ERASMUS UNIVERSITY ROTTERDAM

ERASMUS SCHOOL OF ECONOMICS

[International Bachelor Econometrics and Operations Research]

Bachelor Thesis

Forecasting Accuracy Of Sparse Principal Component Analysis In U.S Inflation

Ruoyu Zhi

424107

Supervisor: Dr. A.M. Schnucker

Second assessor: Dr. D.J.C. van Dijk

July 7, 2019

Contents

ABSTRACT

In order to forecast target variables using a dataset with a large number of variables, dimensionality reduction methods are often applied to extract factors from the dataset and those factors are used in forecasting models. The sparse principal component analysis is such a dimensionality reduction technique that has the advantage of interpretability of its components. In this paper, I apply sparse principal component to the Stock and Watson dataset where 132 variables are used to forecast the U.S inflation and compare its predicting performances with the ordinary principal component analysis and the partial least squares. Empirical results show that the sparse principal component analysis brings improvements in forecasting accuracy compared to the ordinary principal component analysis. And it works exceptionally well in the cases where the number of variables is greater than the number of observations.

1 Introduction

When forecasting macroeconomic variables such as inflation, a dataset with a large number of predictors is often available, and sometimes the number of variables is even greater than the number of observations. For example, the Stock and Watson (2005) dataset contains 132 monthly U.S macroeconomic series. Having more available variables means having more available information and this provides a possibility to enhances the predicting performance. However, it also aggravates the problem of high dimensionality for forecasting concerns. Regressing the response variables on too many explanatory variables might result in over-fitting and reduce the out-of-sample forecasting performance. To deal with the problem caused by too many predictors, one choice is to use factor models. By applying factor models, the large data set of predictors is reduced to a lower-dimensional set consisting of several factors and the response variables are regressed on those factors.

Factor models extract from the data set the informative common factors that are representative for the data. The most popular and classic approach to obtain the common factors is called Principal Components Analysis (PCA). PCA applies the orthogonal transformation to transform the predictors to linearly uncorrelated factors (principal components) capturing maximal variances and ensuring minimal information loss. Stock and Watson (2002b) show that for macroeconomic forecasting, applying PCA in a large set of variables brings significant improvement in forecasting accuracy comparing with the conventional models which use a small number of variables, and PCA gives consistent estimated factors. Bai and Ng (2008) evaluate the performances of PCA and its variations for macroeconomic forecasting.

One disadvantage of PCA is that it is designed to extract the common information from the whole set of predictors, but it is not constructed for prediction, meaning that some of the extracted factors might not provide information for predicting target variables. Wold (1966) proposes a statistical method named partial least squares (PLS), which is a dimension reduction scheme taking into consideration the aim of prediction. PLS is suitable for macroeconomic forecasting since it is valid even when the number of predictors is larger than the number of observations. Groen and Kapetanios (2016) and Fuentes, Poncela, and Rodríguez (2015) show that PLS usually has relatively good performance for forecasting U.S inflation in terms of mean squared forecast error. Kelly and Pruitt (2015) confirm the predictive performance of the three-pass regression filter (3PRF), which is a one-component PLS.

Another drawback of the Principal Component Analysis is that when the number of predictors is large, the obtained PCs might be difficult to interpret. Using PCA, one principal component is a linear combination of all the predictors with typically non-zero loadings, and so interpretation becomes challenging when having numerous predictors. Cadima and Jolliffe (1995) show that in order to better interpret the principal components, truncating those predictors with small-magnitude loadings and dealing with the approximated PCs which are linear combinations of the remaining predictors is unreliable. However, the idea of variable selection is helpful. A bright variable selection scheme called the *lasso* is introduced by Tibshirani (1996), which is a penalized least squares approach with a constraint of the L_1 norm of the coefficients which can result in zero coefficients. Nevertheless, the *lasso* has the problem that when the number of predictors p is larger than the number of samples n, it can at most choose n non-zero coefficients which is inappropriate because sometimes we want to choose more than n variables. Zou and Hastie (2005) generalize the *lasso* to another variable selection method named the *elastic net* by adding the *ridge* penalties to the *lasso*, which does not have the aforementioned limitation.

Zou, Hastie, and Tibshirani (2006) propose the sparse principal component analysis (SPCA), which combines the *lasso* penalty and *elastic net* with the PCA and gives principal components with sparse loadings. The sparsity of loadings can be controlled flexibly and at most p variables can be chosen with non-zero loadings even when $p > n$. Zou, Hastie, and Tibshirani (2006) prove that their algorithm of sparse principal component analysis has efficiency in computation and the ability in identification of crucial predictors. With sparse loadings, the sparse principal components are easier to be interpreted than principal components obtained from the ordinary PCA. However, just like PCA, SPCA is designed without considering any correlation between the dataset and the response variable. Thus, we cannot make any assumption about its predictive power. This brings up the question how SPCA performs in forecasting. Therefore, I focus on investigating the accuracy of out-of-sample forecasting of SPCA for the U.S inflation, comparing with the forecasting performances of PCA and PLS.

Comparisons between the forecasting performances of these approaches are made for Stock and Watson (2005) dataset. Stock and Watson's dataset contains 132 monthly U.S macroeconomic time series for the period from January 1959 to December 2003. I only use the data from January 1960 to December 2003 because some time series are unavailable before January 1960. Therefore, I have monthly observations for 43 years, so the total number of observations $T = 528$. To obtain stationary time series the original time series are transformed. For detailed information on how they are transformed, you can refer to Stock and Watson (2005).

Fuentes, Poncela, and Rodríguez (2015) also use the Stock and Watson dataset and perform PCA and PLS for seven different subsamples. I use the same subsamples and apply PCA, PLS and SPCA. Then I compare my forecasting results with the results from Fuentes, Poncela, and Rodríguez (2015) and both differences and similarities are found. Moreover, I employ different subsamples where the number of predictors is greater than the number of samples and investigate the forecasting performances of PCA, PLS and SPCA using these subsamples. The empirical result indicates that a significant refinement for predicting accuracy can be obtained by applying sparsity in the principal component analysis. For the subsamples used by Fuentes, Poncela, and Rodríguez (2015), PLS usually performs the best and SPCA is better than PCA. For the subsamples where $p > n$, I find that the SPCA has the best performance among all investigated methods.

The structure of this thesis is as follows. Section 2 is about the data used. Section 3 presents how forecasting is done and how factor models and PCA work, followed by the explanations of PLS, SPCA, and forecast evaluation methods. Section 4 provides evaluation results. The conclusion is in Section 5.

2 Methodology

2.1 General version of the forecasting model

I want to perform h-step forecasting for the target variable y at time t, which means I want to predict y_{t+h}, knowing the information up to time t denoted by X that is the set of predictors. X is a $t \times p$ matrix, and p is the number of predictors in X. From X I extract a set of K factors denoted as $\hat{F}$, and $\hat{F} = [\hat{F}_1, ..., \hat{F}_t]'$, where $\hat{F}_i$ are $K \times 1$ vectors, $i = 1, ..., t$. Then the general version of the forecasting model is written as

$$y_{t+h} = \mu + \phi(L)y_t + \beta'(L)\hat{F}_t + \eta_{t+h}. \tag{1}$$

Therefore, the target variable to be predicted h-step ahead from period t, y_{t+h}, is regressed on the constant μ, the lags of y_t and the vector of factors $\hat{F}_t$ and their lags. η_{t+h} denotes the forecasting error.

2.2 Factor Models

The factor models are presented as follows:

$$X = F\Lambda' + \varepsilon_t, \tag{2}$$

where X is the $t \times p$ matrix of observed predictors up to time t; $F = [F_1, ..., F_t]'$ is the $t \times K$ matrix of K common factors extracted from X, and F_i is a $K \times 1$ vector, $i = 1, ..., t$; $\Lambda = [\lambda_1, \lambda_2, ...\lambda_K]$ denotes the $p \times K$ matrix of factor loadings and each λ_i is the $p \times 1$ vector of loadings of factor i, $i = 1, ..., K$; ϵ_t is the idiosyncratic disturbance matrix. It is assumed that disturbances and each of the common factors are uncorrelated. Also the idiosyncratic disturbances are serially uncorrelated. Following this brief introduction of factor models, I discuss the variants of them.

2.3 Principal Component Analysis

The factors extracted from PCA are linear combinations of the predictors. Those factors can be used in a linear regression to forecast the target variable. At time period t, estimating the factors is equivalent to solving the maximization of $Ntr(\tilde{\Lambda}'X'X\tilde{\Lambda})$ subject to $\tilde{\Lambda}'\tilde{\Lambda} = I_r$, where $tr(\cdot)$ is the matrix trace and X denotes the matrix of all observed predictors up to time t, with each vector of the predictors is standardized such that it has zero mean and variance equal to one. This problem is solved by setting $\tilde{\Lambda}$ equal to the eigenvectors of $X'X$ corresponding to its K largest eigenvalues. Therefore, the estimated factors are $\hat{F}_t = \hat{\Lambda}'X$.

From the above explanation of PCA, it is easy to notice that the factors extracted from X does not consider any correction between those factors and the variables to be predicted. The factors are merely representatives for the dataset X, and as what Fuentes, Poncela, and Rodríguez (2015) say, we do not even know whether they have any forecasting power over the response variable.

2.4 Partial Least Squares

Partial least squares (PLS) is a scheme for dimension reduction inspired by Wold (1966) that extracts orthogonal unobserved components. Similarly to PCA, PLS components are obtained from eigenvector decomposition, but in consecutive steps. One of the differences between PLS and PCA is that not only information of the dataset is considered in PLS, but also the correlation between the dataset and the forecasting variable. Therefore, PLS provides one possibility to obtain better forecasting results than PCA and it is interesting to see how PLS performs in forecasting. Fuentes, Poncela, and Rodríguez (2015) explain how PLS work in general and employ three specific PLS approaches .

To find the first PLS component, the eigenvalue decomposition is performed on the matrix:

$$M = X'YY'X, \tag{3}$$

where the $t \times p$ matrix X is again the information up to time t and $Y = (y_1, ..., y_t)'$ is the vector of forecasting variable up to tome t. $\hat{f}_{jt}^{PLS}$, which denotes the first PLS component, is computed by a linear combination of X and the M's first eigenvector. To obtain the second PLS component, the first PLS components are regressed on Y and on each predictor in X respectively. Then the eigenvalue decomposition is applied for the residuals of those regressions, where the unexplained information is in those residuals. This procedure continues until the last PLS component is found.

I use both static and dynamic (considering the dynamics of time series) approaches of PLS used by Fuentes, Poncela, and Rodríguez (2015) for forecasting, which are different in the ways the predictors are used. The forecasting model is given as follows:

$$y_{t+h} = \beta'(L)Z_t + \phi(L)y_t + u_{t+h}, \tag{4}$$

$$Z_t = W'X_t. \tag{5}$$

The h-step ahead forecasting equation for y is given by Equation (4), which contains y_t and its lags, and the components Z_t and their lags. The unobserved components in Equation (5) are $Z_t = \hat{f}_t^{PLS}$, the PLS factors. Z_t are linear combinations of X_t, the $p \times 1$ vector of predictors observed at time t, and $p \times K$ weighting matrix W. Now I introduce one static approach and two dynamic approaches for forecasting using PLS.

The static approach (SA) applies PLS between Y_{t+h} and the original X. The lags of y_t are in Equation (4) and are not included in Equation (5) for forming Z_t. $M = X'Y_hY_h'X$, where $Y_h = (y_{h+1}, \ldots, y_{T+h})$.

The first dynamic approach, (DA_1) uses PLS between Y_h and X_e, which includes the lags of the target variable. The lags of the target variables are excluded from Equation (4) and are as predictors in Equation (5).

The second dynamic approach, (DA_2) first performs an AR(p) regression for the target variable Y_h. Then it applies PLS between the residuals obtained from the AP(p) process and the original X. Equation (4) includes the lags of the target variable.

I give examples of how those three approaches are actually calculated, using a simple case where the number of PLS components is $k = 1$, the number of predictors is $N = 2$ and the number of lags is 1. The examples for each of the approaches are as follows.

For SA,

$$y_{t+h} = \beta_1 Z_t + \phi y_t + u_{t+h}, \tag{6}$$

$$Z_t = w_1 x_{1t} + w_2 x_{2t}, \tag{7}$$

where $Z_t = \hat{f}_t^{PLS}$. $w_i, i = 1, 2$, denote the weights of the predictors in the PLS component. The following optimization problem needs to be solved in order to find the direction vector x,

$$w = \arg\max_{w} w' X_t' Y_{t+h} Y_{t+h}' X_t w \quad \text{subject to } w'w = 1 \tag{8}$$

where $w = (w_1, \ldots, w_r)'$ and in this case $r = 2$. The objective function of optimization problem (8) is given by

$$\max_{(w_1 w_2)} w_1^2 \left[\sum_{t=1}^{T} x_{1t} y_{t+h}\right]^2 + 2 w_2 w_2 \left[\sum_{t=1}^{T} x_{1t} y_{t+h}\right]\left[\sum_{t=1}^{T} x_{2t} y_{t+h}\right] + w_2^2 \left[\sum_{t=1}^{T} x_{2t} y_{t+h}\right]^2 + \lambda\left(w_1^2 + w_2^2 - 1\right). \tag{9}$$

The direction vector w of the first PLS component is obtained by solving this problem. Therefore, $f_t^{PLS} = Z_t$ can be computed and it is used in Equation (6) as the set of explanatory variables.

For DA_1 the model is given by

$$y_{t+h} = \beta_1 Z_t + u_{t+h}, \tag{10}$$

$$Z_t = w_1 x_{1t} + w_2 x_{2t} + w_3 y_t, \tag{11}$$

where the target variables are included in X and are not in the forecasting equation.

DA_2 incorporates the AR(p) process and the optimization problem for the direction vector w is

$$w = \arg\max_{w} w' X_t' Y Y' X_t w \quad \text{subject to } w'w = 1, \tag{12}$$

where $Y = [Y_{t+h} - \phi Y_t]$. And the objective function is given by

$$\max_{(w_1 w_2)} \left[w_1 \left[\sum_{t=1}^{T} x_{1t}\left(y_{t+h} - \phi y_t\right)\right] + w_2 \left[\sum_{t=1}^{T} x_{2t}\left(y_{t+h} - \phi y_t\right)\right]\right]^2 + \lambda\left(w_1^2 + w_2^2 - 1\right). \tag{13}$$

All vectors of predictors in X and X_e of these three approaches are standardized as well. One algorithm of PLS is developed by Wold (1975) and called NIPALS. I make use of this algorithm to compute the PLS factors.

2.5 Sparse Principal Component Analysis

Proposed by Zou, Hastie, and Tibshirani (2006), sparse principal component analysis (SPCA) is a method based on PCA, the *lasso* and the *elastic net*, which instead of giving nonzero loadings to all predictors for each component, allows sparse loadings. The two-objective optimization problem for this method is:

$$\begin{aligned}(\widehat{\mathbf{A}}, \widehat{\mathbf{B}}) = \arg\min_{\mathbf{A},\mathbf{B}} \sum_{i=1}^{n} \left\|\mathbf{x}_i - \mathbf{A}\mathbf{B}^T \mathbf{x}_i\right\|^2 + \lambda \sum_{j=1}^{K} \|\beta_j\|^2 + \sum_{j=1}^{K} \lambda_{1,j} \|\beta_j\|_1 \\ \text{subject to } \mathbf{A}^T\mathbf{A} = I_{K\times K}.\end{aligned} \tag{14}$$

This is call the SPCA criterion. The number of predictors is p and the number of components is K. $A_{p\times K} = [\alpha_1, ..., \alpha_k]$ is the matrix of loadings of factors that transforms the factors to the size of original data. And $B_{p\times K} = [\beta_1, ..., \beta_k]$ denotes the matrix of loadings of predictors, which transforms the predictors to factors. x_i is the ith row vector of X and n is the number of observations. In this optimization problem, the term $\lambda \sum_{j=1}^{k} \|\beta_j\|^2$ are the *ridge* penalties and

$\sum_{j=1}^{k} \lambda_{1,j} \|\beta_j\|_1$ are the *lasso* penalties, which controls the sparsity of the loadings. $\lambda > 0$ should be the same for all k components and $\lambda_{1,j} > 0$ can be different for each component, meaning that the loadings for different components can be penalized to different sparsities. The constraint $\mathbf{A}^T\mathbf{A} = I_{K\times K}$ is call the orthogonal constraint, which ensures orthogonality of the matrix $\mathbf{A}$. Moreover, standardization is also applied for each predictor in X here.

From the SPCA criterion (14) one can see that similarly to PCA, SPCA does not consider any correlation between the dataset and the forecasting variable. Given those penalties, the goal of the SPCA criterion is to minimize the sum of squared difference between the original data X and the transformed data. And if $\mathbf{B}$ is set equal to $\mathbf{A}$, and λ and $\lambda_{1,j}, j = 1, ..., K$ are all 0, then

$$\sum_{i=1}^{n} \left\| \mathbf{x}_i - \mathbf{A}\mathbf{B}^T\mathbf{x}_i \right\|^2 = \sum_{i=1}^{n} \left\| \mathbf{x}_i - \mathbf{A}\mathbf{A}^T\mathbf{x}_i \right\|^2. \tag{15}$$

Together with the orthogonal constraint, the minimizer of $\mathbf{A}$ in Equation (15) is just equal to the first K loading vectors of PCA.

The general algorithm to finding the optimal solution for SPCA provided by Zou, Hastie, and Tibshirani (2006) solves the optimization problem by turning it into *elastic net* problems, which can be solved by the *LARS-EN* algorithm proposed by Zou and Hastie (2005). I make use of the MATLAB toolbox made by Sjöstrand et al. (2018), which implements the general SPCA algorithm and the SPCA algorithm with soft-thresholding based on Zou, Hastie, and Tibshirani (2006).

3 Empirical Application

3.1 Forecast model and subsamples

To see performances of the above approaches in forecasting the U.S inflation, I apply the Stock and Watson (2005) dataset. The variable to be predicted is y, the U.S logarithm of the consumer price index (CPI). According to Stock and Watson (2002a), the target variable is assumed to be integrated of order 2 such that for h-step ahead forecasting:

$$y_{t+h}^h = \frac{1200}{h}\left(y_{t+h} - y_t\right) - 1200\left(y_t - y_{t-1}\right). \tag{16}$$

Moreover,

$$z_t = 1200\left(y_t - y_{t-1}\right) - 1200\left(y_{t-1} - y_{t-2}\right). \tag{17}$$

The final forecasting model is based on regressing y_{t+h}^h on a constant, z_t and its lags, and the estimated factors and their lags. Therefore, following Fuentes, Poncela, and Rodríguez (2015), the h-step ahead forecast model at time t is given by

$$y_{t+h}^h = \mu + \phi(L)z_t + \beta'(L)\hat{F}_t, \tag{18}$$

where μ is the constant; z_t is obtained from (16) and $\phi(L)$ is the lag polynomial for it; $\hat{F}_t$ are the factors extracted at t and $\beta'(L)$ denotes the lag polynomials for them. One needs to note that the number of lags of z_t and $\hat{F}_t$ can be different and the maximum lag is 6 both. For each approach, the numbers of lags as well as the number of factors/components included are selected at every time period t, which means the final forecast model may change each time step, and details are discussed in Section 3.2. Additionally, 4 forecast horizons are considered, $h = 1, 6, 12, 24$.

Table 1: Estimation and forecast subsamples, for h-step ahead forecast

SS	Estimation subsample	Forecast subsample
M1	1960:03 to 1970:03-h	1970:03 to 1980:12
M2	1960:03 to 1980:03-h	1980:03 to 1990:12
M3	1960:03 to 1990:03-h	1990:03 to 2000:12
M4	1960:03 to 1970:03-h	1970:03 to 1990:12
M5	1960:03 to 1970:03-h	1970:03 to 2000:12
M6	1960:03 to 1980:03-h	1980:03 to 2000:12
M7	1960:03 to 1970:03-h	1970:03 to 2003:12

Given the current time period t, the factors, as well as their lags, are computed using the loadings calculated from the predictor matrix with information from the beginning up to t. In order to compare with Fuentes, Poncela, and Rodríguez (2015) I use the same seven forecast subsamples which are listed in Table 1. The initial beginning period of the estimations is always March 1960, while the first and the last forecast periods can be different for each subsample. The column "Estimation subsample" in Table 1 are the first estimation subsamples and those estimation subsamples are expanded with the new observation every time step, namely expanding window is applied here.

3.2 Model estimation and selection

In this subsection, I describe in detail how the forecast models are estimated and selected for each approach for the h-step ahead forecast, provided that the current time period is t.

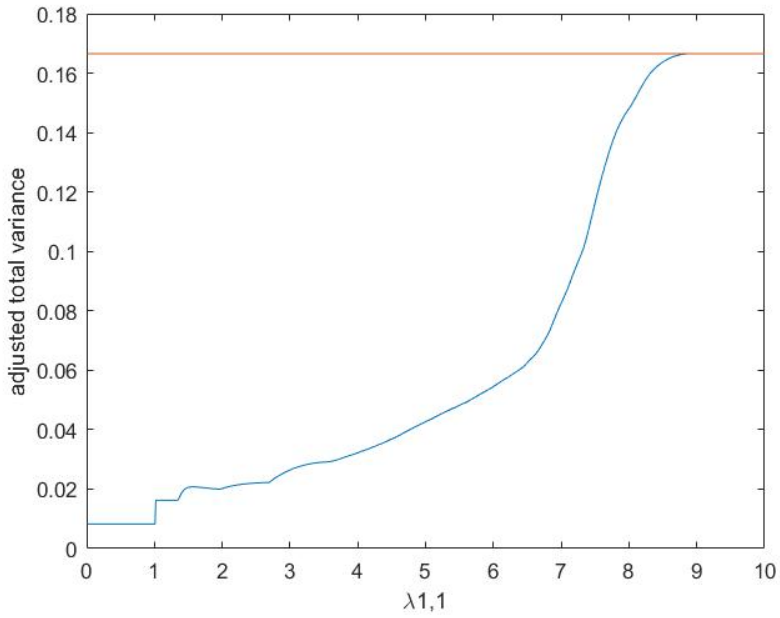
For PCA, the first 10 principal components are always extracted using information up to t. Then I need to decide how many components to include, and how many lags to use for z_t and the components separately. Define $L_z \in \{1, ..., 6\}$ as the number of lags of z_t, $L_f \in \{1, ..., 6\}$ the number of lags of components, and $k \in \{1, ..., 10\}$ the first k principal components to be included. The forecast model (18) is estimated for all combinations of L_z, L_f and k, and the combination with the lowest BIC (Bayesian information criterion) is chosen as the final forecast model.

For each of the three PLS approaches, two types of forecast models are estimated at each time, namely the model using only the first PLS component and the model using both the first and second PLS component. This is done since according to Fuentes, Poncela, and Rodríguez (2015), with the first two components most of the best forecasting results are obtained for PLS. Therefore, for each type, the number of factors is fixed, and the lags of z_t and the factors are selected in accordance to the minimum BIC the same way as PCA. Additionally, for the second dynamic approach (DA_2) with AR(p) process, define $L_z \in \{1, ..., 6\}$ as the number of lags of z_t, $L_f \in \{1, ..., 6\}$ the number of lags of components, and $p \in \{1, ..., 6\}$ the number of lags for the AR process. The combination of L_z, L_f and p with the minimum BIC is chosen for each type of forecast model.

While principal components computed from the ordinary PCA are uncorrelated, SPCA does not restrict the sparse components to be uncorrelated. For the ordinary PCA, the total variance is just the sum of the variances explained by each component. But due to correlation, this is too optimistic when computing the total variance explained by sparse components. Zou, Hastie, and Tibshirani (2006) calculate the total variance in a different way, which considers the correlations between the sparse principal components, and this is called the adjusted total variance. Figure 1 gives an example of how the adjusted total variance explained by the first sparse principal component changes along with $\lambda_{1,1}$ in

equation (14), estimated from the dataset from March 1960 to March 1970. We can see that the adjusted total variance reaches its peak when λ_1 is approximately 9 and has a seemingly increasing shape (not strictly).

Figure 1: Adjusted total variance explained by the first sparse principal component extracted from the sample of 1960:03 to 1970:03. The horizontal line is the adjusted variance explained by the first ordinary principal component.



Algorithm 1 : $\lambda_{1,j}$ searching

Define the wanted proportion δ, $0 < \delta < 1$, and the number of sparse principal components k, $k > 0$.
Compute the adjust total variances explained by the j-th ordinary principal component and denote them as $V_{0,j}$, $j = 1, 2, ..., k$;
$\lambda = 9$;
$\lambda_1 = (0, 0, ..., 0)'$ a $k \times 1$ vector;
$found = (0, 0, ..., 0)'$ a $k \times 1$ vector;
while $found \neq (1, 1, ..., 1)'$ **do**
  Compute the adjusted total variances explained by the j-th sparse principal component according to λ, and denote them as V_j, $j = 1, 2, ..., k$;
  for $i = 1 : k$ **do**
    if $\frac{V_j}{V_0} <= \delta$ and $found(i) = 0$ **then**
      $found(i) = 1$;
      $\lambda_1(i) = \lambda$
    end if
    $\lambda_{1,j} = \lambda_{1,j} - 0.2$;
  end for
end while

I aim to investigate the predictive power of SPCA. However, SPCA with different sparsities gives different sparse principal components and then different foresting results. And it is difficult to say whether a more sparse SPCA or a less sparse SPCA will have better forecasting performance. To compare SPCA with PCA and to see how forecasting performances differ for SPCS with different sparsities, the adjusted total variance explained by sparse components are chosen such that they are different proportions of the adjusted total variance explained by the ordinary principal components. Algorithm 1 summarizes the procedure. A grid search for $\lambda_{1,j}$ for the j-th sparse principal component is

done, such that the adjusted total variance explained by it, divided by the adjusted total variance explained by the j-th ordinary principal component is at most a predefined proportion δ, $0 < \delta < 1$. Following the suggestion from Zou, Hastie, and Tibshirani (2006) that the *ridge* penalty term coefficient λ in Equation 14 should be a small positive number and the it does not affect the output very much, so it is set to be 0.01. Empirical results with setting λ to 0.001 and 0.1 confirms that the output does not change much for varying λ.

One might have noticed that the search starts from $\lambda = 9$. This is because empirical results show that when $\lambda_{1,j} \geqslant 9$, the j-th sparse principal component almost always explains as 100% of the adjusted total variance as the j-th ordinary principal component explains (see Figure (1)). One might also doubt about why $\lambda_{1,j}$ decreases by 0.2 every step and not a smaller number. This choice is due to that although the SPCA algorithm developed by Zou, Hastie, and Tibshirani (2006) and implemented by Sjöstrand et al. (2018) is said to be efficient, it is still relatively computationally expensive. Choosing 0.2 but not some other smaller number does result in more impreciseness, but I have to compromise to the very limited time and the computational power of my devices. Because of the same reason stated above and the fact that the first 2 sparse principal components can explain a large proportion of the total variance explained, I restrict $k \in \{1, 2\}$, meaning that only the first 2 sparse principal components are extracted. Moreover, four different values of δ are used ranging from the case where sparse PC explains most of the adjust total variance explained by PC to the case where the proportion is less than a half, namely $\delta \in \{0.50, 0.65, 0.80, 0.95\}$. For each δ, Algorithm 1 is performed to determine the corresponding $\lambda_{1,j}$, $j = 1, 2$, then the number of components and numbers of lags of components and z_t included in the forecast model are determined by the same way of PCA by BIC. It needs to be mentioned that although compensated with impreciseness, for each value of δ the entire computation of SPCA is still about 10 hours.

3.3 Evaluation methods

This subsection explains how the forecasting performances are evaluated. As the benchmark, I perform the following AR(4) model for each h-step forecast:

$$y_{t+h}^{h} = \mu + \phi_1 z_t + \phi_2 z_{t-1} + \phi_3 z_{t-2} + \phi_4 z_{t-3}. \tag{19}$$

Then I apply the relative mean-squared forecast errors (RMSE) as a measure for the forecast performance, which is computed as follows:

$$RMSE(approach) = \frac{MSE(approach)}{MSE(AR(4))} \tag{20}$$

Therefore, a RMSE less than 1 means that the forecasting performance of the corresponding approach is better than that of AR(4).

The Diebold and Mariano test, proposed byDiebold and Mariano (1995) tests the null hypothesis that the two models have equal forecast accuracy. Suppose there are two forecasts $\{\widehat{y}_{1t}\}_{t=1}^{T}$ and $\{\widehat{y}_{2t}\}_{t=1}^{T}$ for the time series $\{y_t\}_{t=1}^{T}$, and the corresponding forecast errors $\{e_{1t}\}_{t=1}^{T}$ and $\{e_{2t}\}_{t=1}^{T}$. The loss functions are defined as $g(e_{1t}) = e_{1t}^2$ and $g(e_{2t}) = e_{2t}^2$. Then the loss differential is

$$d_t \equiv \left[g\left(e_{1t}\right) - g\left(e_{2t}\right)\right]. \tag{21}$$

The null hypothesis of equal forecast accuracy is equivalent to the null hypothesis that the loss differential series' population mean is zero. Under the assumption of short memory and stationary covariance of the loss differential series

$$\sqrt{T}(\overline{d}-\mu) \xrightarrow{d} N\left(0, 2\pi f_d(0)\right), \tag{22}$$

where μ is the population mean of the loss differential and

$$\overline{d} = \frac{1}{T}\sum_{t=1}^{T}\left[g\left(e_{it}\right) - g\left(e_{jt}\right)\right] \tag{23}$$

is the sample mean of the loss differential series. And

$$f_d(0) = \frac{1}{2\pi}\sum_{\tau=-\infty}^{\infty}\gamma_d(\tau) \tag{24}$$

denotes the spectral density of the loss differential with 0 frequency and $\gamma_d(\tau)$ is the autocovariance. When the sample is large enough, $\overline{d} \xrightarrow{d} N\left(\mu, 2\pi f_d(0)\right)$. Then the null hypothesis has the $N(0,1)$ test statistic equal to

$$S_1 = \frac{\overline{d}}{\sqrt{\frac{2\pi \widehat{f}_d(0)}{T}}}. \tag{25}$$

In macroeconomic forecasting, not only prediction errors are essential, forecasting directions of changes in the time series can also be of interest. Pesaran and Timmermann (1992) develop a non-parametric and distribution-free test for the null hypothesis that given two time series x_t and y_t, X_t has no predictive power in forecasting y_t. First the indicator function if defined as

$$I(\cdot) = \begin{cases} 1 & \text{if } \cdot > 0 \\ 0 & \text{otherwise} \end{cases} \tag{26}$$

Then

$$\hat{P} = n^{-1}\sum_{t=1}^{n} I\left(y_t x_t\right) \tag{27}$$

is the proportion that x_t predicts the sign of y_t correctly. And the estimator of $\hat{P}$'s expectation is

$$\widehat{P}_* = \widehat{P_y}\widehat{P_x} + \left(1-\widehat{P_y}\right)\left(1-\widehat{P_x}\right) \tag{28}$$

under the null hypothesis, where $\widehat{P_y} = n^{-1}\sum_{t=1}^{n} I\left(y_t\right)$ and $\widehat{P_x} = n^{-1}\sum_{t=1}^{n} I\left(x_t\right)$. Moreover, the variances of $\hat{P}$ is

$$\hat{V}(\hat{P}) = n^{-1}\widehat{P}_*\left(1-\widehat{P}_s\right) \tag{29}$$

and the variance of $\hat{P}_*$ is

$$\hat{V}(\hat{P}_*) = n^{-1}(2\widehat{P_y}-1)\widehat{P_x}(1-\widehat{P_x}) + n^{-1}(2\widehat{P_x}-1)^2\widehat{P_y}(1-\widehat{P_y}) + 4n^{-2}\widehat{P_y}\widehat{P_x}(1-\widehat{P_y})(1-\widehat{P_x}) \tag{30}$$

Then the test statistic is calculated as

$$S_n = \frac{\hat{P}-\widehat{P}_*}{\left[\hat{V}(\hat{P}) - \hat{V}\left(\widehat{P}_*\right)\right]^{\frac{1}{2}}}, \tag{31}$$

which follows the standard normal distribution.

3.4 Forecast results

In this subsection, the forecast results are exhibited. Table 2 gives the average ratios of the adjusted total variance explained by the first two sparse PC over those explained by the first two ordinary PC for each δ. All ratios are smaller than but close to from the corresponding δ, which is expected due to the nature of the applied algorithm. Table 3 shows the average number of variables with non-zero weights of the first two sparse principal components selected by sparse principal component analysis for each δ. The average number of variables in the first components is greater than that in the second components. Especially when $\delta = 0.50$, the average number of variables in the second components is less than half of that in the first components.

Table 2: The average ratios of the adjusted total variance explained by the first two SPC over those explained by the first two ordinary PC.

	δ			
	0.95	0.80	0.65	0.50
1st sparse PC	0.9106	0.7651	0.6220	0.4761
2nd sparse PC	0.9315	0.7842	0.6337	0.4867

Table 3: The average number of variables for each δ for the first two sparse principal components

	delta			
	0.95	0.80	0.65	0.50
1st sparse PC	108.34	88.59	69.33	46.29
2nd sparse PC	94.95	60.39	36.89	20.78

Table 4: RMSE, $h = 1$

		PLS			SPCA			
period	PC	SA	DA_1	DA_2	$\delta = 0.50$	$\delta = 0.65$	$\delta = 0.80$	$\delta = 0.95$
70.3-80.12	0.9688	1.1050	1.7096	1.0118*	**0.9632**	0.9687	0.9716	0.9689
80.3-90.12	0.9534	0.9854	1.0201	**0.8966**	0.9516	0.9585	0.9577	0.9524
90.3-00.12	**0.8951**	0.9268*	1.309*	0.9262*	0.9163	0.9139	0.9026	0.8971
70.3-90.12	0.9754	1.0553	1.3845	0.9758	**0.9711**	0.9777	0.9789	0.9748
70.3-00.12	0.9632	1.0334*	1.3706	0.9755	0.9638	0.9686	0.9676	**0.9631**
80.3-00.12	0.9395	0.9691	1.0921	**0.9049**	0.9450	0.9488	0.9450	0.9393
70.3-03.12	**0.9535**	1.0217	1.3471	0.9628	0.9569	0.9590	0.9580	0.9535

$Note$: The table shows the RMSE of PC, PLS and SPCA over the benchmark model for 1-step ahead forecast. For each approach of PLS, the results are only shown for the number of components k ($k = 1$ or 2) which gives the best RMSE. An asterisk means $k = 2$ and those bold entries are the best RMSE for each forecast subsample.

Table 4-7 give the RMSE for all the approaches I applied for the seven subsamples and h =1, 6, 12 and 24. It is obvious that except the first dynamic PLS approach (DA_1), all the other seven approaches outperform the simple AR(4) model in most (192 out of 196) of the cases. for h = 1 PC, DA_2 and SPCA($\delta = 0.50$) yields respectively two of the 7 best RMSE and SPCA($\delta = 0.95$) gives one best result. For h = 6, 12, and 24, 76.2% (16 out of 21) best results are obtained from SA, while DA_2 yields three smallest RMSE for h = 6 and 1 for h = 12, and one best result is from PC for h = 24. Totally, SA alone brings 57% best results. Forecasting performance generally gets better when the forecast horizon

Table 5: RMSE, $h = 6$

		PLS			SPCA			
period	PC	SA	DA_1	DA_2	$\delta = 0.50$	$\delta = 0.65$	$\delta = 0.80$	$\delta = 0.95$
70.3-80.12	0.7117	0.7047	1.8324	**0.6532**	0.6723	0.6726	0.6724	0.6713
80.3-90.12	0.6867	0.6353	1.5013	**0.6319**	0.6848	0.6947	0.6900	0.6922
90.3-00.12	0.7195	**0.6252**	1.9068	0.6751	0.6462	0.6518	0.6528	0.6606
70.3-90.12	0.7234	**0.7007**	1.7358	0.6749	0.7014	0.7070	0.7046	0.7059
70.3-00.12	0.7139	**0.6821**	1.7431	0.6655	0.6881	0.6936	0.6915	0.6941
80.3-00.12	0.6786	**0.6184**	1.5533	0.6252	0.6675	0.6764	0.6726	0.6767
70.3-03.12	0.7208	0.6795	1.7458	**0.6681**	0.6842	0.6896	0.6880	0.6948

$Note$: The table shows the RMSE of PC, PLS and SPCA over the benchmark model for 6-step ahead forecast. For each approach of PLS, the results are only shown for the number of components k ($k = 1$ or 2) which gives the best RMSE. An asterisk means $k = 2$ and those bold entries are the best RMSE for each forecast subsample.

Table 6: RMSE, $h = 12$

		PLS			SPCA			
period	PC	SA	DA_1	DA_2	$\delta = 0.50$	$\delta = 0.65$	$\delta = 0.80$	$\delta = 0.95$
70.3-80.12	0.7320	**0.6728**	1.7135	0.6982	0.6752	0.6738	0.6770	0.6809
80.3-90.12	0.6187	0.5887	1.4561*	**0.5881**	0.6406	0.6414	0.6410	0.6425
90.3-00.12	0.7941	**0.6572***	1.9807	0.6794	0.7069	0.7137	0.7146	0.7185
70.3-90.12	0.6683	**0.6193**	1.6848	0.6312	0.6472	0.6467	0.6476	0.6503
70.3-00.12	0.6745	**0.6175**	1.7205	0.6289	0.6488	0.6488	0.6497	0.6523
80.3-00.12	0.6378	**0.5907**	1.7017	0.5912	0.6445	0.6461	0.6458	0.6474
70.3-03.12	0.6764	**0.6191**	1.7454	0.6286	0.6524	0.6531	0.6544	0.6570

$Note$: The table shows the RMSE of PC, PLS and SPCA over the benchmark model for 12-step ahead forecast. For each approach of PLS, the results are only shown for the number of components k ($k = 1$ or 2) which gives the best RMSE. An asterisk means $k = 2$ and those bold entries are the best RMSE for each forecast subsample.

Table 7: RMSE, $h = 24$

		PLS			SPCA			
period	PC	SA	DA_1	DA_2	$\delta = 0.50$	$\delta = 0.65$	$\delta = 0.80$	$\delta = 0.95$
70.3-80.12	0.7212	**0.5274**	1.2923*	0.5510	0.5450	0.5496	0.5671	0.5817
80.3-90.12	**0.5209**	0.5451	1.1678*	0.5464	0.6114	0.6212	0.6527	0.6494
90.3-00.12	0.7192	**0.5993**	1.8272*	0.6001	0.8125	0.7621	0.7418	0.7373
70.3-90.12	0.6335	**0.5387**	1.2516*	0.5536	0.5864	0.5948	0.6185	0.6244
70.3-00.12	0.6409	**0.5436**	1.2955*	0.5577	0.6054	0.6089	0.6290	0.6338
80.3-00.12	0.5496	**0.5530**	1.2577*	0.5547	0.6421	0.6428	0.6667	0.6629
70.3-03.12	0.6456	**0.5477**	1.3403*	0.5624	0.6067	0.6105	0.6303	0.6356

$Note$: The table shows the RMSE of PC, PLS and SPCA over the benchmark model for 24-step ahead forecast. For each approach of PLS, the results are only shown for the number of components k ($k = 1$ or 2) which gives the best RMSE. An asterisk means $k = 2$ and those bold entries are the best RMSE for each forecast subsample.

increases. For PLS approaches, only 3 best RMSE are obtained from k = 2 for SA, and 2 from DA_2. However, for DA_1 9 out of 28 best RMSE are from k = 2.

Concerning the PLS approaches, it is crucial to address and easy to notice that the first dynamic approach (DA_1), which includes the lags of the target variable in the dataset, always underperforms the AR(4) model. The best RMSE result for DA_1 is 1.0201 and the worst is almost as double (1.9807). The reason for its bad performance is that as relatively important independent variables, the lags of the respond are given their weights by DA_1 in a different way than the others since DA_1 does not include them in the forecasting model directly. On the country, the other two PLS approaches which capture the dynamics of the target variable by including the lags of target variables directly in the forecasting model provide much better results. They together give 79% of the best RMSE.

With regard to the SPCA approaches, all of them outperform PC in at least half of the 28 cases. When δ = 0.50, SPCA performs best in terms of the comparison to PC, which gives 19 out of 28 better results than PC. In general, PC is more precise for 1-step ahead forecasts and SPCA is preferred for h = 6, 12,and 24. Among the four SPCA approaches, 20 out of 28 smallest RMSE lie on the case where δ = 0.50. It is important to note that with δ = 0.50, the average adjusted total variances explained by the first and second SPC are only 47.61% and 48.67% of the total variances explained by the first two ordinary PC. And the average numbers of variables with non-zero weights are only 46.28 (35.07% of the 132 variables) and 20.78 (15.74% of the 132 variables) respectively for the two sparse PC's.

Fuentes, Poncela, and Rodríguez (2015) also make use of PCA and the same approaches of PLS on the those subsamples, but different RMSE are obtained by them. One possible reason is that they use different way to calculate the factor and its lags in Equation (18). For example, at each time period t I compute the factor $\hat{F}_t$ as well as its lags $\hat{F}_{t-1}$,..., $\hat{F}_1$ all using the weights calculated at time t, namely the factor and its lags are updated every time period, but Fuentes, Poncela, and Rodríguez (2015) might only update $\hat{F}_t$ and keep the lags unchanged. Although we get different values of RMSE, similar patterns are present concerning PCA and the three approaches of PLS. Fuentes, Poncela, and Rodríguez (2015) also find that DA_1 has the worst performance in all cases and its RMSE is as more than twice of the values of SA for $h = 6, 12$ and 24. Another similar pattern is that they also find that SA has the most accurate forecasting performance among the four methods, and DA_2 is the second best, followed by PC.

3.5 Results of the Diebold and Mariano test

Table 8 and 9 show the test statistics of the Diebold and Mariano test under the null hypothesis of equal forecast accuracy between SPCA and each of the other four approaches. These tables indicate for each δ, at most four and at least two entries are greater than 1.96 among all cases. This suggests that there is no significant difference in forecast accuracy between SPCA and the other approaches, except for DA_1. Due to the symmetry of the applied error function $g(\cdot)$, the large negative test statistics for DA_1 suggests that its forecast accuracy is worse than that of SPCA, which has already been seen from the results of RMSE. In general, the predictive power of SPCA is not worth than the other approaches. Moreover, I also find that different δ does not make many differences to SPCA's forecast accuracy, according to the Diebold and Mariano test.

Table 8: Diebold and Mariano test for δ = 0.95 and 0.80

	δ = 0.95				δ = 0.80			
period	PCA	SA	DA1	DA2	PCA	SA	DA1	DA2
h=1								
70.3-80.12	0.1824	−2.0422	−2.0061	−0.6686	0.7650	−2.0224	−1.9998	−0.6293
80.3-90.12	−1.0639	−0.7075	−0.4688	**2.1670****	1.0091	−0.6020	−0.4311	**2.3313****
90.3-00.12	**1.6595***	−0.7259	−3.0470	−0.9902	1.5730	−0.5925	−3.0123	−0.8558
70.3-90.12	−1.0082	−1.9833	−1.9358	−0.0251	1.1749	−1.9008	−1.9167	0.0721
70.3-00.12	−0.3040	−1.8127	−2.2741	−0.4724	1.6395	−1.7146	−2.2492	−0.2994
80.3-00.12	−0.2880	−0.8348	−1.3446	1.6446	1.5803	−0.6860	−1.2921	**1.9049***
70.3-03.12	0.0071	−2.1691	−2.4286	−0.3868	**1.8327***	−2.0496	−2.4015	−0.1991
h=6								
70.3-80.12	−1.1246	−0.6203	−2.2673	0.3732	−1.0862	−0.5941	−2.2666	0.3881
80.3-90.12	0.1161	0.9706	−2.3119	1.4552	0.0701	0.9470	−2.3304	1.4275
90.3-00.12	−1.2466	1.3026	−2.3379	−0.5431	−1.3699	1.2885	−2.3659	−0.8588
70.3-90.12	−0.5113	0.1212	−3.0832	0.9779	−0.5571	0.0925	−3.0942	0.9457
70.3-00.12	−0.6418	0.3161	−3.4548	1.0158	−0.7355	0.2504	−3.4717	0.9295
80.3-00.12	−0.0477	1.1984	−2.8611	1.4942	−0.1547	1.1330	−2.8890	1.3978
70.3-03.12	−0.9019	0.4140	−3.6698	1.0141	−1.1173	0.2379	−3.7011	0.7689
h=12								
70.3-80.12	−0.7126	0.3129	−1.9385	−0.4148	−0.7683	0.1732	−1.9349	−0.5176
80.3-90.12	0.4336	1.0265	−1.7114	1.2393	0.4001	0.9868	−1.7167	1.1846
90.3-00.12	−0.8390	0.5541	−1.6638	1.4349	−0.8585	0.5161	−1.6628	1.2636
70.3-90.12	−0.4008	1.0195	−2.5422	0.6164	−0.4573	0.9328	−2.5449	0.5289
70.3-00.12	−0.5287	1.2593	−2.8188	0.8361	−0.5865	1.1669	−2.8210	0.7420
80.3-00.12	0.1955	1.2879	−2.0272	1.5225	0.1610	1.2388	−2.0320	1.4560
70.3-03.12	−0.4801	1.4225	−2.9785	1.0314	−0.5408	1.3276	−2.9808	0.9361
h=24								
70.3-80.12	−0.8146	0.6133	−1.4945	0.7644	−0.8764	0.4409	−1.4856	0.4360
80.3-90.12	1.0041	0.9489	−1.5602	1.0508	1.0889	1.0071	−1.5592	1.1129
90.3-00.12	0.1552	1.3143	−1.2002	1.4349	0.1870	1.2697	−1.1922	1.3812
70.3-90.12	−0.0810	1.1613	−2.0896	1.2669	−0.1342	1.0947	−2.0760	1.2046
70.3-00.12	−0.0677	1.3136	−2.3046	1.4584	−0.1151	1.2565	−2.2869	1.4125
80.3-00.12	1.0126	1.1427	−1.8838	1.2600	1.1044	1.2140	−1.8795	1.3356
70.3-03.12	−0.0991	1.3291	−2.4847	1.4551	−0.1527	1.2628	−2.4668	1.3962

Note: The table shows the Diebold and Mariano test statistics. H_0 is rejected at the 10% level with a test statistic greater than 1.65 and at the 5% level with a test statistic greater than 1.96 . Bold entries show the test statistics that are larger than 1.96. * means H_0 is rejected at the 10% level and ** means H_0 is rejected at the 5% level.

3.6 Results of the Pesaran-Timmermann test

The Pesaran-Timmermann test is performed and the test results are shown in Table 10. Table 10 indicates that except DA_1, the null hypothesis of no predictive power is rejected at 5% level for all other approaches with all subsamples. At the 1% significance level, the null hypothesis is rejected for all subsamples with DA_2 approach and is rejected for all but one case with each of the other approaches, except DA_1. Thus, PCA, SA, DA_2 and the four approaches of SPCA are all able to predict the direction of change in almost all cases, under small significance level.

3.7 Forecasting performance when $p > n$

The general SPCA algorithm developed by Zou, Hastie, and Tibshirani (2006) is efficient to deal with datasets where the number of observations n is larger than the number of variables p, but it is computationally expensive for the case

where $p \gg n$. In the latter situation, Zou, Hastie, and Tibshirani (2006) find that the computation can be boosted when $\lambda \to \infty$ in Equation (14), which results in a soft-thresholding problem instead of the *elastic net* problem. To investigate the forecasting performance of SPCA using soft-thresholding, the subsamples in Table 11 are used. The computation is much faster using soft-thresholding and is done within half an hour. Note that I do not have a dataset where $p \gg n$, but all subsamples in Table 11 satisfy that $p > n$. Note also that because of the proven poor forecasting performance of the first dynamic PLS approach (DA_1) and poor RMSE for PCA for the new subsamples, the forecasting results of them are not present.

The resulting RMSE are presented in Table 12, from where we can see that SPCA with soft-thresholding outperforms the alternative methods in 14 out of 20 cases. Comparing to the results in Table 4 to 7 where 3 out of 28 smallest RMSE are obtained from SPCA, SPCA seems to be more appropriate in the cases where $p > n$ than in the other case.

Table 9: Diebold and Mariano test for δ = 0.65 and 0.50

	δ = 0.65				δ = 0.50			
period	PCA	SA	DA1	DA2	PCA	SA	DA1	DA2
h=1								
70.3-80.12	−0.0353	−2.0863	−2.0080	−0.6758	−0.8380	−2.1970	−2.0225	−0.7590
80.3-90.12	0.6489	−0.6061	−0.4250	**2.4784****	−0.1951	−0.7727	−0.4729	**2.2703****
90.3-00.12	1.4255	−0.3289	−2.9158	−0.4545	**1.6868***	−0.2507	−2.9183	−0.3981
70.3-90.12	0.4718	−1.9663	−1.9223	0.0440	−0.7042	−2.1584	−1.9538	−0.1144
70.3-00.12	1.1223	−1.7225	−2.2431	−0.2637	0.1071	−1.8725	−2.2704	−0.4497
80.3-00.12	1.3247	−0.5999	−1.2558	**2.1840****	0.6987	−0.7201	−1.2926	**2.0751****
70.3-03.12	1.2438	−2.0593	−2.3952	−0.1610	0.6347	−2.1387	−2.4088	−0.2490
h=6								
70.3-80.12	−1.0419	−0.5792	−2.2646	0.3808	−1.0377	−0.5911	−2.2643	0.3766
80.3-90.12	0.1740	1.0439	−2.3419	1.5088	−0.0409	0.8534	−2.3394	1.2861
90.3-00.12	−1.3665	1.2290	−2.3651	−0.8656	−1.4101	0.9115	−2.3685	−0.9836
70.3-90.12	−0.4817	0.1507	−3.0983	0.9976	−0.6309	0.0175	−3.0987	0.8260
70.3-00.12	−0.6616	0.3083	−3.4768	0.9793	−0.8197	0.1597	−3.4769	0.7898
80.3-00.12	−0.0562	1.2299	−2.9040	1.4752	−0.2790	1.0205	−2.8983	1.2308
70.3-03.12	−1.0518	0.2861	−3.7080	0.8109	−1.2042	0.1311	−3.7089	0.6072
h=12								
70.3-80.12	−0.8262	0.0454	−1.9343	−0.6130	−0.7963	0.0962	−1.9348	−0.5615
80.3-90.12	0.3925	0.9623	−1.7201	1.1431	0.3612	0.9055	−1.7250	1.0664
90.3-00.12	−0.8571	0.5056	−1.6618	1.2053	−0.8959	0.4396	−1.6634	0.9638
70.3-90.12	−0.4750	0.8866	−2.5470	0.4905	−0.4499	0.8589	−2.5503	0.4847
70.3-00.12	−0.6030	1.1164	−2.8230	0.7000	−0.5864	1.0606	−2.8273	0.6679
80.3-00.12	0.1611	1.2058	−2.0355	1.4029	0.1246	1.1205	−2.0416	1.2927
70.3-03.12	−0.5698	1.2636	−2.9837	0.8806	−0.5712	1.1839	−2.9893	0.8243
h=24								
70.3-80.12	−0.9456	0.2394	−1.4790	−0.0391	−0.9570	0.1880	−1.4714	−0.1501
80.3-90.12	1.4046	1.1554	−1.4954	1.3076	0.0000	1.1823	−1.3995	1.0890
90.3-00.12	0.3320	1.2372	−1.1687	1.3275	0.6759	1.272	−1.1202	1.3334
70.3-90.12	−0.3804	0.9527	−2.0441	1.1524	−0.4969	0.8395	−1.993	0.8702
70.3-00.12	−0.3384	1.1790	−2.2428	1.4871	−0.4018	1.1428	−2.1753	1.2892
80.3-00.12	1.4445	1.5002	−1.7962	**1.6814***	0.0000	**1.6514***	−1.6743	1.5511
70.3-03.12	−0.3854	1.1768	−2.4188	1.4466	−0.4558	1.1344	−2.3495	1.2426

Note: The table shows the Diebold and Mariano test statistics.H_0 is rejected at the 10% level with a test statistic greater than 1.65 and at the 5% level with a test statistic greater than 1.96. Bold entries show the test statistics that are larger than 1.65. $*$ means H_0 is rejected at the 10% level and $**$ means H_0 is rejected at the 5% level.

Table 10: The Pesaran and Timmermann test

		PLS			SPCA			
Period	PCA	SA	DA_1	DA_2	$\delta = 0.95$	$\delta = 0.80$	$\delta = 0.65$	$\delta = 0.50$
$h = 1$								
70.3-80.12	**2.2500***	2.5151	**−2.3168***	2.6731	**2.2500***	**2.2500***	**2.2500***	**2.2500***
80.3-90.12	3.3423	**2.2044***	**0.8388****	3.5668	3.3423	3.2045	3.0671	2.9802
90.3-00.12	3.7261	3.0042	**1.7409***	2.9243	3.7261	3.7261	4.2571	3.9479
70.3-90.12	3.9157	3.2235	**−0.9007****	3.4365	3.9157	3.8197	3.7237	3.6529
70.3-00.12	4.9995	3.8187	**−0.6756****	4.9747	4.9995	4.9194	5.166	4.9194
80.3-00.12	4.5702	3.8118	**0.6831****	4.3099	4.5702	4.4704	4.7558	4.4704
70.3-03.12	5.3290	4.6063	**−0.0287****	5.1402	5.3290	5.2502	5.4702	5.1249
$h = 6$								
70.3-80.12	5.2913	4.9389	**0.0371****	4.9389	5.1090	5.1090	4.9280	4.7481
80.3-90.12	6.5834	5.7939	2.8760	6.3581	6.6875	6.6875	6.5011	6.6875
90.3-00.12	6.8708	6.8657	5.1225	6.6900	6.8632	6.5107	6.3326	6.3326
70.3-90.12	7.8980	7.3377	2.1609	7.6072	7.8462	7.8462	7.7122	7.7122
70.3-00.12	10.4240	9.9667	4.7983	10.0896	10.3931	10.1845	9.9708	9.9708
80.3-00.12	9.5189	8.9834	5.6732	9.2473	9.6209	9.3669	9.1106	9.2385
70.3-03.12	11.0610	10.7883	5.4281	10.8168	10.9121	10.7127	10.5052	10.6012
$h = 12$								
70.3-80.12	5.7929	6.1258	**−0.0452****	5.8498	5.8498	5.8498	5.8498	6.0596
80.3-90.12	4.9630	4.4553	2.4656	4.8117	4.9630	4.9630	4.9630	4.9630
90.3-00.12	6.5544	7.9186	5.0827	7.3849	6.6855	6.5048	6.5048	6.6855
70.3-90.12	7.4708	7.0946	**0.5673****	7.2170	7.3449	7.3449	7.3449	7.4731
70.3-00.12	9.7837	9.8843	3.3415	10.0933	9.7803	9.6759	9.6759	9.8850
80.3-00.12	7.9805	8.1961	3.2693	8.5659	8.1520	8.0353	8.0353	8.1520
70.3-03.12	10.4290	10.6252	4.2968	11.0234	10.6261	10.5260	10.5260	10.7264
$h = 24$								
70.3-80.12	7.8487	7.9137	3.8915	6.6050	6.2673	6.2673	6.2673	6.2673
80.3-90.12	4.9731	5.3993	3.6065	5.0448	4.7298	4.8872	5.0448	5.0853
90.3-00.12	5.8151	7.4252	3.1662	7.4252	7.3048	7.1394	6.9740	6.4476
70.3-90.12	8.5716	9.0893	5.0687	8.0592	7.6753	7.8033	7.9313	7.9334
70.3-00.12	9.9684	11.4345	5.7765	10.6050	10.2184	10.2184	10.2184	9.9115
80.3-00.12	7.0385	8.7435	4.4009	8.4884	8.1908	8.1908	8.1908	7.8456
70.3-03.12	10.7200	12.2245	6.5565	11.3368	10.9799	11.0856	11.0856	10.7962

$Note$: The table shows the Pesaran and Timmermann test statistics. H_0 is rejected at the 5% level with a test statistic greater than 1.65 and at the 1% level with a test statistic greater than 2.33. Bold entries show the test statistics that are smaller than 2.33. $*$ means H_0 is **not** rejected at the 1% level and $**$ means H_0 is **not** rejected at the 5% level.

Table 11: Estimation and forecast subsamples, $p > n$

SS	Estimation subsample	Forecast subsample
M1	1960:03 to 1966:03$-h$	1966:03 to 1970:03
M2	1970:03 to 1766:03$-h$	1976:03 to 1980:03
M3	1980:03 to 1986:03$-h$	1986:03 to 1990:03
M4	1990:03 to 1996:03$-h$	1996:03 to 2000:03
M5	1993:03 to 1998:03$-h$	1993:03 to 2003:12

To see that the good performance of SPCA in Table 12 is not because that using soft-thresholding is generally better for both situations where $p > n$ and $p < n$, I apply SPCA with soft-thresholding with $\delta = 0.50$ for the seven subsamples in Table 1. The resulting RMSE are presented in Table 13, showing that obviously for those subsamples where $p < n$, SPCA with soft-thresholding gives much worse RMSE than setting λ to a small positive number.

Table 12: RMSE for subsamples where $p > n$

	SA	DA_2	$SPCA_{ST}$		SA	DA_2	$SPCA_{ST}$
$h = 1$				$h = 6$			
M1	1.0131	1.1105	**0.8707**	M1	1.0462	0.9703	**0.9025**
M2	1.1983	1.2368	**1.0000**	M2	1.2121	1.1342	**1.0130**
M3	1.0795	1.0294	**0.9573**	M3	**0.9234**	0.9613	0.9806
M4	1.1631	1.1040	**1.0567**	M4	0.8430	0.9445	**0.8207**
M5	1.0343	**1.0016**	1.1068	M5	0.8626	1.1705	**0.8275**
$h = 12$				$h = 24$			
M1	1.1465	1.0426	**0.8681**	M1	1.3308	1.0257	**0.9514**
M2	0.9461	**0.8005**	1.1030	M2	0.3948	**0.3720**	1.0614
M3	**0.8709**	0.8840	0.9795	M3	**0.3949**	0.9345	1.0498
M4	1.0483	1.1444	**0.9170**	M4	0.9152	0.8225	**0.7916**
M5	1.1828	1.4341	**1.0458**	M5	1.2403	2.2822	**0.9221**

Note: The table shows the RMSE for the subsamples where $p > n$, of the static and the second dynamic PLS, and SPCA using soft-thresholding ($SPCA_{ST}$) with $\delta = 0.50$. Bold entries are the best RMSE for the subsamples.

Table 13: RMSE for SPCA using soft-thresholding, $\delta = 0.50$.

	$SPCA_{ST}$			
Period	$h = 1$	$h = 6$	$h = 12$	$h = 24$
70.3-80.12	1.0161	1.0209	1.0265	0.9959
80.3-90.12	1.0038	0.8821	0.9726	0.9905
90.3-00.12	0.9486	0.7710	0.8250	0.8790
70.3-90.12	1.0094	0.9432	0.9985	0.9929
70.3-00.12	0.9995	0.9256	0.9846	0.9838
80.3-00.12	0.9894	0.8658	0.9527	0.9745
70.3-03.12	0.9927	0.9111	0.9728	0.9762

4 Discussion and Conclusion

In this paper, I examine the performance of the sparse principal component analysis for forecasting U.S inflation, and compare it with the performances of principal component analysis and three different approaches of partial least squares. Empirical results show that for estimation subsamples where the number of observations is larger than the number of variables, the static and the second dynamic approach of the partial least squares are the two best forecast methods for the subsamples with $n > p$ in terms of RMSE. The failure of the first PLS dynamic approach suggests that in forecasting the U.S inflation, the lags of the predicting variable is important and should be included directly in the forecast model.

The sparse principal component analysis generally performs better than the ordinary principal component analysis. Especially, when adjusted total variances explained by the sparse components are less than half of those explained by the ordinary principal components, the predicting accuracy is still better than PCA. This indiactes that the factors extracted by PCA do contain useless information for forecasting and variable selection is of importance. The Diebold and Mariano test (Diebold and Mariano (1995)) indicates that the forecasting accuracy of SPCA is as good as the others and the Pesaran and Timmermann test (Pesaran and Timmermann (1992)) show that SPCA can predict the direction correctly under small significance levels. SPCA with soft-thresholding performs exceptionally well for the subsamples with $p > n$ and the computational efficiency of soft-thresholding has been seen.

Although encouraging results concerning SPCA are found, some limitations have restricted my investigation for it. In practice, my time limitation does not allow a complete and careful grid search of the choices of the *lasso* penalty coefficients $\lambda_{1,j}$, due to the expensive computation of SPCA. Instead of a grid search, one can also employ a cross-validation. Additionally, the limited number of variables in the dataset does not allow me to make further research for the performance of SPCA in the case where $p \gg n$.

All in all, based on my research it is reasonable to say that with respect to forecasting the U.S inflation, sparse principal component analysis performs better than the ordinary principal component analysis, while it also enjoys the proven advantage of interpretability for its components.

References

Bai, Jushan and Serena Ng (2008). "Forecasting economic time series using targeted predictors". In: *Journal of Econometrics* 146.2, 304–317. ISSN: 03044076. DOI: 10.1016/j.jeconom.2008.08.010.

Cadima, Jorge and Ian T. Jolliffe (1995). "Loading and correlations in the interpretation of principle compenents". In: *Journal of Applied Statistics* 22.2, 203–214. ISSN: 0266-4763, 1360-0532. DOI: 10.1080/757584614.

Diebold, FX and R Mariano (1995). "Comparing predictive accuracy". In: *Journal of Business and Economic Statistics* 13, 253–265.

Fuentes, Julieta, Pilar Poncela, and Julio Rodríguez (2015). "Sparse Partial Least Squares in Time Series for Macroeconomic Forecasting". In: *Journal of Applied Econometrics* 30.4, 576–595. ISSN: 08837252. DOI: 10.1002/jae.2384.

Groen, Jan J.J. and George Kapetanios (2016). "Revisiting useful approaches to data-rich macroeconomic forecasting". In: *Computational Statistics & Data Analysis* 100, 221–239. ISSN: 01679473. DOI: 10.1016/j.csda.2015.11.014.

Kelly, Bryan and Seth Pruitt (2015). "The three-pass regression filter: A new approach to forecasting using many predictors". In: *Journal of Econometrics* 186.2, 294–316. ISSN: 03044076. DOI: 10.1016/j.jeconom.2015.02.011.

Pesaran, M. Hashem and Allan Timmermann (1992). "A Simple Nonparametric Test of Predictive Performance". In: *Journal of Business and Economic Statistics* 10.1, pp. 461–465.

Sjöstrand, Karl et al. (2018). "SpaSM: A MATLAB Toolbox for Sparse Statistical Modeling". In: *Journal of Statistical Software* 84.10. ISSN: 1548-7660. DOI: 10.18637/jss.v084.i10. URL: http://www.jstatsoft.org/v84/i10/.

Stock, James and Mark Watson (2002a). "Forecasting Using Principal Components From a Large Number of Predictors". In: *Journal of the American Statistical Association* 97.460, 1167–1179. ISSN: 0162-1459, 1537-274X. DOI: 10.1198/016214502388618960.

— (2002b). "Macroeconomic Forecasting Using Diffusion Indexes". In: *Journal of Business & Economic Statistics* 20.2, 147–162. ISSN: 0735-0015, 1537-2707. DOI: 10.1198/073500102317351921.

— (2005). *Implications of Dynamic Factor Models for VAR Analysis*. w11467, w11467. DOI: 10.3386/w11467. URL: http://www.nber.org/papers/w11467.pdf.

Tibshirani, Robert (1996). "Regression Shrinkage and Selection Via the Lasso". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 58.1, 267–288. ISSN: 00359246. DOI: 10.1111/j.2517-6161.1996.tb02080.x.

Wold, Herman (1966). "Estimation of principal components and related models by iterative least squares". In: *Multivariate Analysis*. Ed. by Krishnaiaah PR, Academic Press: New York; 391–420.

— (1975). "Path models with latent variables: the nonlinear iterative partial least squares (NIPALS) approach". In: *Quantitative Sociology: Intentional Perspective on Mathematical and Statistical Modeling*, pp. 307–357.

Zou, Hui and Trevor Hastie (2005). "Regularization and variable selection via the elastic net". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67.2, 301–320. ISSN: 1369-7412, 1467-9868. DOI: 10.1111/j.1467-9868.2005.00503.x.

Zou, Hui, Trevor Hastie, and Robert Tibshirani (2006). "Sparse Principal Component Analysis". In: *Journal of Computational and Graphical Statistics* 15.2, 265–286. ISSN: 1061-8600, 1537-2715. DOI: 10.1198/106186006X113430.

Appendix

A MATLAB code

The programming part for this thesis is done by MATLAB and the MATLAB code used are in the file "MATLAB code_Ruoyu Zhi 424107", which contains four files "large subsample", "small subsample" "tests" and "SpaSM—-Sjöstrand, Karl et al. (2018)". Here I provide an explanation of each program in the files.

File "large subsample"

Programs in the file "large subsample" are for computations of AR, PCA, PLS and SPCA for the subsamples where $n > p$.

AR_4: forecast using the benchmark AR model.
get_index: gets indices of starting and ending period of the forecast subsample in the dataset.
$main$: main program for all computations.
$nipals$: PLS algorithm proposed by Wold (1975).
$PCA\_factor\_estimation\_k$: computes principal components for different k.
$PCA\_forecast$: computes MSE of forecasting results of PCA.
$PCA\_k\_selection$: selects the best number of principal components included according to BIC.
$PCA\_lag\_estimation$: selects the best numbers of lags to include in PCA according to BIC.
$PLS\_ARp\_residual$: calculates AR coefficients for different number of lags.
$PLS\_forecasting$: computes MSE of forecasting of PLS.
$PLS\_lag\_estimation$: selects the best lags of the forecasting variable and PLS components according to BIC for the three PLS approaches.
$PLS\_XY\_obtaining$: computes X and Y for the three different PLS approaches.
$SPCA\_factor$: computes sparse principal components.
$SPCA\_forecast$: computes MSE of forecasting results of SPCA.
$SPCA\_k\_selection$: applies Algorithm 1 and computes estimates the SPCA forecast model.

$SPCA\_lag\_estimation$: selects the best lags of sparse components and forecasting variable according to BIC.

Programs in the file "small subsample" are for computations of AR, PCA, PLS and SPCA for the subsamples where $p > n$. Each program with "_short_period" does the same thing to the programs in "large subsample" without "_short_period", but is designed for the cases where $p > n$.

File "small subsample"

Programs in the file "Tests", performs the Diebold and Mariano test and the Pesaran-Timmermann test.

DM_test: performs the Diebold and Mariano test.
PT_test: performs the Pesaran-Timmermann test.

File "SpaSM—-Sjöstrand, Karl et al. (2018)"

Programs in the file "SpaSM—-Sjöstrand, Karl et al. (2018)" are programs implemented by Sjöstrand et al. (2018) and are used in the computations for SPCA. For details please refre to Sjöstrand et al. (2018).