# Extracting a Low-Dimensional Predictable Time Series

Yining Dong S. Joe Qin Stephen Boyd

May 12, 2021

**Abstract**

Large scale multi-dimensional time series can be found in many disciplines, including finance, econometrics, biomedical engineering, and industrial engineering systems. It has long been recognized that the time dependent components of the vector time series often reside in a subspace, leaving its complement independent over time. In this paper we develop a method for projecting the time series onto a low-dimensional time-series that is *predictable*, in the sense that an auto-regressive model achieves low prediction error. Our formulation and method follow ideas from principal component analysis, so we refer to the extracted low-dimensional time series as *principal time series*. In one special case we can compute the optimal projection exactly; in others, we give a heuristic method that seems to work well in practice. The effectiveness of the method is demonstrated on synthesized and real time series.

## 1 Introduction

High dimensional time series analysis and applications have become increasingly important in many different domains. In many cases, the high dimensional time series data are both cross-correlated and auto-correlated. Cross-correlations among different time series make it possible to use a set of lower dimensional time series to represent the original, high dimensional time series. For example, principal component analysis (PCA) ([CK86, BN08]) and generalized PCA ([Cho12]) methods utilize the cross-correlations among different time series to extract lower dimensional factors that capture maximal variance. Although PCA has seen wide use as a dimension reduction method, it does not focus on modeling of auto-correlations or dynamics that can exist in time series data. Since auto-correlations make it possible to predict future values from the past values, it is desirable to perform such low dimensional modeling of the dynamics.

In this work, a linear projection method is proposed to extract a lower dimensional most predictable time series from high-dimensional time series. The entries of the low dimensional time series are mutually uncorrelated so that they capture as much dynamics as possible. The advantage of the proposed method is that it focuses on extracting principal time series with most dynamics. Therefore, the dynamic features of the high dimensional data are

concentrated in a set of lower dimensional time series, which makes it very useful for data prediction, dynamic feature extraction and visualization.

The proposed method has numerous potential applications, ranging from finance to industrial engineering. In finance, if the high dimensional time series consist of returns of some assets, applying the proposed method gives the most predictable portfolio. In chemical processes, oscillations are usually undesirable ([TH97, THZ03]) and applying the proposed method to the process measurements can help detect the unwanted oscillations. In biomedical engineering, electroencephalography (EEG) data can be characterized with waves with different frequencies ([Tep02, Tat14]). Applying the proposed method to EEG data has the potential to detect the different waves.

In this work, the extraction of principal time series and the VAR modeling of the principal time series are achieved simultaneously by solving the optimization problem (*i.e.*, there is no need to refit a VAR model for the principal time series after they are extracted). In addition, the extracted principal time series are best predictable from their past values and capture most of dynamics. This property makes the proposed method very useful for prediction, dynamic feature extraction, and visualization.

# 2 The most predictable projected time series

## 2.1 Predictability of a time series

Consider a wide-sense stationary $n$-dimensional vector time series process $z_t \in \mathbf{R}^n$, $t \in \mathbf{Z}$, with

$$\mathbf{E}\, z_t = 0, \quad \mathbf{E}\, z_t z_{t+\tau}^T = \Sigma_\tau, \quad \tau \in \mathbf{Z}. \tag{1}$$

Here $\Sigma_\tau$ is the auto-covariance matrix for lag $\tau$. The zero mean assumption is without loss of generality, since this can be arranged by subtracting the mean from the original process, if it is not zero. We refer to a time series with $\Sigma_0 = I$ as *standardized*.

**Predictability measure.** An $M$-memory auto-regressive (AR) predictor for $z_t$ has the form

$$\hat{z}_t = A_1 z_{t-1} + A_2 z_{t-2} + \cdots + A_M z_{t-M},$$

where $A_i \in \mathbf{R}^{n\times n}$, $i = 1, 2, \ldots, M$ are the AR (matrix) coefficients. We define the ($M$-memory) (un-)predictability measure for the time series as the smallest possible mean square AR prediction error,

$$\alpha = \min_{A_1,\ldots,A_M} \mathbf{E}\, \|z_t - \hat{z}_t\|_2^2, \tag{2}$$

which has the same value for all $t$. To simplify the notation for the rest of paper, we define $A = \begin{bmatrix} A_1 & \cdots & A_M \end{bmatrix}$.

We can easily evaluate the predictability measure $\alpha$. The objective can be expressed as

$$\begin{aligned} & \mathbf{E}\,\|z_t - \hat{z}_t\|_2^2 \\ = \; & \mathbf{Tr}\left(\Sigma_0 - 2\begin{bmatrix}\Sigma_1\\ \Sigma_2\\ \vdots\\ \Sigma_M\end{bmatrix}^T A^T + A\begin{bmatrix}\Sigma_0 & \Sigma_1^T & \cdots & \Sigma_{M-1}^T\\ \Sigma_1 & \Sigma_0 & \cdots & \Sigma_{M-2}^T\\ \vdots & \vdots & \ddots & \vdots\\ \Sigma_{M-1} & \Sigma_{M-2} & \cdots & \Sigma_0\end{bmatrix} A^T\right). \end{aligned}$$

The optimal AR coefficients are readily found to be

$$A^T = \begin{bmatrix}\Sigma_0 & \Sigma_1^T & \cdots & \Sigma_{M-1}^T\\ \Sigma_1 & \Sigma_0 & \cdots & \Sigma_{M-2}^T\\ \vdots & \vdots & \ddots & \vdots\\ \Sigma_{M-1} & \Sigma_{M-2} & \cdots & \Sigma_0\end{bmatrix}^{-1}\begin{bmatrix}\Sigma_1\\ \Sigma_2\\ \vdots\\ \Sigma_M\end{bmatrix},$$

assuming the inverse of the symmetric semidefinite block Toeplitz matrix above exists. It follows that

$$\alpha = \mathbf{Tr}\left(\Sigma_0 - \begin{bmatrix}\Sigma_1\\ \Sigma_2\\ \vdots\\ \Sigma_M\end{bmatrix}^T\begin{bmatrix}\Sigma_0 & \Sigma_1^T & \cdots & \Sigma_{M-1}^T\\ \Sigma_1 & \Sigma_0 & \cdots & \Sigma_{M-2}^T\\ \vdots & \vdots & \ddots & \vdots\\ \Sigma_{M-1} & \Sigma_{M-2} & \cdots & \Sigma_0\end{bmatrix}^{-1}\begin{bmatrix}\Sigma_1\\ \Sigma_2\\ \vdots\\ \Sigma_M\end{bmatrix}\right). \tag{3}$$

It can be shown that $0 \leq \alpha \leq \mathbf{Tr}\,\Sigma_0$. A low value of $\alpha$ indicates high predictability; a high value of $\alpha$ indicates low predictability. The extreme case $\alpha = 0$ means that the AR model has zero residual, and the extreme case $\alpha = \mathbf{Tr}\,\Sigma_0$ occurs when $z_t$ and $z_s$ are uncorrelated for $t \neq s$, so $\Sigma_\tau = 0$ for $\tau \neq 0$, and the optimal AR coefficients are all zero.

## 2.2 The most predictable projected time series

We can obtain a lower-dimensional time series $x_t \in \mathbf{R}^m$ as a linear function of the original time series $z_t \in \mathbf{R}^n$, as $x_t = W^T z_t$, where $W \in \mathbf{R}^{n\times m}$, with $m < n$. We denote the auto-covariance matrices of $x_t$ as

$$S_\tau = \mathbf{E}\,x_t x_{t+\tau}^T = W^T\Sigma_\tau W, \quad \tau \in \mathbf{Z}.$$

Our goal is to choose $W$ so that the series $x_t$ is predictable, *i.e.*, has a low value of $\alpha$. We evidently need to normalize $W$ to rule out the solution $W = 0$; we do this with the constraint

$$W^T\Sigma_0 W = S_0 = I.$$

This ensures that $\mathbf{E}\,x_t x_t^T = I$, *i.e.*, the low-dimensional time series $x_t$ is standardized.

The most predictable projected time series is found by solving the optimization problem

$$\begin{array}{ll}\text{maximize} & f(W)\\ \text{subject to} & S_0 = I,\end{array} \tag{4}$$

with variables $W$ and $A$, where

$$f(W) = \mathbf{Tr} \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_M \end{bmatrix}^T \begin{bmatrix} S_0 & S_1^T & \cdots & S_{M-1}^T \\ S_1 & S_0 & \cdots & S_{M-2}^T \\ \vdots & \vdots & \ddots & \vdots \\ S_{M-1} & S_{M-2} & \cdots & S_0 \end{bmatrix}^{-1} \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_M \end{bmatrix}.$$

The solution is evidently not unique; if $Q$ is an orthonormal $m \times m$ matrix, then $f(WQ) = f(W)$. In other words, all solutions differ by an orthonormal matrix.

### 2.3 Special case with exact solutions

Here we observe that when $M = 1$ and $m = 1$, the problem can be solved exactly. For projection, the problem is

$$\begin{array}{ll} \text{maximize} & \|W^T \Sigma_1 W\|_F^2 \\ \text{subject to} & W^T \Sigma_0 W = I, \end{array}$$

using $S_0 = I$ to simplify the objective. This is readily solved. Define $Z = \Sigma_0^{1/2} W$ and $Y = \Sigma_0^{-1/2} \Sigma_1 \Sigma_0^{-1/2}$, to the problem is to maximize $\|Z^T Y Z\|_F^2$ subject to $Z^T Z = I$.

Let $V$ denote the eigenvector of $Y + Y^T$ corresponding to the eigenvalue with the maximum magnitude. Then $V$ satisfies the constraint and maximizes the objective value. Therefore, the optimal $W$ is $W^\star = \Sigma_0^{-1/2} V$.

Once an optimal $W^\star$ is obtained, the optimal $A^\star$ can be easily calculated as

$$A^\star = S_1^T S_0^{-1} = S_1^T = (W^\star)^T \Sigma_1^T W^\star.$$

## 3 Algorithm

When $M \neq 1$, there is no exact solution to (4), to the best of our knowledge. Problem (4) is essentially an optimization problem over the Grassmannian manifold. There has been research on how to address such optimization problems, including [AMS09, UM14, EAS98]. In this section, we give a heuristic method that seems to work well in practice.

We will construct the columns of $W \in \mathbf{R}^{n \times m}$ sequentially. Each column is chosen satisfy the constraint $S_0 = I$, while maximizing the predictability of the projected time series. In this section, we explain how to achieve this.

Assume that we have already constructed $k$ columns of $W$, with $W^k \in \mathbf{R}^{n \times k}$ and $A^k \in \mathbf{R}^{k \times Mk}$. (We initialize $k = 0$, $W^k = 0$, and $A^0 = 0$ to construct the first column). Then, our goal is to choose $W^{k+1} = [W^k \ w]$, where $w \in \mathbf{R}^n$, such that the $(k+1)$-dimensional projected time series is most predictable.

This is equivalent to the optimization problem

$$\begin{array}{ll} \text{maximize} & f(W^{k+1}) \\ \text{subject to} & W^{k+1} = [W^k \ w] \\ & S_0^{k+1} = I, \end{array} \tag{5}$$

where $W^k$ is fixed and $S_0^{k+1} = (W^{k+1})^T \Sigma_0 W^{k+1}$. We cannot find exact solutions to this problem. However, we know how to solve for $A^{k+1}$ exactly when $W^{k+1}$ is fixed, and how to solve for $W^{k+1}$ exactly when $A^{k+1}$ is fixed. We can iterate these two steps until convergence to obtain an approximate solution to (5).

### 3.1 Solving for $A^{k+1}$ with fixed $w$

When $w$ is fixed, we can solve for $A^{k+1}$ exactly. According to §2.1, when $W^{k+1}$ is known, the solution of $A^{k+1}$ is

$$A^{k+1,T} = \begin{bmatrix} S_0^{k+1} & S_1^{k+1,T} & \cdots & S_{M-1}^{k+1,T} \\ S_1^{k+1} & S_0^{k+1} & \cdots & S_{M-2}^{k+1,T} \\ \vdots & \vdots & \ddots & \vdots \\ S_{M-1}^{k+1} & S_{M-2}^{k+1} & \cdots & S_0^{k+1} \end{bmatrix}^{-1} \begin{bmatrix} S_1^{k+1} \\ S_2^{k+1} \\ \vdots \\ S_M^{k+1} \end{bmatrix},$$

where $S_\tau^{k+1} = W^{k+1,T} \Sigma_\tau W^{k+1}$, $\tau \in \mathbf{Z}$.

The time complexity for forming $S_1^{k+1}, \ldots, S_M^{k+1}$ is $O(M(k+1)n^2)$. Once $S_\tau^{k+1}$, $\tau = 1, \ldots, M$ are calculated, the time complexity for updating $A^{k+1}$ is dominated by the inversion step, which is $O(M^3(k+1)^3)$. Therefore, the overall time complexity for updating $A^{k+1}$ is $\max\{O(M(k+1)n^2), O(M^3(k+1)^3)\}$.

### 3.2 Solving for $w$ with fixed $A^{k+1}$

When $A^{k+1}$ is fixed, we can solve for $W^{k+1}$ exactly. With some derivations, the optimization problem for $w$ can be expressed as

$$\begin{array}{ll} \text{minimize} & f(w) = w^T B w - 2c^T w \\ \text{subject to} & W^{k,T} \Sigma_0 w = 0 \\ & w^T \Sigma_0 w = 1, \end{array} \tag{6}$$

where $B \succ 0$, and $B$, $c$ are known. The derivation and expressions for $B$ and $c$ can be found in Appendix A. The solution of this problem can be obtained explicitly as follows.

Let $U \in \mathbf{R}^{n \times (n-k)}$ be the orthogonal complement of $\Sigma_0^{1/2} W^k$ and $U^T U = I$. Denote the SVD decomposition of $U^T \Sigma_0^{-1/2} B \Sigma_0^{-1/2} U$ as $U^T \Sigma_0^{-1/2} B \Sigma_0^{-1/2} U = V \Lambda V^T$ with $\Lambda = \mathbf{diag}(\lambda_1, \ldots, \lambda_{n-k})$, $(\lambda_1 \leq \lambda_2 \cdots \leq \lambda_{n-k})$. Then, problem (6) can be transformed as

$$\begin{array}{ll} \text{minimize} & f(z) = y^T \Lambda y - 2\beta^T y \\ \text{subject to} & y^T y = 1, \end{array} \tag{7}$$

where $\beta = [\beta_1 \ \beta_2 \ \cdots \ \beta_{n-k}] = V^T U^T \Sigma_0^{-1/2} c$ and $w = \Sigma_0^{-1/2} U V y$. The solution to problem (7) has the form $y = (\Lambda + \mu I)^{-1} \beta$ where $\mu$ can be obtained as the root of

$$\sum_{i=1}^{n-k} \frac{\beta_i^2}{(\lambda_i + \mu)^2} = 1$$

**Algorithm 1** Complete algorithm for approximate solution to (4).

1: Set initial values of $W^0$ as a vector of zeros of proper size, and $A^0 = 0$.
2: **for** $k = 1, 2, \ldots$ **do**
3:  **repeat**
4:  Set $w = 0$
5:  Solve $A^{k+1}$ with fixed $W^{k+1} = [W^k \ w]$ according to §3.1
6:  Solve $W^{k+1} = [W^k \ w]$ with fixed $A^{k+1}$, according to §3.2
7:  **until** termination condition is satisfied
8: **end for**

that satisfies $\mu > -\lambda_1$.

After $\mu$ is found, the optimal $w$ of problem (6) can be obtained as

$$w^\star = \Sigma_0^{-1/2} U V (\Lambda + \mu I)^{-1} \beta.$$

According to the expressions of $B$ and $c$ in Appendix A, the time complexity for forming $B$ and $c$ is $O(M^2 n^2)$. Once $B$ and $c$ are calculated, the time complexity for updating $w$ is dominated by the SVD step, which is $O(n^3)$. Therefore, the overall time complexity for updating $w$ is $\max\{O(M^2 n^2), O(n^3)\}$.

## 3.3 Complete algorithm

We have discussed algorithms to solve for $A^{k+1}$ when $W^{k+1}$ is fixed and to solve for $W^{k+1} = [W^k \ w]$ when $A^{k+1}$ is fixed. The complete procedure to construct the $(k+1)th$ column of $W$ is to iterate these two steps until convergence. As discussed in §3.1 and §3.2, the time complexity for updating $A^{k+1}$ is $\max\big\{O(M(k+1)n^2), O(M^3(k+1)^3)\big\}$, and the time complexity for updating $w$ is $\max\{O(M^2 n^2), O(n^3)\}$. In practice, it is often the case that $m \ll n$ and $M \ll n$. Therefore, the overall time complexity at each iteration step is $O(n^3)$.

Once $W^{k+1}, A^{k+1}$ are obtained, the same procedure can be applied to construct the next column of $W$. The complete algorithm is given in Algorithm 1.

The heuristic method proposed has two advantages over directly solving the original problem (4). First, there is no uniqueness issue in this recursive method, because starting from the first column of $W$, each column of $W$ is deterministic according to the optimization problem (6). Second, the iterative procedure gives an indication of the dimension of the predictable vector time series.

Let $\alpha_k$ denote the predictability measure for the $k$ dimensional projected time series, or the value of the objective function in (4) with optimal $W^k$ and $A^k$. Then, when extracting the $(k+1)$th scalar time series, we have the following upper bound for $\alpha_{k+1}$ of the $(k+1)$-dimensional time series,

$$\alpha_{k+1} \leq \alpha_k + w^T \Sigma_0 w = \alpha_k + 1.$$

When the optimal $\alpha_{k+1}$ is very close to the upper bound $\alpha_k + 1$, we can draw two conclusions. First, adding another scalar time series does not improve the prediction of the $k$ dimensional

vector time series. Second, no other self-predictable scalar time series can be extracted. These two facts suggest that all the predictable components in the time series data are already extracted and hence, we can stop the iteration procedure.

# 4 Examples

In this section, we test our method by applying it to a synthesized high-dimensional time series dataset and a quarterly GDP growth dataset. In both examples, we demonstrate advantages of our proposed method over scalar time series AR fitting.

## 4.1 Simulation dataset

The proposed method is first tested on a synthesized dataset generated from the model

$$\begin{aligned} x_t &= B_1 x_{t-1} + B_2 x_{t-2} + v_t, \\ y_t &= P x_t + e_t, \\ z_t &= Q \Sigma_y^{-1/2} y_t, \end{aligned} \tag{8}$$

where

$$B_1 = \begin{bmatrix} 1.1241 & 0.3045 & 0.3806 \\ 0.3902 & -0.8169 & -0.3114 \\ -0.7166 & -0.8630 & 1.0115 \end{bmatrix}, \quad B_2 = \begin{bmatrix} -0.2482 & 0.3676 & 0.0328 \\ -0.4240 & 0.1101 & 0.0267 \\ 0.6011 & -0.5975 & -0.3224 \end{bmatrix}.$$

The matrix $P \in \mathbf{R}^{1000\times 3}$ is random matrix with orthonormal columns, $Q \in \mathbf{R}^{1000\times 1000}$ is a random orthonormal matrix, $v_t$ is i.i.d. $\mathcal{N}(0, I)$ and $e_t$ is i.i.d $\mathcal{N}(0, 0.02^2 I)$. $\Sigma_y$ is the empirical covariance matrix of $y_t$, calculated as

$$\Sigma_y = \frac{1}{T} \sum_{t=1}^{T} y_t y_t^T,$$

where $T$ in the number of samples. In this example, 10,000 consecutive samples are generated from the model. Hence, $T = 10,000$. The sample autocorrelation functions of the true underlying predictable time series $x_t$ are plotted in Fig. 1. We can see that there are strong temporal dependence in $x_t$.

**Scalar AR approach.** We first fit a 2-memory AR model to each scalar time series in $z_t$ to check the predictability of each scalar time series. We use mean squared error (MSE) to evaluate the prediction performance of the fitted AR predictors. We find that, of all 1000 scalar time series, the minimal MSE is 0.9984 and the maximum MSE is 1.0000. Since each scalar time series has approximately mean 0 and variance 1, this indicates that 2-memory scalar AR fitting fails to extract any significant predictability information from the data. The autocorrelation functions of the first 3 scalar time series in $z_t$ are plotted in Fig. 2. It is quite clear that there are no significant temporal dependence that can be observed.

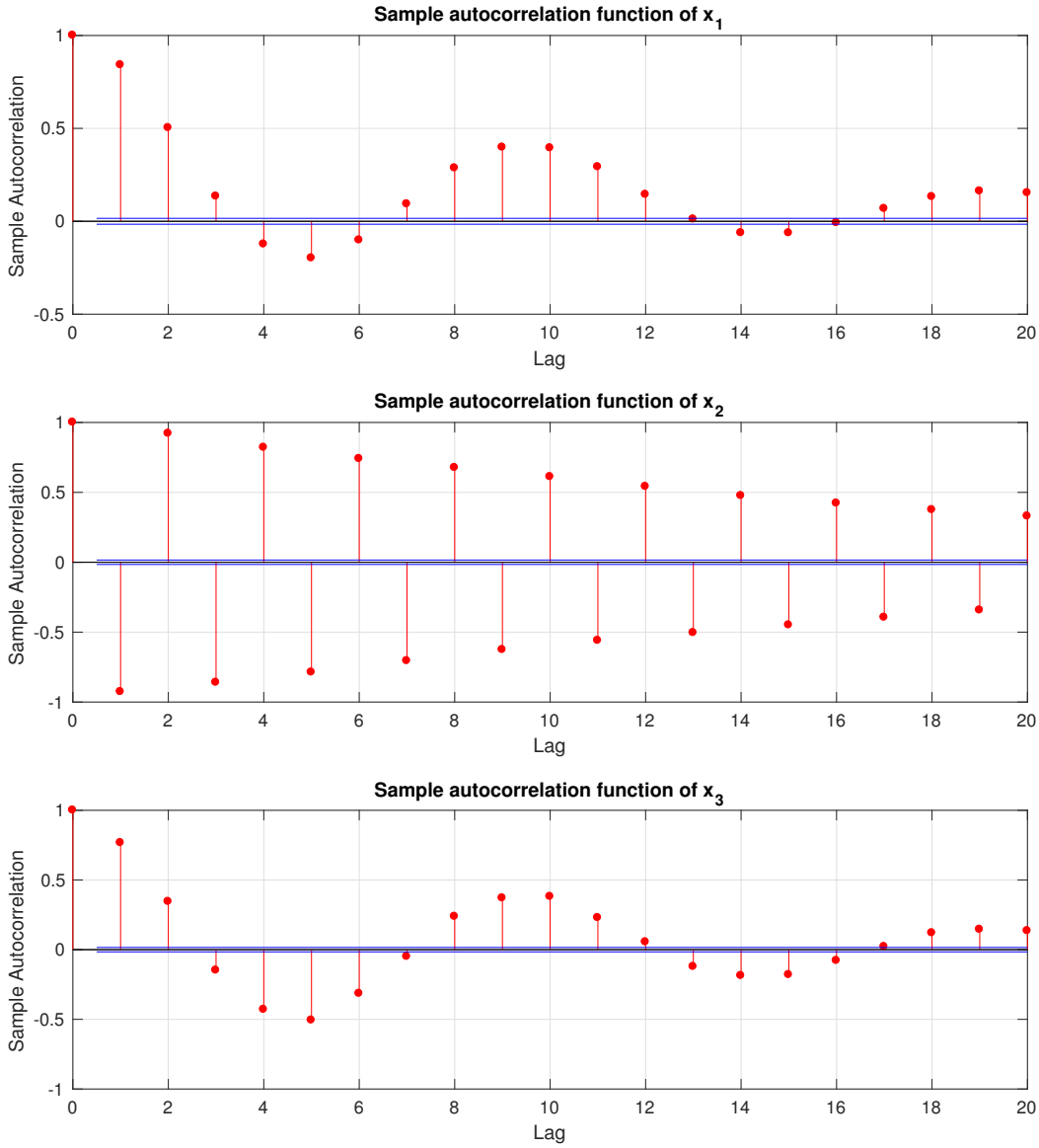


**Figure 1:** Sample autocorrelation functions of the true underlying predictable time series

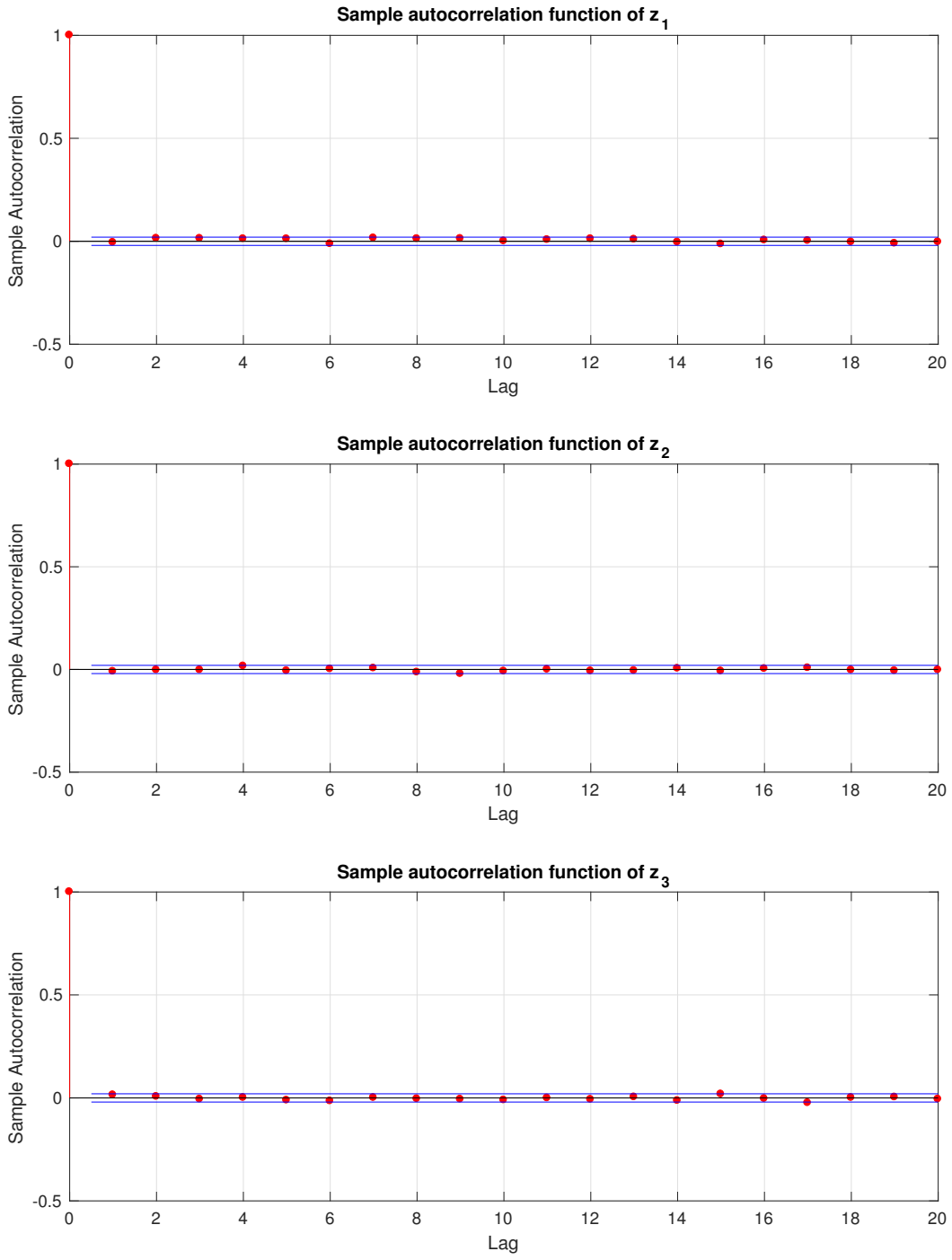


**Figure 2:** Sample autocorrelation functions of the first 3 scalar time series in $z_t$

**Table 1:** MSE of the extracted time series using different approaches

| | Proposed method | Canonical transformation | DiCCA |
|---|---|---|---|
| 1st extracted time series | 0.0568 | 0.0876 | 0.0797 |
| 2nd extracted time series | 0.0824 | 0.0787 | 0.1092 |
| 3rd extracted time series | 0.0679 | 0.0974 | 0.1527 |

**PCA approach.** PCA has been widely utilized in literature to extract latent factors from vector time series. In this example, it is clear from the model assumption that the empirical covariance of $z_t$ is approximately an identity matrix. Therefore, PCA approach would extract the 3-dimensional latent time series as the first 3 scalar time series in $z_t$. As shown in Fig. 2, we can tell that this approach is not able to successfully identify the most predictable factors.

**Proposed approach.** We apply the proposed method to this simulation dataset with 2-memory and $m = 3$. The auto-covariance matrices of $z_t$ are estimated as

$$\Sigma_\tau = \frac{1}{T-M} \sum_{t=1}^{T-M} z_t z_{t+\tau}^T, \quad \tau = 1, 2.$$

and

$$\Sigma_{-\tau} = \Sigma_\tau^T, \quad \tau = 1, 2.$$

Using the extracted VAR predictor, we find that the MSE of the extracted 3-dimensional principal time series are given in Table 1. Since each of the principal time series have approximately mean 0 and variance 1, the low MSEs indicate that the extracted VAR predictor is able to make highly accurate predictions. This is a significant improvement over scalar AR predictors. The autocorrelation functions of the 3-dimensional principal time series extracted using the proposed method are plotted in Fig. 3. It can be seen that there are indeed strong temporal dependence in the extracted predictable time series.

In addition to predictability, we also want to check the similarity between the extracted VAR model and the true model. According to §2.2 and the relationships in (8), the recovered VAR model parameters $A_1$ and $A_2$ can be equivalent to the true model parameters $B_1$, $B_2$ up to a similarity transformation. Therefore, we check how similar the two models are by comparing the eigenvalues of $\begin{bmatrix} A_1 & A_2 \\ I & 0 \end{bmatrix}$ and $\begin{bmatrix} B_1 & B_2 \\ I & 0 \end{bmatrix}$. The closer the two sets of eigenvalues are, the more similar the two models are. Table 2 lists the two sets of eigenvalues ordered in terms of magnitude. We can see that the two sets of eigenvalues are quite close to each other, which implies that the recovered VAR model is close to the true model. The high predictability of the extracted principal time series and the successfully recovered VAR model demonstrate the effectiveness of the proposed method.

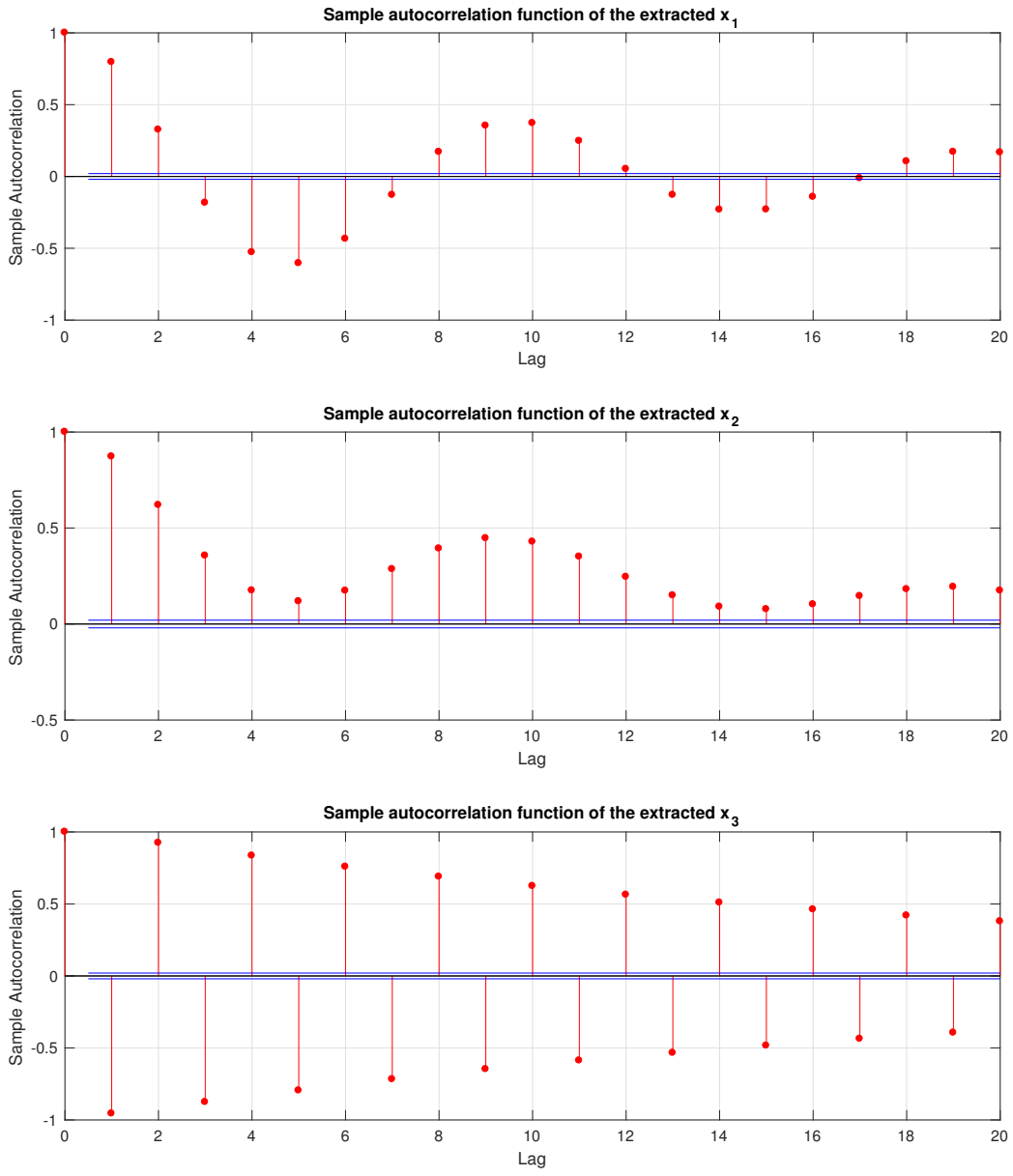


**Figure 3:** Sample autocorrelation functions of the 3-dimensional extracted predictable time series using proposed method

**Table 2:** Eigenvalues of the VAR models extracted using different approaches.

| True model | Proposed method | Canonical transformation | DiCCA |
|---|---|---|---|
| -0.9560 | -0.9522 | -0.9535 | -0.9641 |
| 0.9230 | 0.9302 | 0.9104 | 0.7303 + 0.5533i |
| 0.7117 + 0.5542i | 0.7164 + 0.5574i | 0.7072 + 0.5470i | 0.7303 - 0.5533i |
| 0.7117 - 0.5542i | 0.7164 - 0.5574i | 0.7072 - 0.5470i | 0.6968 + 0.3303i |
| -0.2544 | -0.2702 | -0.2069 | 0.6968 - 0.3303i |
| 0.1827 | 0.1943 | 0.1113 | 0.0624 |

**DiCCA approach.** According to the method described in [DQ18a], a dynamic-inner canonical correlation analysis (DiCCA) model with order 2 is built on the same dataset. A key difference between DiCCA and the proposed method is that DiCCA does not consider interactions among different entries in $x_t$. The MSE of the extracted 3-dimensional time series are listed in Table 1, and eigenvalues of the recovered VAR model are given in Table 2. Since DiCCA does not consider the predictabilities among different entires in $x_t$, it has higher MSE values and large differences from the true eigenvalues.

**Canonical transformation approach.** According to the method described in [BT77], a full-dimensional VAR model with 2-memory is fit to $z_t$ first, and the covariance matrix $\Sigma(\hat{z})$ of the prediction $\hat{z}_t$ is calculated. The linear transformation matrix $W_1$ is selected as the eigenvectors corresponding to the largest magnitude eigenvalues of matrix $\Sigma_0^{-1}\Sigma(\hat{z})$. The MSE values of the extracted 3-dimensional latent time series are listed in Table 1 as well.

In fact, many methods that extract lower-dimensional predictable time series involve fitting a high-dimensional VAR model (which can be very time consuming when $n$ and $M$ are large), and do not consider the prediction models for the lower-dimensional predictable time series in the objective functions. An AR model is often fit subsequently after extracting the lower-dimensional predictable time series for prediction purposes. In order to examine how closely this method recovers the underlying model structure, a VAR model with 2-memory is fit to $W_1^T z_t$, and the eigenvalues are listed in Table 2. Compare to the proposed method, it recovers the small magnitude eigenvalues less accurately than the proposed method.

**PFA approach and reduced rank AR approach.** The predictable factor analysis (PFA) method developed in [RW15] without nonlinear expansion and regularization was tested, as well as the reduced rank AR approach discussed in [VRW86]. In fact, we can show that without special treatment, these two approaches are equivalent to the canonical transformation method in [BT77], and the results are the same as the results of the canonical transformation method.

**Table 3:** (Un-)predictabilities of Mexico and Belgium using single variable AR fitting, (un-)predictabilities of principal time series extracted by the proposed method and (un-)predictabilities of naïve zero predictor.

| | (Un-)predictability | (Un-)predictability of naïve zero predictor |
|---|---|---|
| Mexico, $m = 1$ | 0.8289 | 1 |
| Belgium, $m = 1$ | 0.7939 | 1 |
| $M = 1$, $m = 1$ | 0.4417 | 1 |
| $M = 2$, $m = 1$ | 1.0484 | 2 |

## 4.2 GDP dataset

This dataset is composed of seasonally adjusted quarterly GDP growth data of 17 countries from 1961-Q2 to 2017-Q3. The 17 countries are selected based on the largest GDP countries according to the world bank data in 2016 with complete GDP growth records from 1961-Q2 to 2017-Q3, and this data is downloaded from `https://stats.oecd.org/index.aspx?queryid=350#`. The 17 countries are United States, Japan, United Kingdom, France, Italy, Canada, South Korea, Australia, Spain, Mexico, Netherlands, Switzerland, Germany, Sweden, Belgium, Austria, and Norway.

Two approaches are applied to the first 135 samples from 1961-Q2 to 1994-Q4. The first approach is to fit a single variable AR model to each country's GDP data. The second approach is to use the proposed method in this work to extract the most predictable principal time series. For each approach, the 135 samples are preprocessed such that each variable has zero mean and unit variance, and an AR model with 1-memory is fitted to the preprocessed data.

Table 3 summarizes the (un-)predictabilities of fitting single variable AR model to two representative countries, Mexico and Belgium, the (un-) predictabilities of the one- and two-dimensional most predictable time series extracted by the proposed method and the (un-)predictabilities of naïve zero predictor. The naïve zero predictor is defined to always predict zero. In this case, since all the variables are preprocessed to have zero mean, the naïve zero predictor is the essentially the same as the mean predictor, and the corresponding (un-)predictabilities can serve as a reference to evaluate the performance of other predictors. The reason we pick Mexico and Belgium is that they are the two countries with the best predictabilities (lowest (un-)predictability values) using single variable AR predictor.

We can see from Table 3 that both approaches result in lower (un-) predictabilities than the naïve zero predictor, indicating that both approaches give more efficient predictors that naïve zero predictor. In addition, the improvements in the (un-)predictabilities of principal time series extracted by the proposed method are much more obvious than the improvements in the (un-) predictabilities of single variable AR predictor. This implies that the principal time series extracted by the proposed method are significantly more predictable than any individual scalar time series.

Figure 4 shows the prediction results of the two approaches. We can see that the predictions of the principal time series extracted by our method are closer to their true values, compared to the predictions of Mexico and Belgium's scaled GDP using single variable AR predictor. This again demonstrates the effectiveness of the proposed method.

The predictor obtained for the 1 factor ($m = 1$) case using our proposed method is

$$\hat{x}_t = 0.7382 x_{t-1}.$$

The contribution or weight of each country is shown in Figure 5. 5 countries have relatively large contributions, which are Japan, Italy, Spain, Germany and Belgium.

The predictor obtained for the 2 factors ($m = 2$) case is

$$\hat{x}_t = \begin{bmatrix} 0.7382 & -0.0149 \\ 0.0149 & -0.6208 \end{bmatrix} x_{t-1}.$$

The contribution or weight of each country is shown in Figure 6. 6 countries have relatively large contributions, which are Japan, France, Korea, Australia, Mexico and Austria.

## 4.3 GDP dataset prediction

Next, we examine the prediction performance of the predictors found in §4.2 on unseen data. The data we use in this section are composed of data from 1994-Q4 to 2017-Q3 (samples from 135 to 226 in the same dataset as in §4.2). We understand that the mean and variance of variables in this dataset are considerably different from the mean and variance in the dataset analyzed in §4.2, especially due to the financial crisis around 2008. However, we would still like to check whether the predictors found in §4.2 persist.

First, we shift and scale each scalar time series with its corresponding mean and standard deviation found in §4.2. Then, we apply the predictors found in §4.2 to this preprocessed dataset to make predictions. It is worth noting that this is a very challenging task as it covers the financial crisis period from 2007 to 2009.

We use MSE as a measure of how good the predictions are. Table 4 summarizes the MSE when applying single variable AR predictor to Mexico and Belgium, the MSE when applying the predictor to the principal time series obtained using our proposed method and the MSE of naïve zero predictor. The MSE of naïve zero predictor can be used as a reference to evaluate the performance of the other two predictors.

We can see from Table 4 that both single variable AR predictors and our proposed predictors perform better than naïve zero predictor. However, our proposed method provides dramatic improvements over the naïve zero predictor compared to the single variable AR predictors.

Figure 7 shows the prediction results. We can see that the predictions made using the proposed method are more accurate than the predictions using single variable AR predictor. In addition, the predictor found by our proposed method generalizes well even during the financial crisis period. The first factor captures the big drop around 2008.

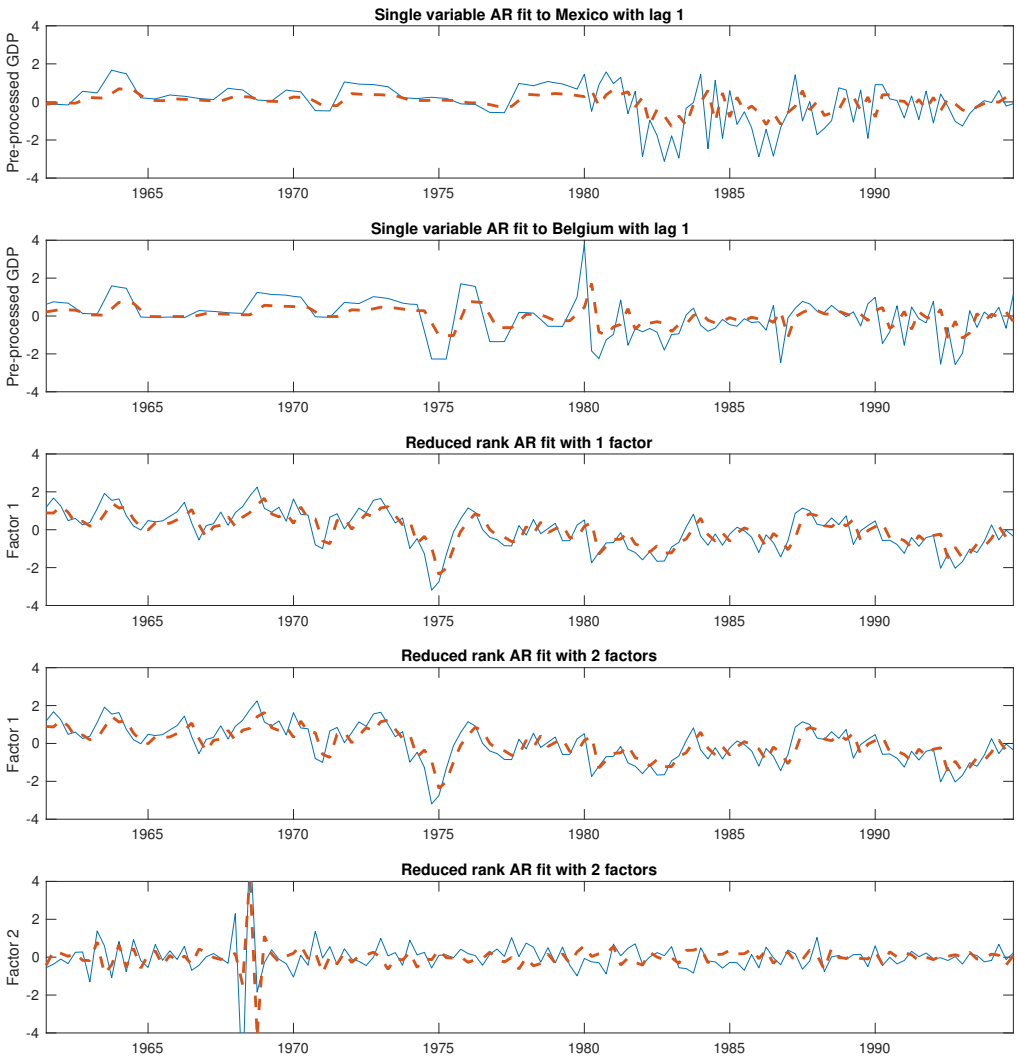


**Figure 4:** Prediction results using two approaches from 1961-Q3 to 1994-Q4. The solid blue line represents the true values and the dashed red line represents the predicted values.

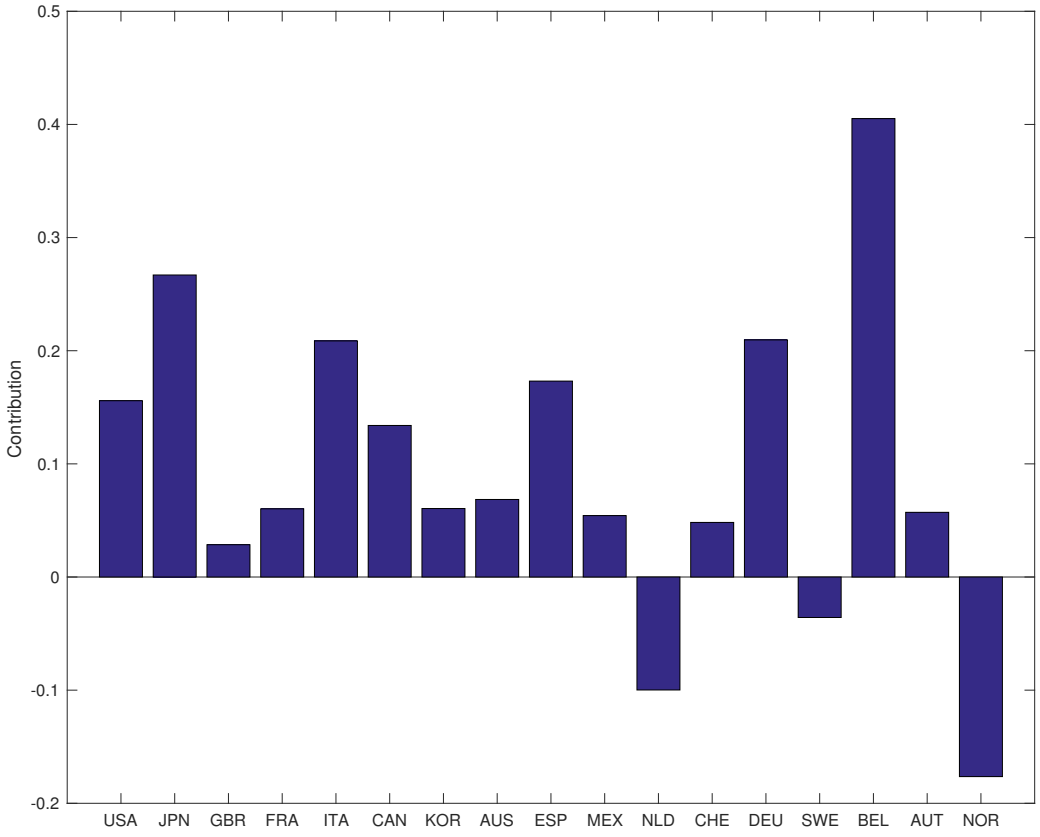


**Figure 5:** Contribution of each country to the most predictable factor when memory is 1 and the number of factors is 1.

**Table 4:** MSE of Mexico and Belgium using single variable AR fitting, MSE of principal time series extracted by the proposed method and MSE of naïve zero predictor.

| | MSE | MSE of naïve zero predictor |
|---|---|---|
| Mexico, m=1 | 1.4072 | 1.8671 |
| Belgium, m=1 | 0.3844 | 0.6536 |
| M = 1, m=1 | 0.5009 | 1.3902 |
| M = 2, m = 1 | 0.6961 | 1.5015 |

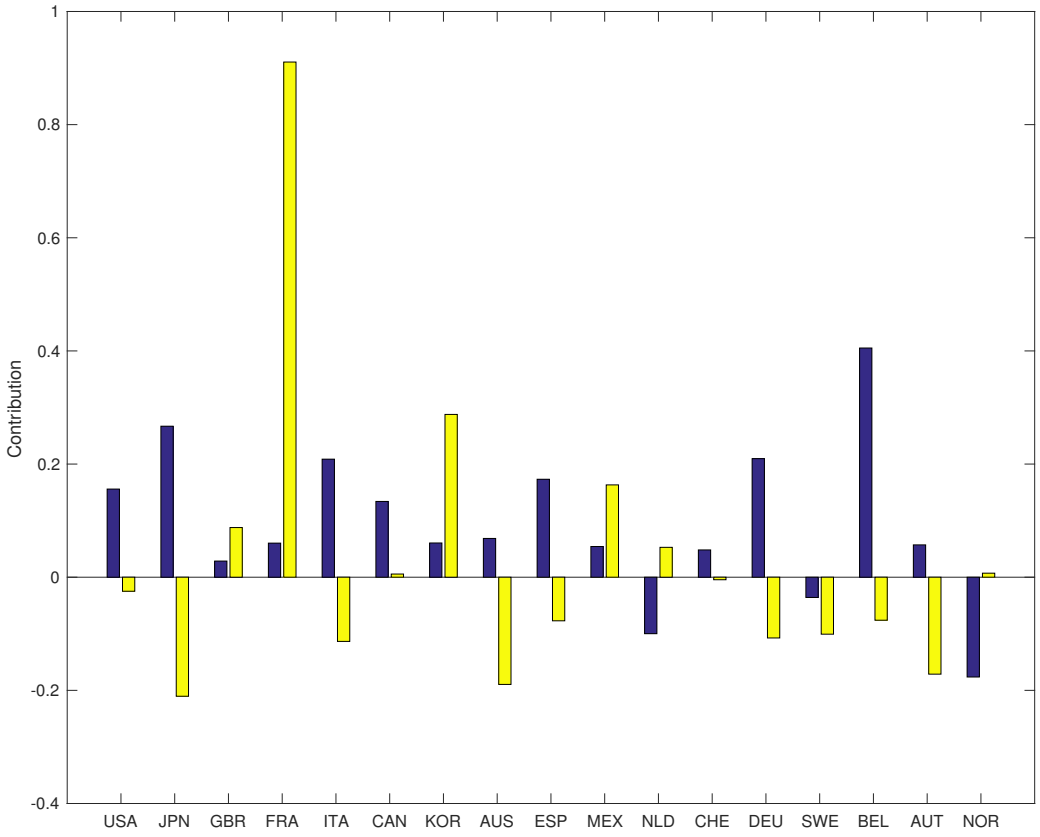


**Figure 6:** Contribution of each country to the most predictable factor when memory is 1 and the number of factors is 2.

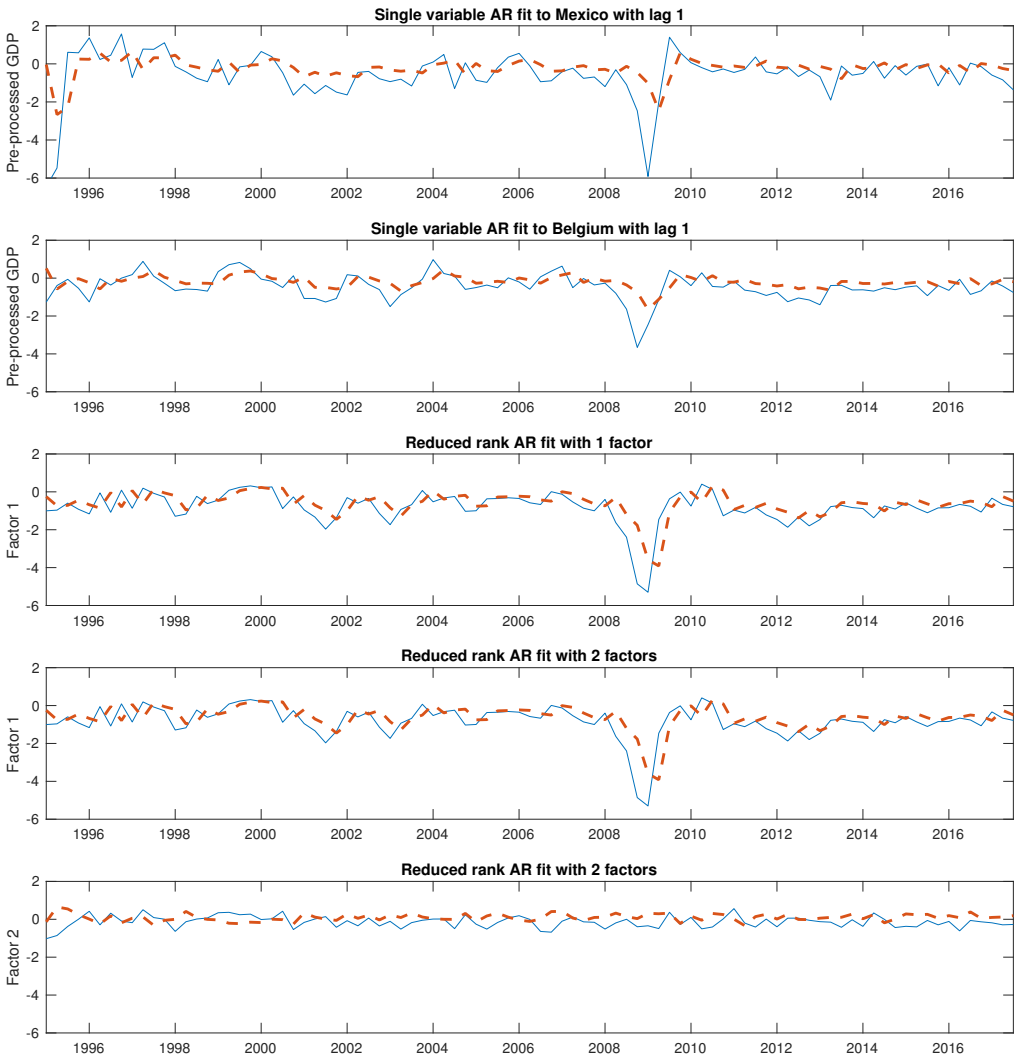


**Figure 7:** Prediction results using two approaches from 1995-Q1 to 2017-Q3. The solid blue line represents the true values and the dashed red line represents the predicted values.

# 5 Extensions and variations

There are several extensions of and variations on the problems that we describe in this paper.

**Regularization.** We can add regularization on $W$ and $A$ to the objective. We denote the regularized problem as

$$\begin{array}{ll} \text{maximize} & f(W) + r(A) + \tilde{r}(W) \\ \text{subject to} & S_0 = I, \end{array} \tag{9}$$

where $r : \mathbf{R}^{mM \times m} \to \mathbf{R}$ and $\tilde{r} : \mathbf{R}^{n \times m} \to \mathbf{R}$. $r$ and $\tilde{r}$ can be chosen to enforce certain properties in $A$ and $W$. For example, regularization using $r(A) = \|A\|_F$ can be added to avoid overfitting of the AR model; $r(A) = \|A\|_1$ can be added to encourage a parsimonious structure of the AR model; $\tilde{r}(W) = \|W_i\|_1$ enforces $i$th column of $W$ to be sparse, such that the $i$th most predictable time series only depends on a few entries in the high dimensional time series.

**Low rank structure.** In many cases, high dimensional time series have low rank structure. We can use this low rank structure to reduce the computational complexity of our problem. When the high dimensional time series has low rank structure, the covariance matrix $\Sigma_0$ has many eigenvalues close 0. Let $\tilde{U}$ denote the collections of all the eigenvectors of $\Sigma_0$ corresponding to the significantly non-zero eigenvalues, then original inputs $\Sigma_\tau$, $\tau = 0, 1, \ldots, M$ can be approximately transformed into $\Phi_\tau$, $\tau = 0, 1, \ldots, M$, where $\Phi_\tau = \tilde{U}^T \Sigma_\tau \tilde{U}$. Since the dimension of $\Phi_\tau$ is much lower than the dimension of $\Sigma_\tau$, by working with the new input series $\Phi_\tau$, the computational complexity reduces significantly. Let $W_\phi$ and $A_\phi$ denote the solutions with the new inputs $\Phi_\tau$, then the solutions with the original inputs $\Sigma_\tau$ can be obtained as

$$W = \tilde{U} W_\phi, \quad A = A_\phi.$$

**Filtering.** As an extension of the projection method, we can consider extracting a lower-dimensional time series $x_t$ using a finite impulse response (FIR) filter with length $L$:

$$x_t = W_1^T z_t + W_2^T z_{t-1} + \cdots + W_L^T z_{t-L+1}, \quad t \in \mathbf{Z},$$

where $W_1, \ldots, W_L \in \mathbf{R}^{n \times m}$ are the filter coefficients. For $L = 1$, this reduces to the projection problem (4). We can write this as

$$x_t = W^T \begin{bmatrix} z_t \\ z_{t-1} \\ \vdots \\ z_{t-L+1} \end{bmatrix}, \quad t \in \mathbf{Z},$$

where

$$W = \begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_L \end{bmatrix}.$$

Define the following auto-covariance matrices of $\begin{bmatrix} z_t \\ z_{t-1} \\ \vdots \\ z_{t-L+1} \end{bmatrix}$:

$$\Omega_0 = \mathbf{E} \begin{bmatrix} z_t \\ z_{t-1} \\ \vdots \\ z_{t-L+1} \end{bmatrix} \begin{bmatrix} z_t \\ z_{t-1} \\ \vdots \\ z_{t-L+1} \end{bmatrix}^T = \begin{bmatrix} \Sigma_0 & \Sigma_1^T & \cdots & \Sigma_{L-1}^T \\ \Sigma_1 & \Sigma_0 & \cdots & \Sigma_{L-2}^T \\ \vdots & \vdots & \ddots & \vdots \\ \Sigma_{L-1} & \Sigma_{L-2} & \cdots & \Sigma_0 \end{bmatrix},$$

$$\Omega_1 = \mathbf{E} \begin{bmatrix} z_t \\ z_{t-1} \\ \vdots \\ z_{t-L+1} \end{bmatrix} \begin{bmatrix} z_{t+1} \\ z_t \\ \vdots \\ z_{t-L+2} \end{bmatrix}^T = \begin{bmatrix} \Sigma_1 & \Sigma_0 & \cdots & \Sigma_{L-2}^T \\ \Sigma_2 & \Sigma_1 & \cdots & \Sigma_{L-3}^T \\ \vdots & \vdots & \ddots & \vdots \\ \Sigma_L & \Sigma_{L-1} & \cdots & \Sigma_1 \end{bmatrix}.$$

The goal is to choose $W$ so that the series $x_t$ is most predictable by an $M$-memory AR predictor. Similar to the projection problem, we add the constraint $S_0 = W^T \Omega_0 W = I$ to rule out the trivial $W = 0$ case. This also ensures that the low-dimensional time series $x_t$ is standardized.

# 6 Related work

The general problem of extracting low-dimensional latent variables from high-dimensional time series has been studied for decades in many different research fields from control systems, signal processing, to economics. Many methods have been developed, and here we survey a subset of representative methods that are closely related to the proposed method.

## 6.1 Extracting predictable latent variables

The extraction of predictable latent variables can be traced back to the work [BT77], where canonical transformation is proposed to extract the lower dimensional components ordered from least to most predictable. Mathematically, the transformation matrix $W$ in [BT77] is selected as the eigenvectors corresponding to the $m$ largest eigenvalues in $\Sigma_0^{-1}\Sigma(\hat{z})$, where $\Sigma(\hat{z}) = \mathbf{E}\, \hat{z}_t \hat{z}_t^T$, and $\hat{z}_t$ is the one-step ahead prediction of $z_t$ found in (2). It is clear that the method in [BT77] is different from the proposed method unless $M = 1$ and $m = 1$.

Later in [PB87], the $z_t$ vector is decomposed into one component that contains factors mixed up with noise, where the transformation matrix $W$ is obtained by analyzing the eigenstructure of the VAR model coefficients of $z_t$, and one component contains white noise.

The slow feature analysis (SFA) method proposed in [WS02] aims to extract a lower-dimensional "slowly varying" time series. Without nonlinear expansion, SFA can be treated as a special case of the proposed method if we restrict $M = 1$ and $A_1 = I$. Even though "slowly varying" time series are predictable, predictable time series do not necessarily have slow variations. To deal with this, the later work [RW15] proposed a more general predictable feature analysis (PFA) approach to extract lower-dimensional predictable time series. In PFA method, the selection of the transformation matrix $W$ involves an eigen-decomposition of $\mathbf{E}(z_t - \hat{z}_t)(z_t - \hat{z}_t)^T$. As we can see, the PFA method is in fact closely related to the method in [BT77], and is only equivalent to the proposed method when $M = 1$ and $m = 1$.

In all the above mentioned methods except SFA, a VAR model needs to be fit for the original high-dimensional time series. There have also been methods developed that do not involve fitting a high-dimensional VAR model. For example, in [DQ18b, DQ18a], dynamic-inner principal component analysis (DiPCA) and DiCCA are developed to extract a lower-dimensional most predictable latent variables. In both methods, the columns of $W$ are extracted sequentially, with a scalar AR predictor for each entry in $x_t$. DiPCA extracts each entry in $x_t$ such that it has maximal covariance between its predicted value using an AR predictor, while DiCCA maximizes the correlation. In fact, it can be shown that DiCCA is a special case of the proposed method where $A_1, A_2, \ldots, A_M$ are diagonal matrices.

Instead of using expected mean squared error as a predictability measure, there have been methods developed using different predictability measures. For example, forecastable component analysis (ForeCA) proposed in [Goe13] utilizes the differential entropy as the predictability (forecastability) measure, which yields the lower bound of the expected squared loss of any estimator. Graph-based predictable feature analysis (GPFA) in [WFW17] maximizes a predictability measure defined in terms of graph embedding. Dynamical component analysis in [CLB19] uses mutual information between the past and future data. The method developed in [Sto01] proposed to use a measure of temporal predictability for blind source separation. In atmospheric, optimal persistence analysis (OPA) maximizes the decorrelation time ([Del01]), and average predictability time decomposition (APTD) maximizes the average predictability time ([DT09a, DT09b]).

## 6.2 Factor models

This work is also closely related to the extensively studied factor models in econometrics. Here we analyze some representative methods and compare them with the proposed method. The early work [Bri81] is a frequency domain approach that extracts dynamic principal components (DPC) as linear combinations of the past and future observations to minimize the mean squared reconstruction error of the original high-dimensional time series. Similar structure exists in [PY16], which is a time domain approach that uses non-causal models. In the later work [PSY19], a causal model is utilized to extract one-sided dynamic principal components (ODPC) as linear combinations of the current and past values of the series that

minimize the reconstruction mean squared error. All of these methods extract latent factors differently from the proposed method, where the extraction of $x_t$ only depends on the current data. In addition, these methods extract latent factors by minimizing the mean squared reconstruction error, while the proposed method minimizes the mean squared prediction error.

In [LY12, LYB11], for standardized high-dimensional time series, $W$ is selected as the eigenvectors corresponding to the largest $m$ eigenvalues in $\sum_{i=1}^{M} \Sigma_i^T \Sigma_i$. In [PY08], the latent factors are identified via expanding the white noise space step by step. Compare to the proposed method, these methods provide little characterization on the lower-dimensional $x_t$. To make predictions on $x_t$, [LYB11] suggest to subsequently build an AR predictor. However, in the proposed method, the extraction and prediction of $x_t$ are achieved simultaneously by solving one optimization problem. A lot of the above analysis were also given in [QDZ+20], where [LY12] is linked to subspace identification. Refer to [SW06, SW11, FHLR00, AW07, BN07] for more general discussions on factor models.

## 6.3 Reduced rank time series

Another related work is reduced rank time series modeling. Early work such as [Rei83, VRW86, AR88, WB04] fit AR models to the vector time series with reduced rank coefficients. Later the reduced rank time series models have been generalized into the structured AR modeling problem ([BLM19, ABDG20, MB16, BDB21]), where regularization terms on the AR model coefficients are imposed to encourage certain structures, such as low rank and sparsity. In summary, most of these papers focus on a parsimonious parametrization on the vector time series models, rather than extracting low-dimensional predictable time series.

## 6.4 State space models

The proposed method also has connections with state space models. There have been many ways to fit a state space model to vector time series, such as expectation-maximization (EM) ([SS82]), and N4SID ([MDMVV89, VODM93]) and CVA approach ([Lar83]). In state space models, there have been methods developed to encourage sparse or low rank structures on the state transition matrix, such as [SGXC18, CLY+17]. Instead of adding regularizations on the state transition matrix, [ARG09, CARR11] directly regularize the latent state to be sparse or low rank. In addition, there have been approaches that consider more general settings. For example, [LM20, KDS18] consider the identification of linear dynamical systems with serially correlated output noise components. In the work of [QDZ+20], state space models are also compared with latent variable models, and it pointed out that subspace identification does not naturally yield reduced dimensional models.

In summary, the key difference between the proposed method and many of the existing methods is that first, it extracts low-dimensional predictable time series without fitting a full-dimensional VAR model; second, the extraction of principal time series and the VAR

modeling of the principal time series are achieved simultaneously by solving the optimization problem.

# 7 Conclusion

In this paper we have described a new method to extract a low-dimensional most predictable time series from high-dimensional time series, in the sense that an auto-regressive model achieves minimum prediction error. The method is heuristic, since the algorithm does not guarantee globally optimal. Numerical examples suggest, however, that the method works very well in practice.

# 8 acknowledgements

We would like to express our appreciation to Professor Peter Stoica for his valuable and constructive suggestions during the preparation of this paper. We also thank Peter Nystrup for pointing us to related work.

# 9 acknowledgements

# Appendix A Derivation of (6)

We show how to derive the expression (6) in this appendix. For simplicity, we ignore the superscript $k+1$ in $A^{k+1}$, $A_i^{k+1}$, $i = 1, \ldots, M$, and $S_\tau^{k+1}$, $\tau \in \mathbf{Z}$, and the superscript $k$ in $W^k$.

When $A$ is fixed, we have

$$\begin{aligned} f(w) &= \mathbf{Tr}\left( -2A \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_M \end{bmatrix} + A \begin{bmatrix} S_0 & S_1^T & \cdots & S_{M-1}^T \\ S_1 & S_0 & \cdots & S_{M-2}^T \\ \vdots & \vdots & \ddots & \vdots \\ S_{M-1} & S_{M-2} & \cdots & S_0 \end{bmatrix} A^T \right) \\ &= -2 \sum_{i=1}^M \mathbf{Tr}(A_i S_i) \\ &\quad + \mathbf{Tr} \begin{bmatrix} S_0 & S_1^T & \cdots & S_{M-1}^T \\ S_1 & S_0 & \cdots & S_{M-2}^T \\ \vdots & \vdots & \ddots & \vdots \\ S_{M-1} & S_{M-2} & \cdots & S_0 \end{bmatrix} \begin{bmatrix} A_1^T A_1 & A_1^T A_2 & \cdots & A_1^T A_M \\ A_2^T A_1 & A_2^T A_2 & \cdots & A_2^T A_M \\ \vdots & \vdots & \ddots & \vdots \\ A_M^T A_1 & A_M^T A_2 & \cdots & A_M^T A_M \end{bmatrix}. \end{aligned}$$

We divide $A_i$, $i = 1, 2, \ldots, M$ into the following submatrices,

$$A_i = \begin{bmatrix} A_{i,11} & A_{i,12} \\ A_{i,21} & A_{i,22} \end{bmatrix} \quad \text{for } i = 1, 2, \ldots, M,$$

where $A_{i,11} \in \mathbf{R}^{k\times k}$, $A_{i,12} \in \mathbf{R}^{k\times 1}$, $A_{i,21} \in \mathbf{R}^{1\times k}$, $A_{i,22} \in \mathbf{R}$. With this notation, we can expand $\mathbf{Tr}(A_iS_i)$ as

$$\begin{aligned}\mathbf{Tr}(A_iS_i) &= \mathbf{Tr}\begin{bmatrix}A_{i,11} & A_{i,12}\\ A_{i,21} & A_{i,22}\end{bmatrix}\begin{bmatrix}W^T\Sigma_iW & W^T\Sigma_iw\\ w^T\Sigma_iW & w^T\Sigma_iw\end{bmatrix}\\ &= w^T(A_{i,22}\Sigma_i)w + (\Sigma_iWA_{i,12} + \Sigma_i^TWA_{i,21}^T)^Tw + d,\end{aligned}$$

where $d$ is a constant. For the second term in $f(w)$, we have

$$\begin{aligned}&\mathbf{Tr}\begin{bmatrix}S_0 & S_1^T & \cdots & S_{M-1}^T\\ S_1 & S_0 & \cdots & S_{M-2}^T\\ \vdots & \vdots & \ddots & \vdots\\ S_{M-1} & S_{M-2} & \cdots & S_0\end{bmatrix}\begin{bmatrix}A_1^TA_1 & A_1^TA_2 & \cdots & A_1^TA_M\\ A_2^TA_1 & A_2^TA_2 & \cdots & A_2^TA_M\\ \vdots & \vdots & \ddots & \vdots\\ A_M^TA_1 & A_M^TA_2 & \cdots & A_M^TA_M\end{bmatrix}\\ &= \mathbf{Tr}(S_0A_1^TA_1 + S_1^TA_2^TA_1 + \cdots + S_{M-1}^TA_M^TA_1) + \mathbf{Tr}(S_1A_1^TA_2 + S_0A_2^TA_2 + \cdots\\ &+S_{M-2}^TA_M^TA_2) + \cdots + \mathbf{Tr}(S_{M-1}A_1^TA_M + S_{M-2}^TA_2^TA_M + \cdots + S_0A_M^TA_M)\\ &= \textstyle\sum_{i,j} S_{j-i}A_i^TA_j,\end{aligned}$$

where $\mathbf{Tr}(S_{j-i}A_i^TA_j)$ can be expanded as

$$\begin{aligned}&\mathbf{Tr}(S_{j-i}A_i^TA_j)\\ &= \begin{bmatrix}W^T\Sigma_{j-i}W & W^T\Sigma_{j-i}w\\ w^T\Sigma_{j-i}W & w^T\Sigma_{j-i}w\end{bmatrix}\begin{bmatrix}A_{i,11}^TA_{j,11} + A_{i,21}^TA_{j,21} & A_{i,11}^TA_{j,12} + A_{i,21}^TA_{j,22}\\ A_{i,12}^TA_{j,11} + A_{i,22}A_{j,21} & A_{i,12}^TA_{j,12} + A_{i,22}A_{j,22}\end{bmatrix}\\ &= (A_{i,12}^TA_{j,11} + A_{i,22}A_{j,21})W^T\Sigma_{j-i}w + (A_{i,11}^TA_{j,12} + A_{i,21}^TA_{j,22})^TW^T\Sigma_{j-i}^Tw\\ &+w^T(A_{i,12}^TA_{j,12} + A_{i,22}A_{j,22})\Sigma_{j-i}w.\end{aligned}$$

Summing all terms, we can obtain the following expression for $f(w)$,

$$f(w) = w^TBw - 2c^Tw + d,$$

where $d$ is a constant and

$$\begin{aligned}B &= \sum_{1\le i,j\le M}(A_{i,12}^TA_{j,12} + A_{i,22}A_{j,22})\Sigma_{j-i} - \sum_{i=1}^M A_{i,22}(\Sigma_i + \Sigma_i^T),\\ c &= \sum_{i=1}^M(\Sigma_iWA_{i,12} + \Sigma_i^TWA_{i,21}^T) - \sum_{1\le i<j\le M}\Sigma_{j-i}^TW(A_{j,11}^TA_{i,12} + A_{i,22}A_{j,21}^T)\\ &\quad - \sum_{1\le i<j\le M}\Sigma_{j-i}W(A_{i,11}^TA_{j,12} + A_{j,22}A_{i,21}^T).\end{aligned}$$

The constant term can be ignored when we want to minimize $f(w)$. It is easy to show that $B \succ 0$.

# References


[ABDG20] P. Alquier, K. Bertin, P. Doukhan, and R. Garnier. High-dimensional VAR with low-rank transition. *Statistics and Computing*, 30(4):1139–1153, 2020.

[AMS09] P. A. Absil, R. Mahony, and R. Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009.

[AR88] S. K. Ahn and G. C. Reinsel. Nested reduced-rank autoregressive models for multiple time series. *Journal of the American Statistical Association*, 83(403):849–856, 1988.

[ARG09] D. Angelosante, S. I. Roumeliotis, and G. B. Giannakis. Lasso-Kalman smoother for tracking sparse signals. In *2009 Conference record of the forty-third asilomar conference on signals, systems and computers*, pages 181–185. IEEE, 2009.

[AW07] D. Amengual and M. W. Watson. Consistent estimation of the number of dynamic factors in a large $N$ and $T$ panel. *Journal of Business & Economic Statistics*, 25(1):91–96, 2007.

[BDB21] S. Barratt, Y. Dong, and S. Boyd. Low rank forecasting. *arXiv preprint arXiv:2101.12414*, 2021.

[BLM19] S. Basu, X. Li, and G. Michailidis. Low rank and structured modeling of high-dimensional vector autoregressions. *IEEE Transactions on Signal Processing*, 67(5):1207–1222, 2019.

[BN07] J. Bai and S. Ng. Determining the number of primitive shocks in factor models. *Journal of Business & Economic Statistics*, 25(1):52–60, 2007.

[BN08] J. Bai and S. Ng. Large dimensional factor analysis. *Foundations and Trends® in Econometrics*, 3(2):89–163, 2008.

[Bri81] D. R. Brillinger. *Time Series: Data Analysis and Theory, Expanded Edition*. Holden-Day, Inc, 1981.

[BT77] G. E. Box and G. C. Tiao. A canonical analysis of multiple time series. *Biometrika*, 64(2):355–365, 1977.

[CARR11] A. Charles, M. S. Asif, J. Romberg, and C. Rozell. Sparsity penalties in dynamical system estimation. In *2011 45th annual conference on information sciences and systems*, pages 1–6. IEEE, 2011.

[Cho12] I. Choi. Efficient estimation of factor models. *Econometric Theory*, 28(2):274–308, 2012.

[CK86] G. Connor and R. A. Korajczyk. Performance measurement with the arbitrage pricing theory: A new framework for analysis. *Journal of financial economics*, 15(3):373–394, 1986.

[CLB19] D. G. Clark, J. A. Livezey, and K. E. Bouchard. Unsupervised discovery of temporal structure in noisy data with dynamical components analysis. *arXiv preprint arXiv:1905.09944*, 2019.

[CLY+17] S. Chen, K. Liu, Y. Yang, Y. Xu, S. Lee, M. Lindquist, B. S. Caffo, and J. T. Vogelstein. An M-estimator for reduced-rank system identification. *Pattern recognition letters*, 86:76–81, 2017.

[Del01] T. DelSole. Optimally persistent patterns in time-varying fields. *Journal of the atmospheric sciences*, 58(11):1341–1356, 2001.

[DQ18a] Y. Dong and S. J. Qin. Dynamic latent variable analytics for process operations and control. *Computers & Chemical Engineering*, 114:69–80, 2018.

[DQ18b] Y. Dong and S. J. Qin. A novel dynamic pca algorithm for dynamic data modeling and process monitoring. *Journal of Process Control*, 67:1–11, 2018.

[DT09a] T. DelSole and M. K. Tippett. Average predictability time. part i: theory. *Journal of the Atmospheric Sciences*, 66(5):1172–1187, 2009.

[DT09b] T. DelSole and M. K. Tippett. Average predictability time. part ii: Seamless diagnoses of predictability on multiple time scales. *Journal of the Atmospheric Sciences*, 66(5):1188–1204, 2009.

[EAS98] A. Edelman, T. A. Arias, and S. T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM journal on Matrix Analysis and Applications*, 20(2):303–353, 1998.

[FHLR00] M. Forni, M. Hallin, M. Lippi, and L. Reichlin. The generalized dynamic-factor model: Identification and estimation. *Review of Economics and statistics*, 82(4):540–554, 2000.

[Goe13] G. Goerg. Forecastable component analysis. In *International Conference on Machine Learning*, pages 64–72, 2013.

[KDS18] O. Kost, J. Duník, and O. Straka. Correlated noise characteristics estimation for linear time-varying systems. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 650–655. IEEE, 2018.

[Lar83] W. E. Larimore. System identification, reduced-order filtering and modeling via canonical variate analysis. In *1983 American Control Conference*, pages 445–451. IEEE, 1983.

[LM20] J. Lin and G. Michailidis. System identification of high-dimensional linear dynamical systems with serially correlated output noise components. *IEEE Transactions on Signal Processing*, 68:5573–5587, 2020.

[LY12] C. Lam and Q. Yao. Factor modeling for high-dimensional time series: inference for the number of factors. *The Annals of Statistics*, 40(2):694–726, 2012.

[LYB11] C. Lam, Q. Yao, and N. Bathia. Estimation of latent factors for high-dimensional time series. *Biometrika*, 98(4):901–918, 2011.

[MB16] I. Melnyk and A. Banerjee. Estimating structured vector autoregressive models. In *Proc. Intl. Conf. Machine Learning*, pages 830–839, 2016.

[MDMVV89] M. Moonen, B. De Moor, L. Vandenberghe, and J. Vandewalle. On-and off-line identification of linear state-space models. *International Journal of Control*, 49(1):219–232, 1989.

[PB87] D. Pena and G. E. Box. Identifying a simplifying structure in time series. *Journal of the American statistical Association*, 82(399):836–843, 1987.

[PSY19] D. Peña, E. Smucler, and V. J. Yohai. Forecasting multiple time series with one-sided dynamic principal components. *Journal of the American Statistical Association*, 2019.

[PY08] J. Pan and Q. Yao. Modelling multiple time series via common factors. *Biometrika*, 95(2):365–379, 2008.

[PY16] D. Peña and V. J. Yohai. Generalized dynamic principal components. *Journal of the American Statistical Association*, 111(515):1121–1131, 2016.

[QDZ+20] S. J. Qin, Y. Dong, Q. Zhu, J. Wang, and Q. Liu. Bridging systems theory and data science: A unifying review of dynamic latent variable analytics and process monitoring. *Annual Reviews in Control*, 2020.

[Rei83] G. Reinsel. Some results on multivariate autoregressive index models. *Biometrika*, 70(1):145–156, 1983.

[RW15] S. Richthofer and L. Wiskott. Predictable feature analysis. In *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, pages 190–196. IEEE, 2015.

[SGXC18] Q. She, Y. Gao, K. Xu, and R. Chan. Reduced-rank linear dynamical systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[SS82] R. H. Shumway and D. S. Stoffer. An approach to time series smoothing and forecasting using the EM algorithm. *Journal of time series analysis*, 3(4):253–264, 1982.

[Sto01] J. V. Stone. Blind source separation using temporal predictability. *Neural computation*, 13(7):1559–1574, 2001.

[SW06] J. H. Stock and M. W. Watson. Forecasting with many predictors. *Handbook of economic forecasting*, 1:515–554, 2006.

[SW11] J. H. Stock and M. Watson. Dynamic factor models. *Oxford handbook on economic forecasting*, 2011.

[Tat14] W. O. Tatum. Ellen R. Grass lecture: Extraordinary EEG. *The Neurodiagnostic Journal*, 54(1):3–21, 2014.

[Tep02] M. Teplan. Fundamentals of EEG measurement. *Measurement science review*, 2(2):1–11, 2002.

[TH97] N. F. Thornhill and T. Hägglund. Detection and diagnosis of oscillation in control loops. *Control Engineering Practice*, 5(10):1343–1354, 1997.

[THZ03] N. F. Thornhill, B. Huang, and H. Zhang. Detection of multiple oscillations in control loops. *Journal of Process control*, 13(1):91–100, 2003.

[UM14] K. Usevich and I. Markovsky. Optimization on a Grassmann manifold with application to system identification. *Automatica*, 50(6):1656–1662, 2014.

[VODM93] P. Van Overschee and B. De Moor. Subspace algorithms for the stochastic identification problem. *Automatica*, 29(3):649–660, 1993.

[VRW86] R. P. Velu, G. C. Reinsel, and D. W. Wichern. Reduced rank models for multiple time series. *Biometrika*, 73(1):105–118, 1986.

[WB04] Z. Wang and D. A. Bessler. Forecasting performance of multivariate time series models with full and reduced rank: An empirical examination. *International Journal of Forecasting*, 20(4):683–695, 2004.

[WFW17] B. Weghenkel, A. Fischer, and L. Wiskott. Graph-based predictable feature analysis. *Machine Learning*, 106(9-10):1359–1380, 2017.

[WS02] L. Wiskott and T. J. Sejnowski. Slow feature analysis: Unsupervised learning of invariances. *Neural computation*, 14(4):715–770, 2002.