at each time point as we move in time to account for non-stationary series.

Based on the above-stated definitions, both visualization and multiple hypotheses testing methods can be developed to check for the significance of correlations between the components of either stationary or non-stationary series. For visualization, one can plot the sample moving cross-correlation matrices with different time lags $l = 0, \pm 1, \pm 2, \ldots \pm p$; where p is a positive integer taken to be $p = 10 \log_{10}(N/M)$, similarly to those in ACF plot. The significance of the correlation can be evaluated by looking at the 95% confidence interval computed using $\pm 1.96/\sqrt{N}$. We demonstrate the use of the moving cross-correlation function in the following examples.

Example 1 This example is a short simulation study to test the ability of the moving cross-correlation plots to capture the dynamic relationship among different variables of a multivariate time series. A window with size 101 will be used in the calculation of the moving cross-correlation function. The results then will be compared with those based on the cross-correlation function.

The simulated data in this example consists of eight variables and a sample of size 1200, where three different non-stationary models were used to generate three sub-series of 4, 3, and 1 variable as described below. Let a_t, b_t, and c_t be three independent standard normal white noises, which are the innovation terms of the following three models, respectively, then:

$$\begin{cases} y_{j,t} = u_{t+j-1}, & j = 1, 2, 3 \text{ and } 4 \\ y_{j,t} = v_{t+j-4}, & j = 5, 6 \text{ and } 7 \\ y_{j,t} = w_t, & j = 8 \end{cases} \tag{16}$$

where $(u_t, v_t, w_t)^{'}$ satisfies

$$\begin{cases} \bigtriangledown u_t = 0.6 \bigtriangledown u_{t-1} + a_t + 0.5a_{t-1} \\ \bigtriangledown v_t = b_t + 0.8b_{t-1} - 2.2b_{t-2} \\ \bigtriangledown w_t = -0.55 \bigtriangledown w_{t-1} - 0.5 \bigtriangledown w_{t-2} + 0.4 \bigtriangledown w_{t-3} + c_t \end{cases} \tag{17}$$

A time series plot of the simulated multivariate time series is available in Fig. 2, where all variables exhibit non-stationary behaviours. First, we examine the sample cross-correlation plots (i.e., using the classical cross-correlation function) of the data; See Figs. 3, 4, 5, and 6. Based on these plots, a strong dynamic relationship exists among the eight variables, which implies that all three simulated subgroups are strongly correlated. The last result contradicts with the way that we simulated the data. Therefore, the cross-correlation plots could lead to non-correct results when dealing with non-stationary series.

On the other hand, the sample moving cross-correlation plots of the simulated series are provided in Figs. 7, 8, 9, and 10. Three uncorrelated subgroups of 4, 3, 1 variable are detected, where the variables in each subgroup are strongly correlated.

The main reason that we obtained different results using the two visualization methods above is the non-stationarity nature of the data. Therefore, we can conclude that the moving cross-correlation plots can capture the dynamic relationship between non-stationary series. Furthermore, it can be shown that the above two methods will produce similar conclusions when applied to stationary series.

3 Theoretical properties

To show the reliability of the results obtained by the proposed MDPCA, we shall prove the consistency of the estimated MDPCs, which are generated by left multiplying the extended data matrix by $\hat{\mathbf{U}}^{'}$ in (5). Therefore, we shall show that $\hat{\mathbf{U}}$ is a consistent estimator of $\mathbf{U}$. We are going to approach the consistency by showing that $D(\mathscr{M}(\hat{\mathbf{U}}), \mathscr{M}(\mathbf{U})) \to 0$ as $W \to \infty$. Here, W is the window size used in the calculation of MDPCA, $\mathscr{M}(\mathbf{U})$ is the linear space spanned by $\mathbf{U}$'s columns, and $D(\mathscr{M}(\hat{\mathbf{U}}), \mathscr{M}(\mathbf{U}))$ is the distance between the spaces $\mathscr{M}(\hat{\mathbf{U}})$ and $\mathscr{M}(\mathbf{U})$. For ease of notation, we are going to use $c, c_1, c_2, \ldots$ to denote constants whose values might be different from place to place.

For two positive integers $c_1 < c_2$, let $\mathbf{B}_1$ and $\mathbf{B}_2$ be any $c_2 \times (c_2 - c_1)$ matrices satisfying the condition $\mathbf{B}_i^{'}\mathbf{B}_i = \mathbf{I}_{(c_2-c_1) \times (c_2-c_1)}$, where $i \in \{1, 2\}$. Define the distance between the $\mathbf{B}_1$ and $\mathbf{B}_2$ to be

$$D(\mathbf{B}_1, \mathbf{B}_2) = \sqrt{c_2 - c_1 - \text{tr}(\mathbf{B}_1\mathbf{B}_1^{'}\mathbf{B}_2\mathbf{B}_2^{'})}. \tag{18}$$

Notice that $D(\mathbf{B}_1, \mathbf{B}_2) = 0$ if and only if $\mathscr{M}(\mathbf{B}_1) = \mathscr{M}(\mathbf{B}_2)$. This measurement was applied by Pan and Yao (2008), Chang et al. (2018).

It is important to know that the convergence of $\hat{\mathbf{U}}$ is implied by the convergence of $\hat{\mathbf{M}}\Gamma$ in (4). This can be seen since $\hat{\mathbf{U}}$ consists of the eigenvectors of $\hat{\mathbf{M}}\Gamma$. The

fact that $\hat{\mathbf{M}\Gamma}$ is calculated based on moving windows whose width depends on the stationarity of the data makes the moving cross-covariance function more complicated than the ordinary cross-covariance function. However, the consistency of $\hat{\mathbf{M}\Gamma}$ still can be reached as each window in $\hat{\mathbf{M}\Gamma}$ is calculated as in the stationary case. Therefore, the convergence of the estimated moving principal components MDPCs will depend on the size of W. Recall that $W \leq N$, where $W = N$ when data is stationary, and W gets smaller as the data becomes more non-stationary.

In the following work, we approach the consistency assuming the dimension M to be fixed. The needed conditions will be stated. Moreover, since time series data is known to be dependent data, we are going to consider the following measurement of dependence:

$$\theta_l = \sup_c \sup_{A \in \mathscr{F}_{-\infty}^c, B \in \mathscr{F}_{c+l}^\infty} |P(A \cap B) - P(A)P(B)| \tag{19}$$

where $\mathscr{F}_{c_3}^{c_4}$ is the σ-field generated by $\mathbf{y}_t$ for $c_3 \leq t \leq c_4$. This measurement of dependence [i.e., θ_l in (19)] is called the mixing coefficients in literature and $\theta_l = 0$ if the time series is a sequence of independent random variables. This measurement indicates that the two data observed at two time points, which are l times apart, are going to be independent as $l \to \infty$. More information on the use of the mixing coefficients can be found in Bradley (1986).

Assumption 1 Assume that

$$\sup_i \max_{1 \leq j \leq M} E|y_{j,i} - E(y_{j,i})|^{2q}$$

to be upper bounded by a positive constant c for some constant $q > 2$, where $y_{j,i}$ is the jth component of $\mathbf{y}_i$.

Assumption 2 Assume that

$$\sum_{l=1}^\infty \theta_l^{1-2/q} < \infty$$

for q defined in Assumption 1, where θ_l is the mixing coefficient defined in (19).

Theorem 1 *Assume the dimension M is fixed. Let Assumptions 1 and 2 hold. Then, for $\hat{U} = (\hat{U}_1, \ldots, \hat{U}_M)$ obtained by (5), it is true that*

$$\max_{1 \leq j \leq M} D(\mathscr{M}(\hat{U}_j), \mathscr{M}(U_j)) \xrightarrow{P} 0$$

as $W \to \infty$, where "$\xrightarrow{P}$" means convergence in probability.

The proof of Theorem 1 will be provided in the "Appendix".

4 Simulations and real data examples

In this section, we are going to test the ability of the proposed method on both real and simulated multivariate time series data. The following examples will focus on non-stationary series. Recall, MDPCA and DPCA produce identical results when applied to stationary data since MDPCA uses a window of size $W = N$ in the stationary case. The performance of MDPCA will be assessed by considering the percentage of explained variance (i.e., contribution percentage), the MSE of reconstructed data, and the moving cross-correlation plots of the retained MDPCs. All analyses are done using R software. The needed functions to produce and assess MDPCA's results can be found under the following R packages (i.e. libraries): `MACF` of Alshammri (2020a), `MCOV` of Alshammri (2020b), `MDPCA` of Alshammri (2020c) and `RCC_MDPCA` of Alshammri (2020d).

4.1 Simulations

In the following simulation studies, we are going to apply MDPCA with different combinations of window and lag sizes on simulated datasets of different dimensions and sample sizes. Each simulation will be replicated 500 times. Data sets will be generated using `arima.sim` command in R.

Example 2 In this example, we apply the MDPCA on a non-stationary series $\mathbf{z}_t$ with ten variables. This example consists of two parts. The first part studies the results of MDPCA when using different combinations of W and l. The second part compares the effect of the size of T on MDPCA's results. The series $\mathbf{z}_t$ is generated using five different models, such that each model produces two correlated variables as follows.

Let a_t, b_t, c_t, d_t and e_t be independent standard normal white noises, which are the innovation terms of the following five models, respectively, then:

$$\begin{cases} z_{j,t} = u_{t+j-1}, & j = 1 \text{ and } 2 \\ z_{j,t} = v_{t+j-3}, & j = 3 \text{ and } 4 \\ z_{j,t} = w_{t+j-5}, & j = 5 \text{ and } 6 \\ z_{j,t} = x_{t+j-7}, & j = 7 \text{ and } 8 \\ z_{j,t} = q_{t+j-9}, & j = 9 \text{ and } 10 \end{cases} \tag{20}$$

where $(u_t, v_t, w_t, x_t, q_t)'$ satisfies

$$\begin{cases} \nabla u_t = 0.75 \nabla u_{t-1} + a_t + 0.9a_{t-1} \\ \nabla v_t = 0.6 \nabla v_{t-1} + b_t - 1.4b_{t-1} \\ \nabla w_t = -0.7 \nabla w_{t-1} + c_t - 2.3c_{t-1} \\ \nabla x_t = -0.5 \nabla x_{t-1} + d_t + 0.55d_{t-1} \\ \nabla q_t = 0.6 \nabla q_{t-1} + e_t + 1.65e_{t-1} \end{cases} \tag{21}$$

Table 1 A comparison between the results of MDPCA with different options of W and l in Example 2

Results	Options of W and l		
	$W = 101$ and $l = 1$	$W = 201$ and $l = 1$	$W = 301$ and $l = 1$
Contribution %	96.48%	97.01%	97.34%
Mean (SD)	(0.01)	(0.01)	(0.01)
MSE	359.27	350.18	309.09
Mean (SD)	(234.25)	(246.62)	(226.98)
	$W = 101$ and $l = 5$	$W = 201$ and $l = 5$	$W = 301$ and $l = 5$
Contribution %	94.59%	96.11%	96.71%
Mean (SD)	(0.01)	(0.01)	(0.01)
MSE	403.74	335.89	333.04
Mean (SD)	(288.54)	(221.02)	(213.36)

A time series plot of the simulated data is shown in Fig. 11, where it can be seen that every two variables represent a different non-stationary model. First, we would like to see the results of MDPCA with different options of W and l. Based on 500 replicas, Table 1 is a comparison between the results of MDPCA with different sizes of W and l when applied to the simulated series $\mathbf{z}_t$ with 1500 samples, where two MDPCs are considered. For the mean percentages of explained variance by the two MDPCs, the percentages obtained by using $l = 1$ are higher than those obtained by using $l = 5$. However, the percentages differ by a small amount, which can be justified as using more lagged data (i.e., $l = 5$) can include more information to the analysis. For example, two MDPCs, on average, explain 96.48% of the variance of the data when using MDPCA with $W = 101$ and $l = 1$, compared with 94.59% when using $W = 101$ and $l = 5$. The standard errors of explained variance in all cases are 0.01, which indicates steady percentages in all replicas. For the mean of MSE of reconstructed data, it ranges between 309.09 and 403.74, where it has its lowest when $W = 301$ and $l = 1$ are used, and it has its highest when $W = 101$ and $l = 5$ are used.

Furthermore, the dynamic dependence between the two MDPCs can be revealed by plotting the means of the absolute value of the moving cross-correlation with $W = 101$; See Figs. 12, 13 and 14 and the corresponding standard errors in Table 6. There are no dynamic relationships between the two MDPCs for both cases when using $W = 101$. However, the correlations get slightly larger and cross the significance line as we increase the window size from $W = 201$ to $W = 301$.

Therefore, based on the above discussion, the dimension of the simulated non-stationary series with ten variables in this example can be best reduced by using MDPCA with $W = 101$ and $l = 1$ or MDPCA with $W = 101$ and $l = 5$.

Second, we would like to see the results of MDPCA when the series $\mathbf{z}_t$ has different sample sizes (i.e., $T = 200, 400, 600,$ and 800). In this part, we are considering MDPCA with $W = 101$ and $l = 5$. Based on 500 replicas, the results of applying MDPCA on different sizes when simulating $\mathbf{z}_t$ are shown in Table 2, where two MDPCs are considered. For the mean percentage of explained variance, the highest percentage

Table 2 A comparison between the results of MDPCA with different sample sizes in Example 2

Results	MDPCA ($W = 101, l = 5$)			
	$T = 200$	$T = 400$	$T = 600$	$T = 800$
Contribution%	96.68%	94.54%	94.61%	94.67%
Mean (SD)	(0.03)	(0.02)	(0.01)	(0.01)
MSE	52.68	95.24	148.57	191.59
Mean (SD)	(37.04)	(66.27)	(87.82)	(125.03)

of 96.68% was obtained when $T = 200$, then decreased to 94.54% when $T = 400$, then followed by a continued increase to 94.61% when $T = 600$ and 94.67% when $T = 800$. The standard errors of the percentage of explained variance decreased from 0.03 when $T = 200$ to 0.01 when $T = 600$, then stay at this value for larger sample sizes. This indicates a more steady performance for $T \geq 600$. For the mean of MSE of reconstructed data, its value has a positive relationship with T. For example, the mean MSE was 52.68 when $T = 200$, then increased to 191.59 when $T = 800$. The MSE values are not steady for each replica as they have large standard errors (e.g., 66.27 standard error when $T = 400$).

On the other hand, the means of the absolute value of the moving cross-correlation with $W = 101$ indicates a significant dynamic relationship between the two MDPCs when $T = 200$; See Fig. 15. The correlations then decreased between the two MDPCs when $T = 400$ with some minor significant cross-correlations. For the cases where $T = 600$ and 800, the plots indicate uncorrelated MDPCs; See Fig. 16. Also, Table 7 shows the standard errors of the absolute value of the moving cross-correlation. We notice an improvement on the errors when increasing the sample size of the data. For example, the standard errors range between 0 and 0.24 when $T = 200$, comparing with 0–0.12 when $T = 800$. Notice by applying MDPCA to the simulated data with $T = 600$ we are able to obtain similar results to those with $T = 1500$. Therefore, even though we increased the dimension of $\mathbf{z}_t$ to $m = 10$, MDPCA still performs well on data with moderate sample sizes.

Example 3 In the following simulation study, MDPCA is applied to a non-stationary series $\mathbf{z}_t$ that consists of 15 variables. This study consists of two parts. The first part compares the MDPCA's results when using different combinations of W and l. The second part compares the effect of the size of T on MDPCA's results. The data is generated using five different models, such that each model produces three correlated variables.

Let a_t, b_t, c_t, d_t and e_t be independent standard normal white noises, which are the innovation terms of the following five models, respectively, then:

Table 3 A comparison between the results of MDPCA with different options of W and l in Example 3

Results	Options of W and l		
	$W = 101$ and $l = 1$	$W = 201$ and $l = 1$	$W = 301$ and $l = 1$
Contribution%	96.13%	96.85%	97.22%
Mean (SD)	(0.01)	(0.01)	(0.01)
MSE	348.56	331.78	314.06
Mean (SD)	(230.96)	(223.55)	(207.18)
	$W = 101$ and $l = 5$	$W = 201$ and $l = 5$	$W = 301$ and $l = 5$
Contribution%	94.43%	95.90%	96.54%
Mean (SD)	(0.01)	(0.01)	(0.01)
MSE	364.54	345.85	327.43
Mean (SD)	(254.41)	(227.12)	(221.58)

$$
\begin{cases}
z_{j,t} = u_{t+j-1}, & j = 1, 2 \text{ and } 3 \\
z_{j,t} = v_{t+j-4}, & j = 4, 5 \text{ and } 6 \\
z_{j,t} = w_{t+j-7}, & j = 7, 8 \text{ and } 9 \\
z_{j,t} = x_{t+j-10}, & j = 10, 11 \text{ and } 12 \\
z_{j,t} = q_{t+j-13}, & j = 13, 14 \text{ and } 15
\end{cases}
\tag{22}
$$

where $(u_t, v_t, w_t, x_t, q_t)'$ satisfies

$$
\begin{cases}
\nabla u_t = 0.75 \, \nabla u_{t-1} + a_t + 0.9a_{t-1} \\
\nabla v_t = 0.6 \, \nabla v_{t-1} + b_t - 1.4b_{t-1} \\
\nabla w_t = -0.7 \, \nabla w_{t-1} + c_t - 2.3c_{t-1} \\
\nabla x_t = -0.5 \, \nabla x_{t-1} + d_t + 0.55d_{t-1} \\
\nabla q_t = 0.6 \, \nabla q_{t-1} + e_t + 1.65e_{t-1}
\end{cases}
\tag{23}
$$

Based on 500 replicas, Table 3 shows the results of MDPCA with different combinations of W and l applied to $\mathbf{z}_t$ with 2000 samples where two MDPCs are considered. It can be seen that the mean percentages of the explained variance by the two MDPCs increase slightly as we increase W. For example, the mean percentage is 96.13% when using MDPCA with $W = 101$ and $l = 1$ then increases to 97.22% when using MDPCA with $W = 301$ and $l = 1$. Also, the percentages are slightly lower when using more lagged data (i.e., when increasing l). For example, the mean percentage is 96.85% when using MDPCA with $W = 201$ and $l = 1$ then decreases to 95.90% when using MDPCA with $W = 201$ and $l = 5$. The standard error of the percentage of explained variance has a small value of 0.01 in all cases. Additionally, the MSE of reconstructed data has large standard errors, which means that its value can be small or large, depending on the data. For example, the mean MSE is 359.27 when using MDPCA with $W = 101$ and $l = 1$, and has a standard error of 234.25.

Additionally, the plot of the means of the absolute value of the moving cross-correlation with $W = 101$ indicates no dynamic relationships between the two components when using MDPCA with $W = 101$; See Fig. 17. However, minor, but significant, correlations between the two components for all cases where MDPCA is used with $W = 201$; See Fig. 18. The correlations become slightly larger when using MDPCA with $W = 301$; See Fig. 19. Small standard errors of the absolute value of the moving cross-correlation are reported in Table 8, where their values range between 0 to 0.08 in all cases.

Based on the above results, we can best reduce the dimension of the simulated non-stationary series with 15 variables in this example by using MDPCA with $W = 101$ and $l = 1$, or MDPCA with $W = 101$ and $l = 5$.

To see the effect of changing the sample size of $\mathbf{z}_t$ on the MDPCA, we are going to apply MDPCA with $W = 101$ and $l = 1$ on $\mathbf{z}_t$ with sample sizes $T = 200, 400, 600,$ and 800. The results based on 500 replicas are summarized in Table 4, where two MDPCs are considered. Similar values, about 96.25%, for the mean percentage of explained variance are obtained in all cases with small standard errors. For example, a mean percentage of 96.35% and a standard error of 0.02 are obtained when $T = 200$, compared with 96.22% and 0.01 when $T = 800$. The mean MSE of reconstructed data and the corresponding standard error get larger as we increase T. For example, a mean MSE of 42.66 with a standard error of 28.68 is obtained when $T = 200$, compared with 183.39 and 129.85 when $T = 800$.

Additionally, the means of the absolute value of the moving cross-correlation with $W = 101$ are plotted in Figs. 20 and 21. It can be seen that the two MDPCs have significant cross-correlations when $T = 200$. The correlations become smaller with minor significant correlations for $| l | \geq 10$ when increasing the dimension to $T = 400$. Uncorrelated MDPCs are obtained when $T = 600$ and $T = 800$. The standard errors of the absolute value of the moving cross-correlation are reported in Table 9, where the values can be improved by increasing the sample size. For instance, the standard errors range between 0 and 0.23 when $T = 200$ comparing with 0–0.11 when $T = 800$. Therefore, in this example, MDPCA performs well when applied to $\mathbf{z}_t$ with $T \geq 600$. The MDPCA was able to reduce the dimension of $\mathbf{z}_t$ from 15 to 2.

By the end of the above simulation studies, we conclude that the proposed MDPCA is able to reduce the dimension of a multivariate time series by taking into account both the dynamic and non-stationarity behaviors of the data. It was noticed that the performance of MDPCA was steady even though we increased the dimension of the tested series.

4.2 Real data examples

Example 4 In this example, the MDPCA will be applied to daily stock prices of 17 USA companies in US Dollars from November 07, 2013 to December 18, 2017. The sample size of the data is 1036 days. The names of the 17 companies are shown in Table 5. The data is accessible on Yahoo! Finance.

Table 4 A comparison between the results of MDPCA with different sample sizes in Example 3

Results	MDPCA ($W = 101, l = 1$)			
	$T = 200$	$T = 400$	$T = 600$	$T = 800$
Contribution%	96.35%	96.25%	96.23%	96.22%
Mean (SD)	(0.02)	(0.01)	(0.01)	(0.01)
MSE	42.66	91.74	130.54	183.39
Mean (SD)	(28.68)	(63.65)	(92.59)	(129.85)

The time series plots of the daily stock prices of the 17 companies reveal the non-stationarity behavior of the daily stock prices; See Figs. 22 and 23. Figure 24 shows the last 36 sample moving cross-correlation plots with $W = 101$ of the stock prices before the transformation. The daily stock prices of the 17 companies are moderately correlated. For example, the company MetLife Insurance is strongly correlated with Prudential Financial and weakly correlated with McKesson.

By looking at Figs. 22 and 23, MDPCA with $W = 101$ will be used. The optimal number of lags is $l = 1$, as shown in Fig. 25, where ten static and seven dynamic relations were found. Figure 26 consists of an eigenvalues' plot along with the relative RCC criterion plot. The RCC has values of 0.508, 0.490, 0.489, 0.486, and 0.502 for the first three, four, five, six, and seven MDPCs receptively. Thus, the optimal number of MDPCs to retain is six MDPCs, as suggested by the RCC criterion. The six MDPCs explain 90.25% of the total variation in the data and produce a reconstruction error of 304.55. The retained six MDPCs are uncorrelated, as shown in the sample moving cross-correlation plots in Fig. 27.

To conclude, MDPCA with $W = 101$ and $l = 1$ was able to reduce the dimension of the daily stock prices of the USA companies from 17 to 6 by accounting for the non-stationarity and the dynamic dependence in the stock prices. Notice that MDPCA was applied directly to the original stock prices. This will prevent any loss of information caused by dealing with, for example, the log return of the prices.

5 Concluding remarks

In this paper, we introduced MDPCA, which is a PCA-based dimension reduction method that is used to reduce the dimension of multivariate time series data by transforming them into uncorrelated components. MDPCA is a generalization of DPCA of Ku et al. (1995) to non-stationary series. DPCA can be considered as a special case of MDPCA when $W = N$.

We used three methods to assess MDPCA's results. The moving cross-correlation function which evaluates the dynamic relationships between the final retained MDPCs. The MSE of reconstructed data. The percentage of explained variance.

Table 5 Names and labels of the 17 USA companies in Example 4

No.	Label	Name	No.	Label	Name
1	ESRX	Express scripts holding	10	KR	Kroger-The Kroger
2	FB	Facebook	11	LOW	Lowe's
3	F	Ford motor	12	MCK	McKesson
4	GE	General electric	13	MET	MetLife Insurance
5	GM	General motor health	14	MSFT	Microsoft
6	HD	The home depot	15	PEP	PepsiCo
7	IBM	IBM	16	PG	The Procter & Gamble
8	INTC	Intel	17	PRU	Prudential Financial
9	JNJ	Johnson & Johnson			

Choosing the window size for MDPCA depends on the stationarity of the data. Shorter windows are suitable for series with strong non-stationary behavior, and the opposite is true. Determining an optimal window size for MDPCA can be a subject for further research.

The RCC criterion is a new tool to determine the optimal number of MDPCs to retain. This criterion balances between the following two desires, reducing the dimension of the data and increasing the accuracy of the final results. Additionally, the RCC criterion can be employed to determine the optimal number of retained components in PCA-based reduction methods.

The asymptotic properties of our estimator $\hat{\mathbf{U}}$, the matrix that consists of the eigenvectors of the moving cross-covariance matrix of the data, are studied. Under some regularity assumptions, we show that $\hat{\mathbf{U}}$ is a consistent estimator of $\mathbf{U}$ with $W^{-1/2}$ convergence rate.

We carried out many simulations considering non-stationary series with different dimensional and sample sizes. MDPCA was able to reach dimensional reduction and performs well even for reasonably small sample sizes (i.e., $T = 400$). A real data example was used to confirm the results of the simulations.

Declarations

Appendix A: Proofs

The proof of Theorem 1 is shown here. The following lemma of Chang et al. (2018) is useful to state our results.

Lemma 1 *Assume for $\gamma > 2$, $E(|y_{j,t} - \mu_j|^{2\gamma})$ is uniformly bounded away from infinity for $j = 1, \ldots, p$, and $t \geq 1$. Let the mixing coefficients*

$$\theta_l = \sup_j \sup_{A \in \mathscr{F}_{-\infty}^j, B \in \mathscr{F}_{j+l}^\infty} (|P(A \cap B) - P(A)P(B)|)$$

satisfy the condition $\sum_{l=1}^\infty \theta_l^{1-2/\gamma} < \infty$. Let the dimension M is fixed, then $\|\hat{\sum}_y(l) - \sum_y(l)\|_2 = O_p(1/\sqrt{T})$ for each $l \leq l_1$, where l_1 is a pre-described positive integer number.

The following lemma is based on Lemma 1.

Lemma 2 *Let the Assumptions 1 and 2 hold. Also, if we assume the dimension M to be fixed, then for each i, $\|\hat{\Gamma}_i - \Gamma_i\|_2 = O_p(1/\sqrt{W})$ as $W \to \infty$, where Γ_i and $\hat{\Gamma}_i$ are defined in (1) and (2), respectively.*

Proof of Lemma 2 By assuming Assumptions 1 and 2, then Lemma 1 can be applied on each window, and we have

$$\|\hat{\Gamma}_i(l) - \Gamma_i(l)\|_2 = O_p(1/\sqrt{W})$$

for the lagged covariance matrices up to any specified lag l_1. Notice that MDPCA is only considering cross-covariance with no lags (i.e., $\Gamma_i = \Gamma_i(0)$), thus the following is true

$$\|\hat{\Gamma}_i - \Gamma_i\|_2 = O_p(1/\sqrt{W}).$$

□

The following lemma is based on the results from Lemma 2.

Lemma 3 *Assume the dimension M is fixed, and under the Assumptions 1 and 2, then the following holds as $W \to \infty$,*

$$\|\hat{M}\Gamma - M\Gamma\|_2 \xrightarrow{P} 0,$$

where $M\Gamma$ and $\hat{M}\Gamma$ are defined in (3) and (4), respectively.

Proof of Lemma 3 By assuming the dimension M to be fixed, and under Assumptions 1 and 2, from Lemma 2, we have

$$\|\hat{\Gamma}_i - \Gamma_i\|_2 = O_p(1/\sqrt{W})$$

for each i, which implies

$$\|\hat{\Gamma}_i - \Gamma_i\|_2 \xrightarrow{P} 0$$

for each i, as $W \to \infty$. Therefore,

$$\|\hat{\mathbf{M}}\Gamma - \mathbf{M}\Gamma\|_2 = \|\frac{1}{N-2w} \sum_{i=w+1}^{N-w} \hat{\Gamma}_i - \frac{1}{N-2w} \sum_{i=w+1}^{N-w} \Gamma_i\|_2$$

$$\leq \frac{1}{N-2w} \sum_{i=w+1}^{N-w} \|\hat{\Gamma}_i - \Gamma_i\|_2 \xrightarrow{P} 0.$$

Proof of Theorem 1 Assume the dimension M is fixed. Let the Assumptions 1 and 2 hold. By applying the results of theorem 8.1.10 of Golub and Van Loan (1996), we have

$$\max_{1 \leq j \leq M} D(\mathcal{M}(\hat{\mathbf{U}}_j), \mathcal{M}(\mathbf{U}_j)) = O_p(\|\hat{\mathbf{M}}\Gamma - \mathbf{M}\Gamma\|_2)$$

Therefore, from Lemma 3, it holds that, as $W \to \infty$,

$$\max_{1 \leq j \leq M} D(\mathcal{M}(\hat{\mathbf{U}}_j), \mathcal{M}(\mathbf{U}_j)) \xrightarrow{P} 0.$$

Appendix B: Tables and figures

See Tables 6, 7, 8, 9 and Figs. 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26 and 27.

Table 6 Standard deviations of the absolute value of the moving cross-correlations between two MDPCs with different options of W and l in Example 2

lag	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
MDPCA ($W = 101, l = 1$)																					
MDPC1 versus MDPC1	0.00	0.00	0.00	0.01	0.01	0.02	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.06	0.06	0.06	0.07	0.07	0.07	0.07	0.08
MDPC1 versus MDPC2	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.05	0.05	0.05	0.05	0.05
MDPC2 versus MDPC1	0.03	0.03	0.03	0.03	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.06
MDPC2 versus MDPC2	0.00	0.00	0.01	0.01	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.06	0.06	0.07	0.07	0.07	0.07	0.08	0.08	0.08	0.08
MDPCA ($W = 201, l = 1$)																					
MDPC1 versus MDPC1	0.00	0.00	0.01	0.01	0.02	0.02	0.03	0.03	0.04	0.04	0.04	0.05	0.05	0.05	0.05	0.06	0.06	0.06	0.07	0.07	0.07
MDPC1 versus MDPC2	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05
MDPC2 versus MDPC1	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05
MDPC2 versus MDPC2	0.00	0.00	0.01	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.05	0.06	0.06	0.06	0.07	0.07	0.07	0.08	0.08	0.08	0.08
MDPCA ($W = 301, l = 1$)																					
MDPC1 versus MDPC1	0.00	0.00	0.01	0.01	0.02	0.02	0.03	0.03	0.04	0.04	0.04	0.05	0.05	0.05	0.06	0.06	0.06	0.07	0.07	0.07	0.07
MDPC1 versus MDPC2	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06
MDPC2 versus MDPC1	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06
MDPC2 versus MDPC2	0.00	0.00	0.01	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.05	0.06	0.06	0.06	0.07	0.07	0.07	0.07	0.08	0.08	0.08
MDPCA ($W = 101, l = 5$)																					
MDPC1 versus MDPC1	0.00	0.00	0.00	0.01	0.01	0.02	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.06	0.06	0.06	0.07	0.07	0.07	0.08	0.08
MDPC1 versus MDPC2	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.06
MDPC2 versus MDPC1	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.06
MDPC2 versus MDPC2	0.00	0.00	0.00	0.01	0.01	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.06	0.06	0.06	0.07	0.07	0.07	0.08	0.08	0.08
MDPCA ($W = 201, l = 5$)																					
MDPC1 versus MDPC1	0.00	0.00	0.00	0.01	0.01	0.02	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.06	0.06	0.06	0.07	0.07	0.07	0.07	0.08
MDPC1 versus MDPC2	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.06	0.06	0.06

Table 6 continued

lag	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
MDPC2 versus MDPC1	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.06
MDPC2 versus MDPC2	0.00	0.00	0.00	0.01	0.01	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.06	0.06	0.06	0.07	0.07	0.07	0.07	0.08	0.08
MDPCA ($W = 301, l = 5$)																					
MDPC1 versus MDPC1	0.00	0.00	0.01	0.01	0.01	0.02	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.05	0.06	0.06	0.06	0.06	0.07	0.07	0.07
MDPC1 versus MDPC2	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06
MDPC2 versus MDPC1	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06
MDPC2 versus MDPC2	0.00	0.00	0.01	0.01	0.01	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.06	0.06	0.06	0.06	0.07	0.07	0.07	0.07	0.07

Table 7 Standard deviations of the absolute value of the moving cross-correlations between two MDPCs with different sample sizes in Example 2

lag	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$T = 200$																					
MDPC1 versus MDPC1	0.00	0.00	0.01	0.02	0.03	0.04	0.06	0.07	0.09	0.11	0.12	0.14	0.15	0.16	0.18	0.19	0.20	0.21	0.22	0.23	0.24
MDPC1 versus MDPC2	0.06	0.06	0.06	0.07	0.07	0.08	0.09	0.10	0.11	0.11	0.12	0.13	0.13	0.14	0.15	0.15	0.16	0.16	0.17	0.17	0.18
MDPC2 versus MDPC1	0.06	0.06	0.06	0.07	0.07	0.08	0.09	0.10	0.10	0.11	0.12	0.13	0.14	0.15	0.15	0.16	0.17	0.17	0.18	0.18	0.19
MDPC2 versus MDPC2	0.00	0.00	0.02	0.03	0.05	0.08	0.10	0.13	0.15	0.16	0.18	0.19	0.20	0.19	0.19	0.18	0.17	0.17	0.17	0.17	0.16
$T = 400$																					
MDPC1 versus MDPC1	0.00	0.00	0.01	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10	0.11	0.12	0.12	0.13	0.14	0.14	0.15	0.15
MDPC1 versus MDPC2	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.07	0.07	0.07	0.08	0.08	0.08	0.09	0.09	0.10	0.10	0.11	0.11	0.11	0.12
MDPC2 versus MDPC1	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.07	0.07	0.08	0.08	0.08	0.08	0.09	0.09	0.10	0.10	0.11	0.11	0.11
MDPC2 versus MDPC2	0.00	0.00	0.01	0.02	0.03	0.05	0.06	0.08	0.09	0.10	0.11	0.12	0.13	0.14	0.15	0.15	0.16	0.16	0.16	0.16	0.16
$T = 600$																					
MDPC1 versus MDPC1	0.00	0.00	0.00	0.01	0.02	0.03	0.03	0.04	0.05	0.06	0.07	0.07	0.08	0.08	0.09	0.10	0.10	0.11	0.11	0.12	0.12
MDPC1 versus MDPC2	0.05	0.05	0.05	0.05	0.05	0.05	0.06	0.06	0.06	0.06	0.07	0.07	0.07	0.08	0.08	0.08	0.08	0.09	0.09	0.09	0.09
MDPC2 versus MDPC1	0.05	0.05	0.05	0.05	0.05	0.05	0.06	0.06	0.06	0.06	0.07	0.07	0.07	0.07	0.08	0.08	0.08	0.09	0.09	0.09	0.09
MDPC2 versus MDPC2	0.00	0.00	0.01	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.09	0.10	0.11	0.11	0.11	0.12	0.12	0.12	0.13
$T = 800$																					
MDPC1 versus MDPC1	0.00	0.00	0.00	0.01	0.01	0.02	0.03	0.04	0.04	0.05	0.06	0.06	0.07	0.08	0.08	0.09	0.09	0.10	0.10	0.10	0.11
MDPC1 versus MDPC2	0.04	0.04	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.06	0.06	0.06	0.06	0.06	0.06	0.07	0.07	0.07	0.07	0.07	0.08
MDPC2 versus MDPC1	0.04	0.04	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.06	0.06	0.06	0.06	0.07	0.07	0.07	0.07	0.08	0.08
MDPC2 versus MDPC2	0.00	0.00	0.01	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.08	0.09	0.08	0.10	0.10	0.11	0.11	0.11	0.11	0.12

Table 8 Standard deviations of the absolute value of the moving cross-correlations between two MDPCs with different options of W and l in Example 3

Lag	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
MDPCA ($W = 101, l = 1$)																					
MDPC1 versus MDPC1	0.00	0.00	0.00	0.01	0.01	0.02	0.02	0.03	0.04	0.04	0.04	0.05	0.05	0.06	0.06	0.06	0.07	0.07	0.07	0.08	0.08
MDPC1 versus MDPC2	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.05	0.05	0.05	0.05
MDPC2 versus MDPC1	0.03	0.03	0.03	0.03	0.03	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.06
MDPC2 versus MDPC2	0.00	0.00	0.01	0.01	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.06	0.06	0.07	0.07	0.07	0.07	0.08	0.08	0.08	0.08
MDPCA ($W = 201, l = 1$)																					
MDPC1 versus MDPC1	0.00	0.00	0.00	0.01	0.01	0.02	0.03	0.03	0.04	0.04	0.04	0.05	0.05	0.06	0.06	0.06	0.07	0.07	0.07	0.07	0.08
MDPC1 versus MDPC2	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05
MDPC2 versus MDPC1	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05
MDPC2 versus MDPC2	0.00	0.00	0.01	0.01	0.02	0.02	0.03	0.04	0.04	0.05	0.05	0.06	0.06	0.07	0.07	0.07	0.07	0.07	0.08	0.08	0.08
MDPCA ($W = 301, l = 1$)																					
MDPC1 versus MDPC1	0.00	0.00	0.00	0.01	0.02	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.05	0.06	0.06	0.06	0.07	0.07	0.07	0.07	0.07
MDPC1 versus MDPC2	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06
MDPC2 versus MDPC1	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06
MDPC2 versus MDPC2	0.00	0.00	0.01	0.01	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.06	0.06	0.07	0.07	0.07	0.07	0.08	0.08	0.08	0.08
MDPCA ($W = 101, l = 5$)																					
MDPC1 versus MDPC1	0.00	0.00	0.00	0.01	0.01	0.02	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.05	0.06	0.06	0.06	0.07	0.07	0.07	0.07
MDPC1 versus MDPC2	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.05	0.05	0.05	0.05	0.05	0.05
MDPC2 versus MDPC1	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05
MDPC2 versus MDPC2	0.00	0.00	0.01	0.01	0.02	0.02	0.03	0.04	0.04	0.05	0.05	0.06	0.06	0.06	0.07	0.07	0.07	0.08	0.08	0.08	0.08
MDPCA ($W = 201, l = 5$)																					
MDPC1 versus MDPC1	0.00	0.00	0.00	0.01	0.01	0.02	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.05	0.06	0.06	0.06	0.07	0.07	0.07	0.07
MDPC1 versus MDPC2	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.06

Table 8 continued

Lag	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
MDPC2 versus MDPC1	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05
MDPC2 versus MDPC2	0.00	0.00	0.01	0.01	0.01	0.02	0.03	0.03	0.04	0.05	0.05	0.06	0.06	0.07	0.07	0.07	0.07	0.07	0.07	0.08	0.08
MDPCA ($W = 301, l = 5$)																					
MDPC1 versus MDPC1	0.00	0.00	0.01	0.01	0.01	0.02	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.06	0.06	0.06	0.06	0.07	0.07	0.07	0.08
MDPC1 versus MDPC2	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06
MDPC2 versus MDPC1	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06
MDPC2 versus MDPC2	0.00	0.00	0.01	0.01	0.01	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.06	0.06	0.06	0.07	0.07	0.07	0.07	0.08	0.08

Table 9 Standard deviations of the absolute value of the moving cross-correlations between two MDPCs with different sample sizes in Example 3

Lag	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
T = 200																					
MDPC1 versus MDPC1	0.00	0.00	0.01	0.02	0.03	0.04	0.06	0.07	0.09	0.10	0.11	0.13	0.14	0.16	0.17	0.18	0.19	0.20	0.21	0.22	0.23
MDPC1 versus MDPC2	0.05	0.05	0.06	0.06	0.07	0.08	0.09	0.09	0.10	0.11	0.12	0.12	0.13	0.14	0.15	0.15	0.16	0.17	0.17	0.18	0.18
MDPC2 versus MDPC1	0.05	0.05	0.05	0.06	0.06	0.07	0.08	0.09	0.10	0.11	0.11	0.12	0.13	0.14	0.15	0.15	0.16	0.17	0.17	0.18	0.18
MDPC2 versus MDPC2	0.00	0.01	0.02	0.04	0.07	0.09	0.12	0.14	0.16	0.17	0.18	0.19	0.19	0.20	0.20	0.19	0.19	0.18	0.18	0.17	0.17
T = 400																					
MDPC1 versus MDPC1	0.00	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.06	0.07	0.08	0.09	0.10	0.11	0.11	0.12	0.13	0.13	0.14	0.14	0.15
MDPC1 versus MDPC2	0.06	0.06	0.06	0.06	0.06	0.06	0.07	0.07	0.08	0.08	0.08	0.09	0.09	0.09	0.10	0.10	0.10	0.11	0.11	0.11	0.12
MDPC2 versus MDPC1	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.07	0.07	0.08	0.08	0.08	0.09	0.09	0.09	0.09	0.10	0.10	0.10	0.10	0.11
MDPC2 versus MDPC2	0.00	0.00	0.01	0.02	0.03	0.05	0.06	0.07	0.09	0.10	0.11	0.12	0.13	0.13	0.14	0.14	0.15	0.15	0.15	0.15	0.15
T = 600																					
MDPC1 versus MDPC1	0.00	0.00	0.01	0.01	0.02	0.03	0.04	0.04	0.05	0.06	0.07	0.07	0.08	0.09	0.09	0.10	0.10	0.11	0.11	0.12	0.12
MDPC1 versus MDPC2	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.06	0.06	0.06	0.06	0.06	0.07	0.07	0.07	0.08	0.08	0.08	0.08	0.09	0.09
MDPC2 versus MDPC1	0.05	0.05	0.05	0.05	0.05	0.05	0.06	0.06	0.06	0.06	0.06	0.06	0.07	0.07	0.07	0.07	0.07	0.07	0.08	0.08	0.08
MDPC2 versus MDPC2	0.00	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10	0.10	0.11	0.11	0.12	0.12	0.12	0.12	0.12	0.13
T = 800																					
MDPC1 versus MDPC1	0.00	0.00	0.01	0.01	0.02	0.02	0.03	0.04	0.04	0.05	0.06	0.06	0.07	0.07	0.08	0.08	0.09	0.09	0.10	0.10	0.10
MDPC1 versus MDPC2	0.04	0.04	0.04	0.04	0.05	0.05	0.05	0.05	0.05	0.06	0.05	0.06	0.06	0.06	0.06	0.06	0.06	0.07	0.07	0.07	0.07
MDPC2 versus MDPC1	0.04	0.04	0.04	0.04	0.05	0.05	0.05	0.05	0.05	0.05	0.06	0.06	0.06	0.06	0.06	0.06	0.07	0.07	0.07	0.07	0.08
MDPC2 versus MDPC2	0.00	0.00	0.01	0.02	0.02	0.03	0.04	0.05	0.06	0.07	0.07	0.08	0.08	0.09	0.09	0.09	0.10	0.10	0.10	0.11	0.11

Fig. 2 A time series plot of the simulated series with 8 variables
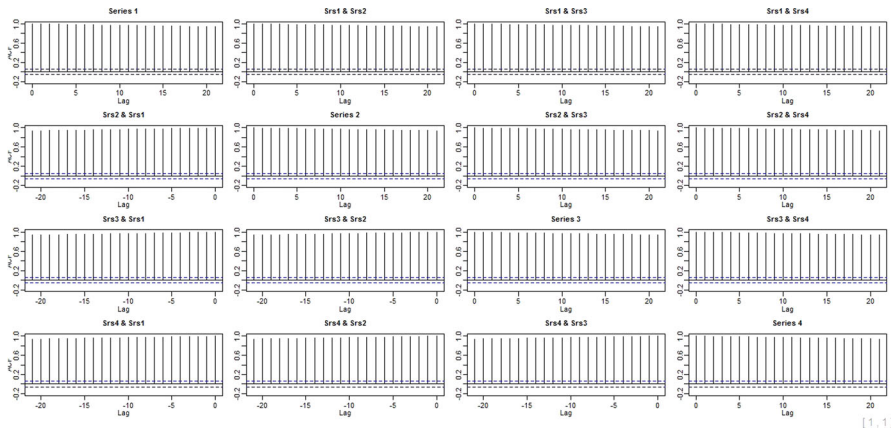
Fig. 3 Sample cross-correlation plots of the simulated series with 8 variables—part 1
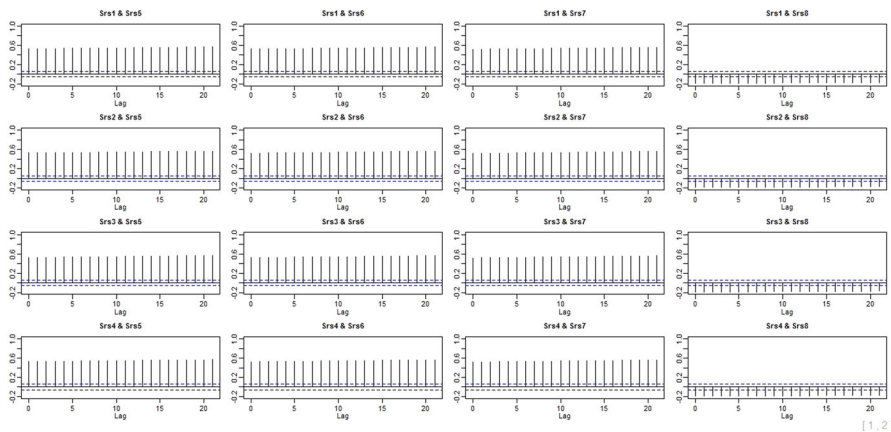
Fig. 4 Sample cross-correlation plots of the simulated series with 8 variables—part 2
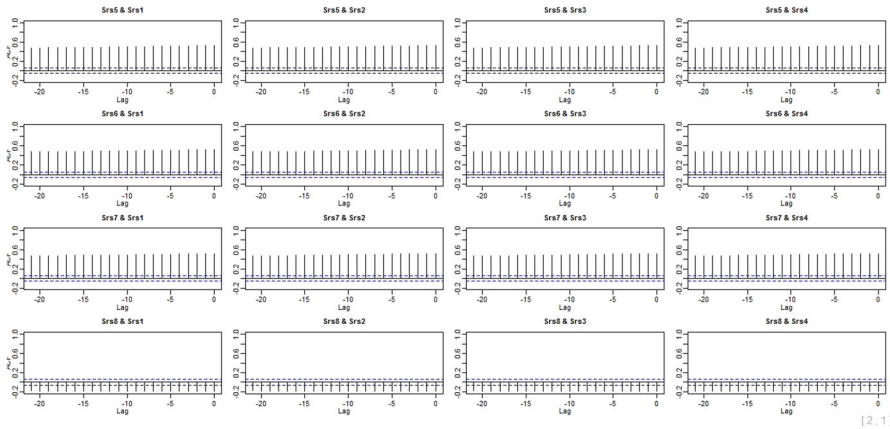
Fig. 5 Sample cross-correlation plots of the simulated series with 8 variables—part 3
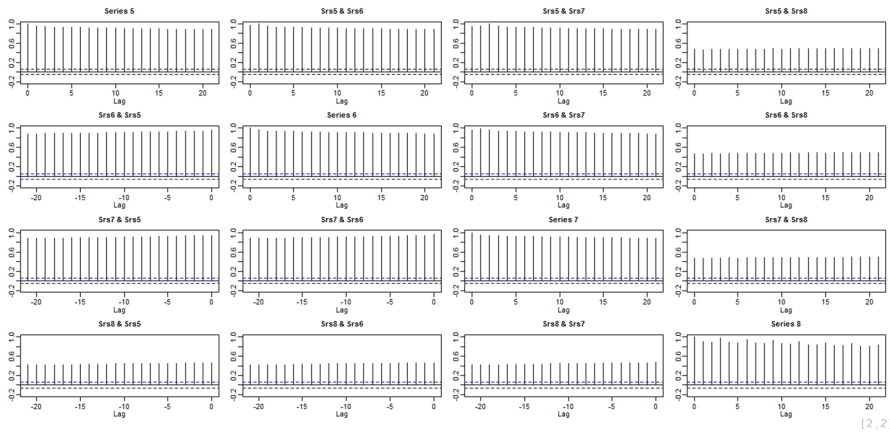
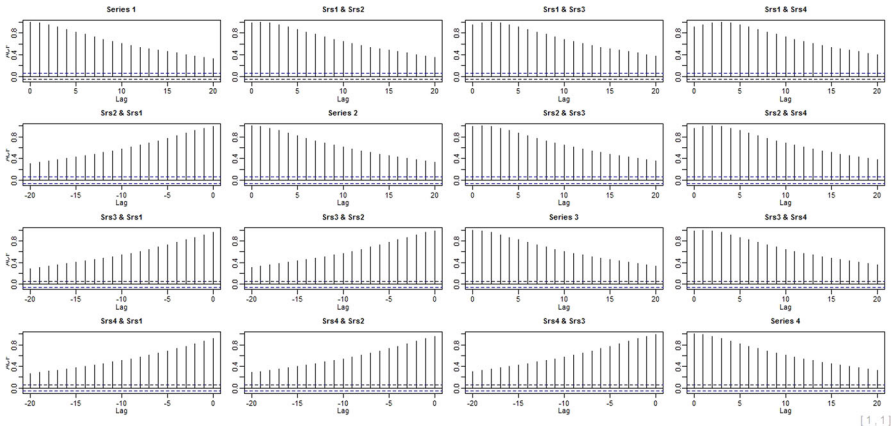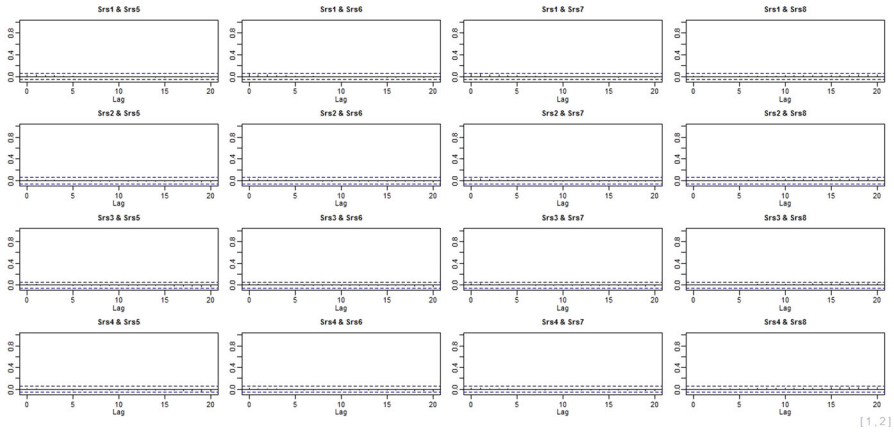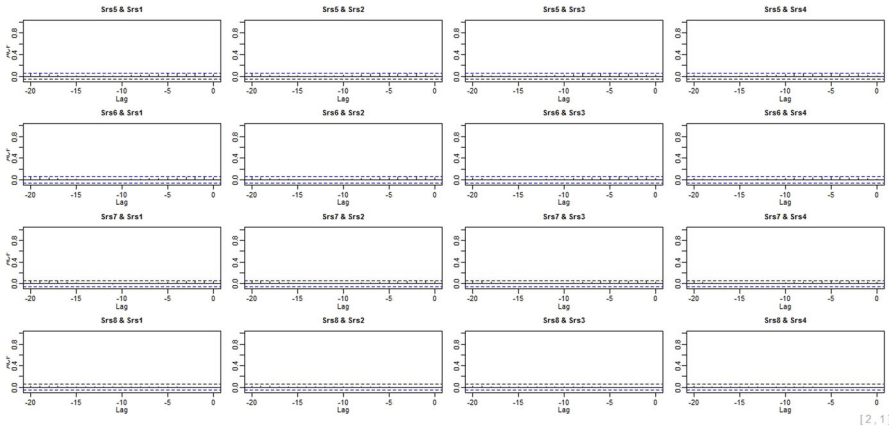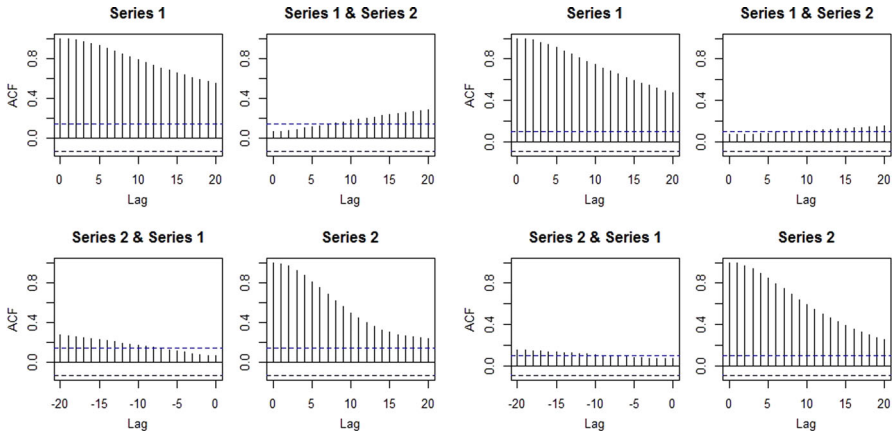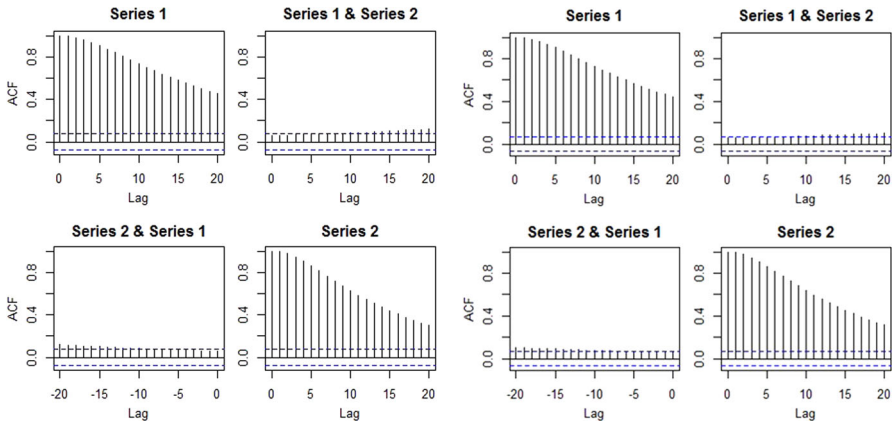**Fig. 6** Sample cross-correlation plots of the simulated series with 8 variables—part 4

Fig. 7 Sample moving cross-correlation plots with a window size 101 of the simulated series with 8 variables—part 1

Fig. 8 Sample moving cross-correlation plots with a window size 101 of the simulated series with 8 variables—part 2

Fig. 9 Sample moving cross-correlation plots with a window size 101 of the simulated series with 8 variables—part 3
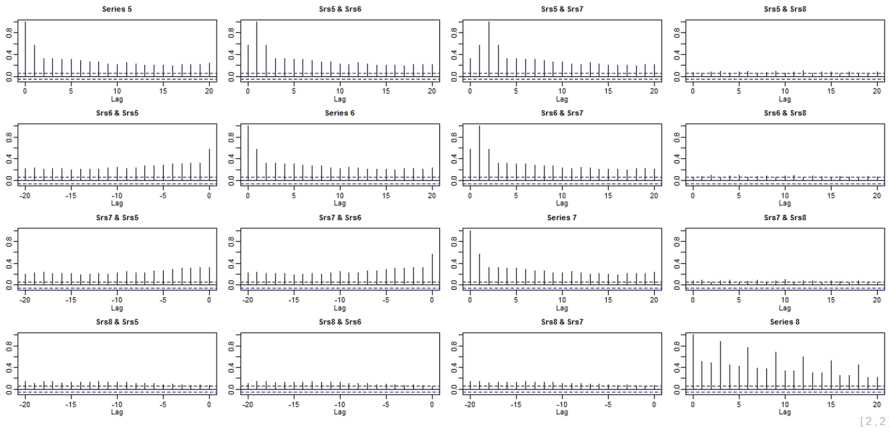
Fig. 10 Sample moving cross-correlation plots with a window size 101 of the simulated series with 8 variables—part 4
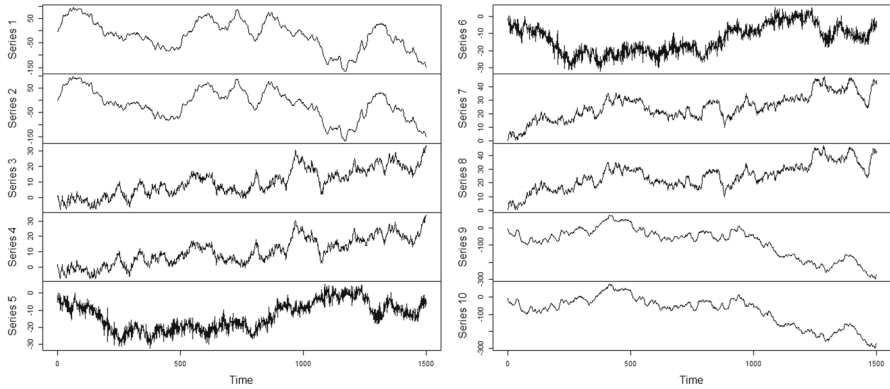
Fig. 11 A time series plot of the simulated series with 10 variables in Example 2
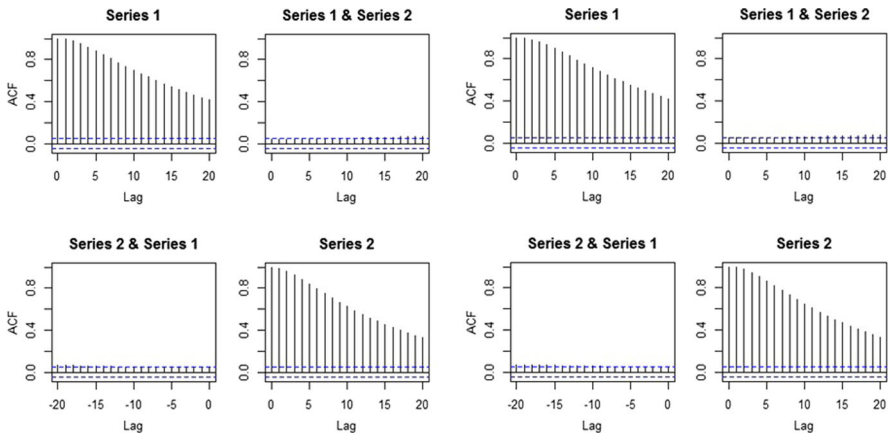
Fig. 12 Plots of the mean of the absolute value of the moving cross-correlations between two MDPCs in Example 2 [Left: MDPCA ($W = 101, l = 1$), Right: MDPCA ($W = 101, l = 5$)]
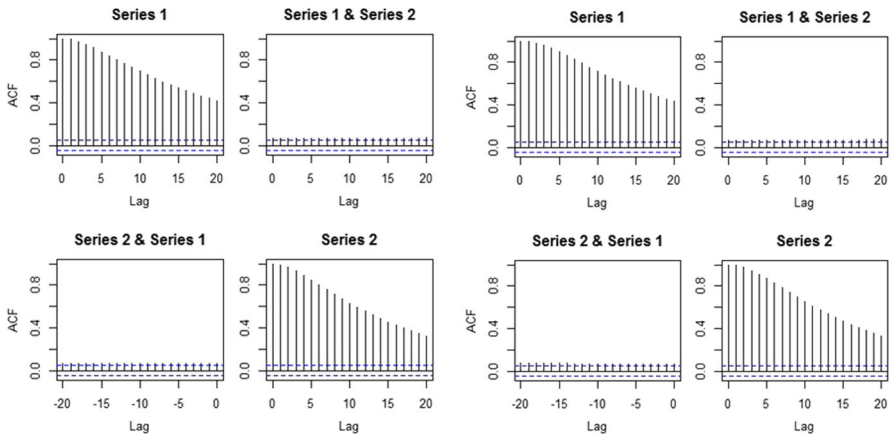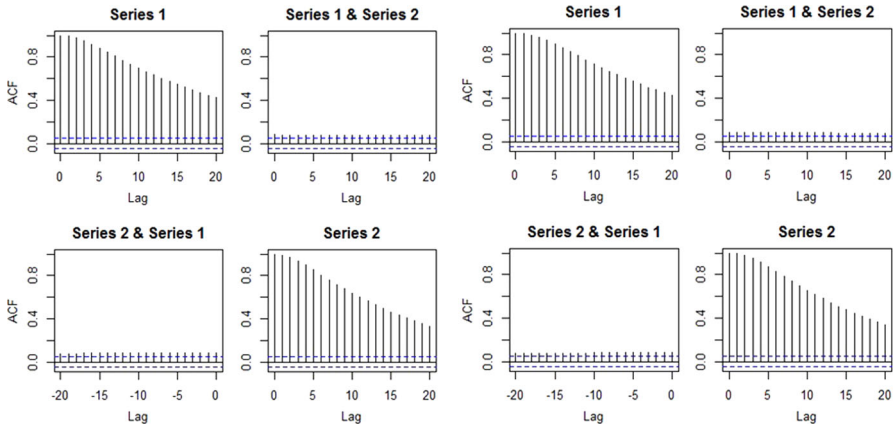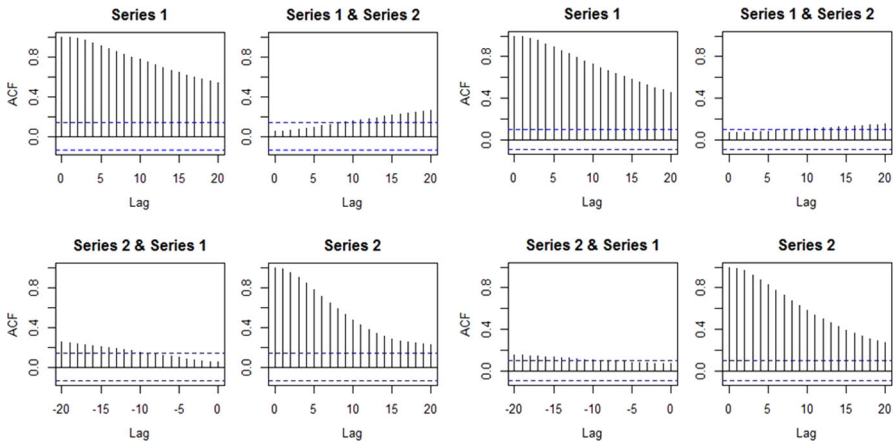
Fig. 13 Plots of the mean of the absolute value of the moving cross-correlations between two MDPCs in Example 2 [Left: MDPCA ($W = 201, l = 1$), Right: MDPCA ($W = 201, l = 5$)]
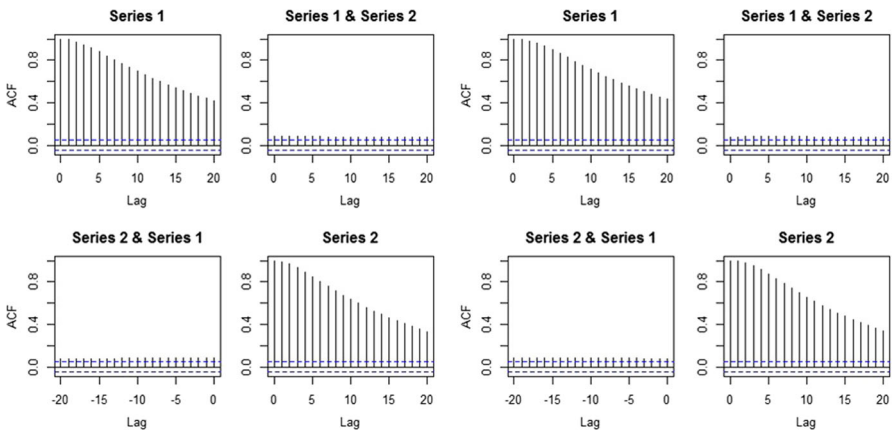
Fig. 14 Plots of the mean of the absolute value of the moving cross-correlations between two MDPCs in Example 2 [Left: MDPCA ($W = 301, l = 1$), Right: MDPCA ($W = 301, l = 5$)]

Fig. 15 Plots of the mean of the absolute value of the moving cross-correlations between two MDPCs with different sample sizes in Example 2 [Left: $T = 200$, Right: $T = 400$]
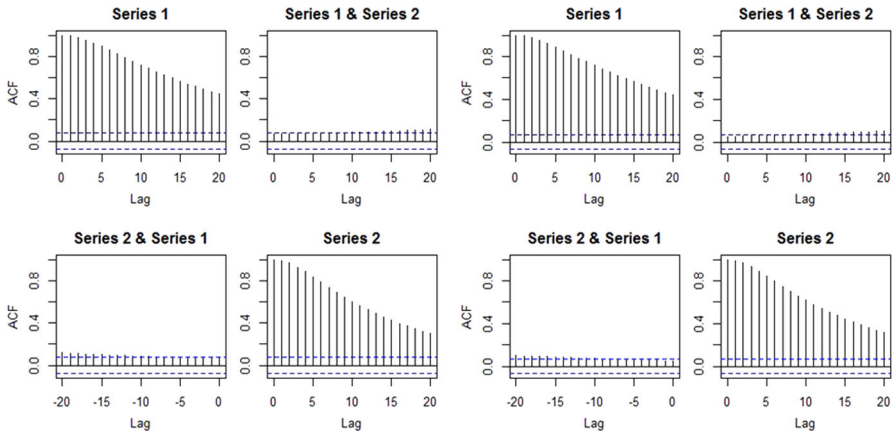

Fig. 16 Plots of the mean of the absolute value of the moving cross-correlations between two MDPCs with different sample sizes in Example 2 [Left: $T = 600$, Right: $T = 800$]
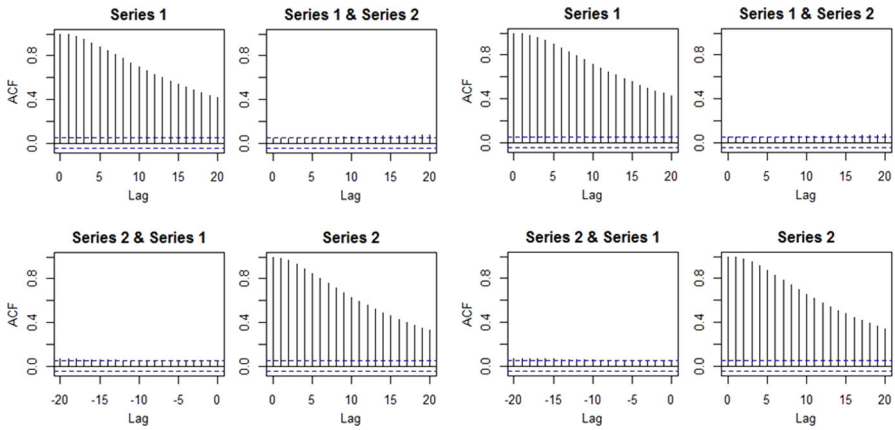
Fig. 17 Plots of the mean of the absolute value of the moving cross-correlations between two MDPCs in Example 3 [Left: MDPCA ($W = 101, l = 1$), Right: MDPCA ($W = 101, l = 5$)]
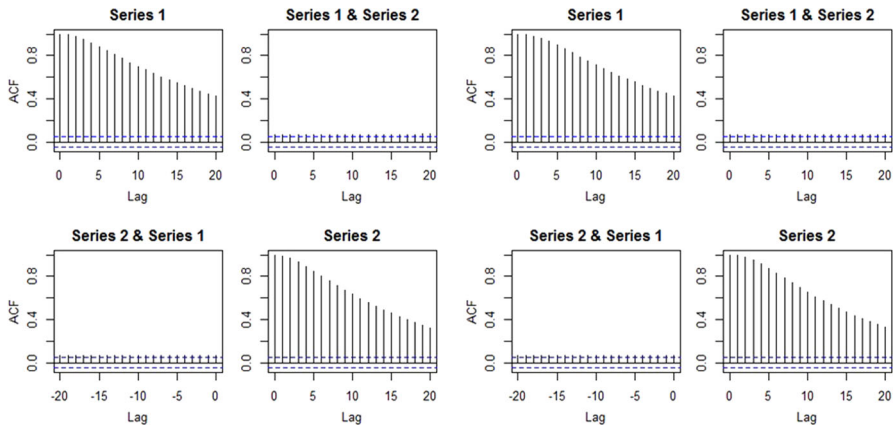
Fig. 18 Plots of the mean of the absolute value of the moving cross-correlations between two MDPCs in Example 3 [Left: MDPCA ($W = 201, l = 1$), Right: MDPCA ($W = 201, l = 5$)]

Fig. 19 Plots of the mean of the absolute value of the moving cross-correlations between two MDPCs in Example 3 [Left: MDPCA ($W = 301, l = 1$), Right: MDPCA ($W = 301, l = 5$)]

Fig. 20 Plots of the mean of the absolute value of the moving cross-correlations between two MDPCs with different sample sizes in Example 3 [Left: $T = 200$, Right: $T = 400$]

Fig. 21 Plots of the mean of the absolute value of the moving cross-correlations between two MDPCs with different sample sizes in Example 3 [Left: $T = 600$, Right: $T = 800$]
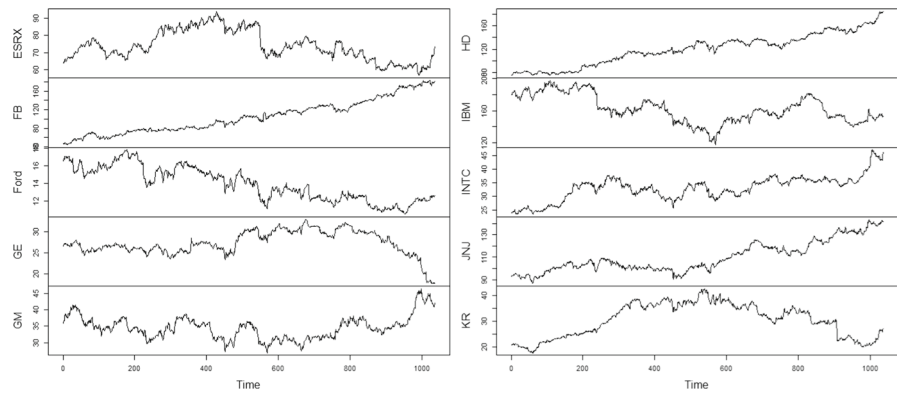
Fig. 22 A time series plot of the daily stock prices of the 17 USA companies in Example 4—part 1
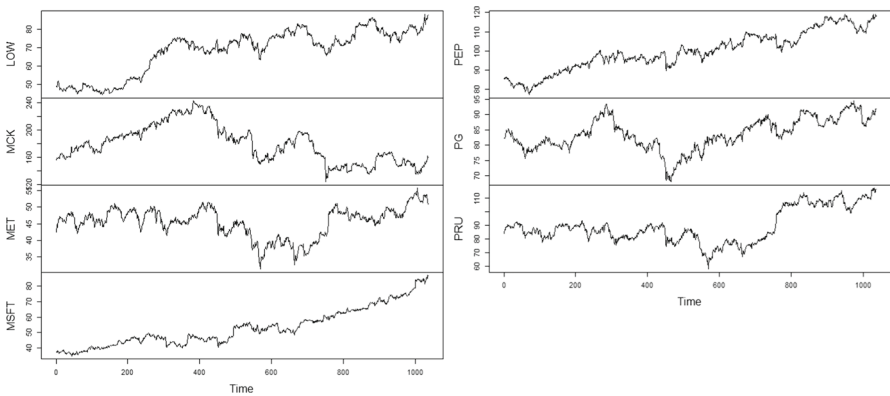
Fig. 23 A time series plot of the daily stock prices of the 17 USA companies in Example 4—part 2
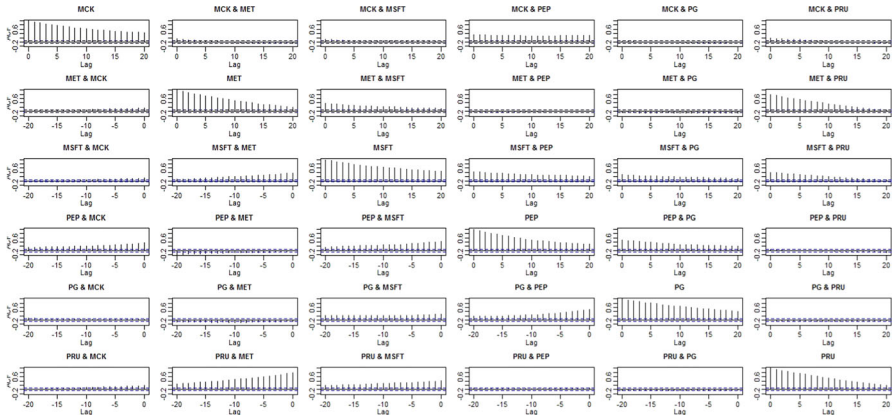
Fig. 24 Last 36 sample moving cross-correlation plots with $W = 101$ of the daily stock prices of the 17 USA companies before applying MDPCA in Example 4
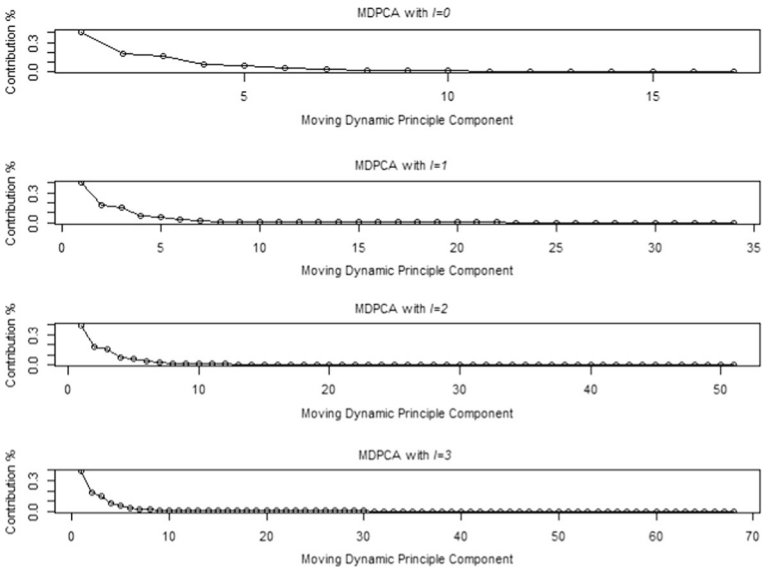
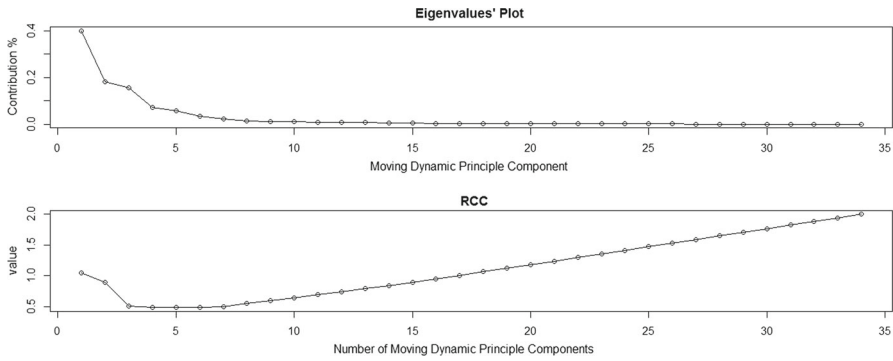**Fig. 25** Eigenvalues' plots after applying MDPCA with $W = 101$ to the 17 USA companies in Example 4

Fig. 26 Top: An eigenvalues' plot of the 17 USA companies in Example 4 after applying MDPCA with $W = 101$ and $l = 1$, Bottom: RCC plot
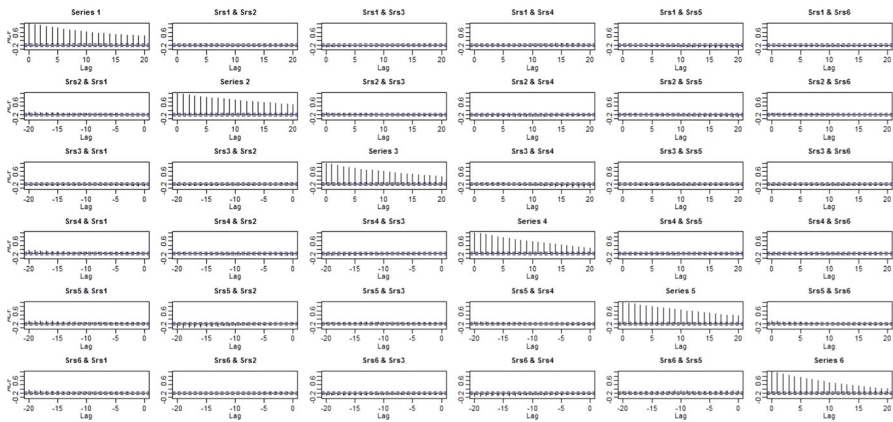

Fig. 27 Sample moving cross-correlation plots of the MDPCs of the 17 USA companies after MDPCA with $W=101$ and $l=1$ is applied in Example 4

References

Alshammri F (2020a) MACF: moving auto- and cross-correlation function. R package version

Alshammri F (2020b) MCOV: moving cross-covariance matrix. R package version

Alshammri F (2020c) MDPCA: moving dynamic principal component analysis for non-stationary multivariate time series. R package version

Alshammri F (2020d) RCCM: retained component criterion for the moving dynamic principal component analysis. R package version

Bai J, Ng S (2002) Determining the number of factors in approximate factor models. Econometrica 70(1):191–221

Box G, Jenkins G (1976) Time series analysis: forecasting and control. Holden-Day San Francisco

Box GE, Tiao GC (1977) A canonical analysis of multiple time series. Biometrika 64(2):355–365

Bradley RC (1986) Basic properties of strong mixing conditions. In: Dependence in probability and statistics. Springer, Berlin, pp 165–192

Brillinger D (1981) Time series: data analysis and theory, vol 36. SIAM, Philadelphia

Broomhead DS, King GP (1986) On the qualitative analysis of experimental dynamical systems. In: Sarkar S (ed) Nonlinear phenomena and chaos. Adam Hilger, Bristol, pp 113–144

Chang J, Guo B, Yao Q (2018) Principal component analysis for second-order stationary vector time series. Ann Stat 46(5):2094–2124

Engle RF, Granger CWJ (1987) Co-integration and error correction: representation, estimation, and testing. Econometrica 55(2):251–276

Forni M, Hallin M, Lippi M, Reichlin L (2005) The generalized dynamic factor model: one-sided estimation and forecasting. J Am Stat Assoc 100(471):830–840

Golub GH, Van Loan CF (1996) Matrix computations, 3rd edn. Johns Hopkins University Press, Baltimore

Golyandina N, Nekrutkin V, Zhigljavsky A (2001) Analysis of time series structure: SSA and related techniques. Chapman & Hall/ CRC, London

He XB, Yang YP (2008) Variable MWPCA for adaptive process monitoring. Ind Eng Chem Res 47(2):419–427

Hosking JR (1980) The multivariate portmanteau statistic. J Am Stat Assoc 75(371):602–608

Hossein H, Rahim M (2013) Multivariate singular spectrum analysis: a general view and new vector force frocasting approach. Int J Energy Stat 1(1):55–83

Johansen S (1995) Likelihood-based inference in cointegrated vector autoregressive models. Oxford University Press, Oxford

Ku W, Storer RH, Georgakis C (1995) Disturbance detection and isolation by dynamic principal component analysis. Chemom Intell Lab Syst 30(1):179–196

Lam C, Yao Q (2012) Factor modeling for high-dimensional time series: inference for the number of factors. Ann Stat 40(2):694–726

Lennox B, Montague G, Hiden H, Kornfeld G, Goulding P (2001) Process monitoring of an industrial fed-batch fermentation. Biotechnol Bioeng 74(2):125–135

Pan J, Yao Q (2008) Modelling multiple time series via common factors. Biometrika 95(2):365–379

Peña D, Box GE (1987) Identifying a simplifying structure in time series. J Am Stat Assoc 82(399):836–843

Peña D, Poncela P (2006) Nonstationary dynamic factor analysis. J Stat Plan Inference 136(4):1237–1257

Peña D, Yohai VJ (2016) Generalized dynamic principal components. J Am Stat Assoc 111(515):1121–1131

Stock JH, Watson MW (1988) Testing for common trends. J Am Stat Assoc 83(404):1097–1107

Stock JH, Watson MW (2002) Forecasting using principal components from a large number of predictors. J Am Stat Assoc 97(460):1167–1179

Tiao GC, Tsay RS (1989) Model specification in multivariate time series. J R Stat Soc Ser B (Methodol) 51(2):157–213