Erasmus University Rotterdam
Erasmus School of Economics
Bachelor Thesis Econometrie en Operationele Research

Random Forest combined with scaled PCA: A proper enhancement of a computationally expensive ML method

Kai Geenen (544074)

Abstract

In the last decade, the advent of big data has given investors of the stock market access to much broader data sets, possibly leading to the usage of many predictors while constructing forecasts. This often results in complex and computationally expensive models that use data typically characterized by a high degree of noise. Our research investigates if we can solve this issue by combining the RF method with PCA or sPCA to properly reduce the dimension of a data set, leading to a less complex and computationally expensive ML model while maintaining forecasting accuracy. We use the FRED-MD database proposed by McCracken and Ng (2016) to forecast the monthly simple returns and volatility of the S&P-500 and Nasdaq-100. We construct forecasts of these stock indices through PCR, sPCR, RF, PCA-RF, and sPCA-RF. Our results show that the forecasts made through PCA-RF and sPCA-RF do not significantly outperform each other. However, they consistently outperform the forecasts of PCR, sPCR, and RF on a 1% significance level. Additionally, we find that when RF is augmented with PCA and sPCA, there is not only a reduction in RF's complexity and computational expensiveness but also an improvement in performance, particularly during crises, indicating the effectiveness of these methods in periods of high market volatility.

Supervisor:	Opschoor, PA
Second assessor:	Zhelonkin, M
Date final version:	16th July 2023

1 Introduction

Hedge funds, investors, and financial analysts all have a common interest: Constructing accurate stock market forecasts. Accurately predicting the market provides the opportunity to make tremendous profits and prevent massive losses. With the rise of big data, these parties have access to substantial databases containing viable input in forecasting models, leading to complex and computationally expensive models. However, not all data is useful; in some cases, extra data could worsen forecasting performance (Boivin and Ng, 2006; Heij et al., 2004). Consequently, it is vital to look at techniques that can remove noisy, uninformative data from large data sets, leading to less complex and computationally expensive models. This research addresses this by integrating RF with both sPCA and PCA to forecast simple returns and volatility of the S&P-500 and Nasdaq-100 using macroeconomic data. Thus, our central research question is:

How does the sPCA-RF method perform while forecasting stock indices, compared to PCA-RF and RF?

Various results of the current literature regarding stock indices forecasting led us to our main research goal. Firstly, even though PCA is a widely used dimension reduction technique, it has a main drawback which is that when factors are strong, it fails to distinguish target-relevant and irrelevant latent factors, which in turn means when reducing k amount of predictors to g predictors, it does not mean that these g predictors can best forecast the target (Huang et al., 2022). Furthermore, in a weak factor case, PCA could fail to extract signals from a big noisy data set, resulting in biased forecasts even when using all factors (Huang et al., 2022). Consequently, Huang et al. (2022) present a new approach to dimension reduction related to PCA, namely, scaled PCA (sPCA), which addresses the problems PCA faces in weak and strong factor cases, which makes sPCA interesting for our research since our macroeconomic data could contain much noise. More specifically, we use macroeconomic data of the big FRED-MD database proposed by McCracken and Ng (2016). Forecasting stock returns and volatility through macroeconomic variables has great support within the literature (Chen, 2009; Maysami et al., 2004; Pilinkus, 2009). Hence, the usage of FRED-MD in our research is warranted.

Furthermore, Huang et al. (2022) investigate the performance of sPCA while constructing one-month-ahead forecasts for four target variables: Inflation, industrial production growth, unemployment rate, and the S&P 500 index volatility. Of these four applications, this research zooms in and extends the last application by forecasting simple returns and volatility of the S&P-500 and Nasdaq-100 through the same macroeconomic data of FRED-MD. Furthermore, Huang et al. (2022) constructs forecasts through PCR and scaled PCR, however another type of method that has recently grown in popularity due to advances in technology and big data is machine learning. There is extensive research on which ML methods are the most accurate when forecasting stock markets. For instance, Rossi (2018) describes the good performances of ML methods in forecasting stock returns and volatility at the monthly frequency compared to other econometric models. In addition, Medeiros et al. (2021) found that RF performs exceptionally well in times of high volatility compared to other ML models. The result from Medeiros

et al. (2021) is interesting since some studies show promising results when RF is augmented with PCA (PCA-RF). For example, Waqar et al. (2017) find that ML techniques enhance forecasting performance through PCA while predicting the stock market. In addition to the results of Waqar et al. (2017), Wang et al. (2022) forecasts wind power through various ML methods, one of which is PCA-RF, and their results show that RF greatly benefits from the dimension reduction of PCA. Consequently, Huang et al. (2022) would suggest that sPCA could be an interesting contender to augment the RF method in a financial context. Next to the promising results of RF augmentations, there also is evidence that RF without augmentations performs well when predicting the stock market (Khaidem et al., 2016; Nti et al., 2019; Yin et al., 2023). Furthermore, Nti et al. (2019) specifically shows that forecasting the stock market using macroeconomic data combined with RF gives good results. Consequently, we also look at RF without augmentations. Therefore, we construct forecasts through an expanding window using five models: PCR, sPCR, RF, PCA-RF, and sPCA-RF. These forecasts are evaluated using MSE, MAE, and, for subsample performance, CSSED. Additionally, we test for significant outperformance by means of a modified DM-test proposed by Harvey et al. (1997). Subsequently, we find the answer to our research question by investigating if there is a significant difference in forecasting performance between sPCA-RF and PCA-RF and whether sPCA and PCA enhance the forecasting performance of RF. Additionally, we examine how the nonlinear sPCA-RF and PCA-RF models perform compared to the linear PCA and sPCA models.

Our results show that the forecasts of PCA-RF and sPCA-RF do not significantly outperform each other. However, they consistently outperform PCR, sPCR, and RF on a 1% significance level while forecasting the returns and volatility of both S&P-500 and Nasdaq-100. Not only does augmenting RF with PCA and sPCA significantly improve forecasting performance, but it also greatly reduces the complexity and computational expensiveness of the model. The outperformance of (s)PCA-RF is increased during unstable economic events, such as 9/11, which indicate that PCA-RF and sPCA-RF especially perform well in times of high market volatility. Lastly, we observe that variance explained by the chosen number of principal components says little about forecasting performance, which makes variance thresholding a bad criterion for selecting principal components while forecasting stock indices. The results of our research fill gaps in the current literature because there are few studies about PCA-RF in a forecasting context (Liu and Sun, 2019; Waqar et al., 2017; Wild Ali, 2021; Ziane et al., 2021), none of which is a financial forecasting context. Moreover, there are even fewer studies about sPCA (Huang et al., 2022; Lu et al., 2022; Wei and Ouyang, 2023), and, to our knowledge, no studies about sPCA-RF. Hence, from an academic standpoint, there is still much to learn about sPCA and its merits. With our promising results on forecasting performance of sPCA-RF and PCA-RF, we build further on current literature (Chepurko et al., 2020; Cunningham, 2008; Waqar et al., 2017; Wong et al., 2016) that show that augmenting ML methods with dimension reduction techniques enhances forecasting performance, with Waqar et al. (2017) specifically showing this while predicting the stock market. Moreover, we show that the results of Wang et al. (2022) for PCA-RF, while predicting wind energy, also hold in a stock forecasting context. Next, we extend the results of Huang et al. (2022) regarding forecasting performance of sPCA and PCA. Additionally, we show

that RF performs well in times of high volatility, which is in line with Medeiros et al. (2021) and Rossi (2018), but also show that PCA-RF and sPCA-RF perform significantly better than RF, which is an extension of current literature. Lastly, our observation that variance thresholding appears to be a bad criterion for selecting principal components extends the results of Ferré (1995), who show similar results concerning variance thresholding as principal component selection criteria. However, scientific relevance is not the only motive for this research. There are also some practical applications of this research. Investors can make a more educated decision in constructing an accurate forecasting model for the S&P 500 and NASDAQ-100, which could lead to better portfolio management and more profits (Pojarliev and Polasek, 2001). Therefore, the findings of this paper could play a key role in portfolio management and investment plans that contain these stock indices.

The rest of this paper is organized as follows. Section 2 gives insight into the data, its origins, and our alterations. Next, in Section 3, we discuss our models and techniques in detail. After this, in Section 4, we show our results. Furthermore, in Section 5, we present the answer to our research question and discuss theoretical and practical implications, after which we provide suggestions for further research. Lastly, we have Appendix A, which contains a theoretical derivation with regards to the size of an out-of-bag sample, and Appendix B, which contains our replication of the results of Huang et al. (2022).

2 Data

2.1 S&P 500 and NASDAQ-100

In this research, we forecast four dependent variables connected to the S&P-500 and the NASDAQ-100. The S&P-500 and the NASDAQ-100 are stock market indices from the U.S. of which we forecast their returns and volatility. The S&P-500 is an index of the 500 largest U.S. companies, and it is a representation of the U.S. economy, as opposed to the Nasdaq-100, which is more specialized and consists primarily of tech and growth companies. We collect monthly data of the variables: *US500*, *NDX*, *VIX*, and *VXN*, using *investing.com*. The $US500$ represents S&P-500 returns, and the data runs from February 1970 through December 2019, which leads to 599 monthly observations. Next, NDX represents Nasdaq-100 returns, of which the data runs from October 1985 through December 2019, which leads to 411 observations. Furthermore, VIX represents S&P-500 volatility, and the data runs from February 1990 through December 2019, corresponding with 359 observations. Lastly, the VXN represents the Nasdaq-100 volatility, of which the data runs from November 2003 through December 2019, which leads to 194 monthly observations. The version of S&P-500 we currently know, which consists of 500 companies, was launched in 1957. On the other hand, the NASDAQ-100 was launched in 1985, which is more recent. In addition, the VIX and VXN were founded in January 1993 and January 2001, respectively. Consequently, our target variables can not have the same amount of monthly observations due to different founding dates. Furthermore, the database of *investing.com* is limited and does not contain all the observations since the founding dates of the variables. The website provides monthly simple returns and the percentual change of the stocks, which we utilize and forecast

in this research. We use an in-sample period of 120 observations corresponding to 10 years. In addition, we also use an expanding window to make our forecasts as realistic as possible; this is explained more extensively in Section 3.

In addition to *VXN*, there is a stock called the *VOLQ*, which also represents the volatility of the Nasdaq-100. However, we choose to use the *VXN*. The reason for this is that the *VOLQ* is relatively new compared to the *VXN* (founded in October 2020). Therefore the usage of *VOLQ* has less support within the literature. Alternatively, the *VXN* has a better and longer track record backed by multiple papers. For example, Corrado and Miller (2005) found that the *VXN* provides even higher quality forecasts of future volatility than the *VIX*. This is in line with Arak and Mijid (2006) and Giot (2002), which show similar results when constructing forecasts for the *VXN*.

2.1.1 Descriptive statistics of the target variables

Next, we look at the descriptive statistics of our target variables. The mean, standard deviation, maximum, minimum, and number of observations per target variable are shown in Table 1. We

Table 1: Descriptive statistics of the target variables for the whole sample period

	Mean	Std. Dev.	Maximum	Minimum	Observations
US500	0.007	0.043	0.163	-0.218	599
NDX	0.013	0.069	0.250	-0.270	411
VIX	19.159	7.364	59.890	9.510	359
VXN	21.016	7.430	60.300	11.530	194

see that NDX has a higher standard deviation than $US500$. In addition, NDX has a higher maximum and a lower minimum, indicating that the Nasdaq-100 returns are more volatile than the S&P-500 returns. The higher volatility could be explained since Nasdaq-100 comprises tech- and growth-oriented companies. These companies are characterized by higher volatility due to rapid innovation, intense competition and are more sensitive to macroeconomic factors and market sentiment. Moreover, when looking at VIX and VXN we see that the mean and standard deviation of VXN is higher than that of VIX, which is consistent with the notation that the Nasdaq-100 is more volatile than the S&P-500. The notion that some target variables are more volatile than others is important. Recent literature concerning RF shows interesting results in dealing with more volatile target variables. For example, Medeiros et al. (2021) found that RF performs exceptionally well when there is high volatility compared to other forecasting models. Therefore, it is interesting to see if we obtain similar results with these relatively volatile target variables. Lastly, it is important to look at the number of observations of our target variables. Some variables have fewer observations than others, whereas the VXN has the fewest with 194 observations. A limited number of observations can negatively influence forecasting models' accuracy and lead to inconsistent results (Mikołajczyk and Grochowski, 2018; Raudys et al., 1991). However, the results of Luan et al. (2020) show that the RF method gives decent predictions of the target variables even with a limited sample size. Consequently, due to the results of Luan et al. (2020) and the usage of an expanding window, we incorporate the *VXN*

in our research despite the limited number of observations.

2.2 Macroeconomic dataset

This research follows Huang et al. (2022) and uses the FRED-MD database, which is a database proposed by McCracken and Ng (2016), consisting of 128 macroeconomic variables which act as the explanatory variables in our forecast construction. The database consists of macroeconomic variables categorized into eight groups. The groups range from the stock market and prices to interest rates& exchange rates and more. For a full description of all the macroeconomic variables and their corresponding groups, we refer to McCracken and Ng (2016). Fred-MD contains data from January 1959 up until December 2022. We use data from this database between February 1970 through December 2019, depending on the specific target variable we forecast. Forecasting stock index volatility with macroeconomic variables has support within the literature (Chen, 2009; Maysami et al., 2004; Pilinkus, 2009). Therefore following Huang et al. (2022) and using the FRED-MD database for our data is warranted.

2.2.1 Alterations of the dataset

The FRED-MD database originally consists of 128 macroeconomic variables, but the time series of 5 variables are either missing or incomplete. The variables in question are: *ACOGNO*, *ANDENOx*, *TWEXMMTH*, *UMCSENTx* and *VXOCLSx*. We decide to follow Huang et al. (2022) and remove these five variables from our data set, which means we are left with 123 macroeconomic variables that act as our predictors during the construction of forecasting models. In addition, to ensure the stationarity of our data, we follow McCracken and Ng (2016) and apply various data transformations to certain variables of our data set. This leads to the time series in the data set becoming stationary. The specific steps of these 7 data transformations are described in the Appendix of McCracken and Ng (2016).

3 Methodology

In this Section, we examine the construction of forecasts through 5 different models, namely: PCR, sPCR, RF, PCA-RF, and sPCA-RF. The usage of these models is inspired by current literature, which is extensively discussed in Section 1. Furthermore, we discuss PCA and PCR in Subsection 3.1. Next, in Subsection 3.2, we discuss sPCA and sPCR. Thereafter, in Subsection 3.4, we examine the Random Forest method. After this, we talk about the construction of forecasts, their evaluation metrics, and the comparison between them in Subsections 3.5, 3.6, and 3.7.

3.1 PCA and the selection of components

Principal component analysis (PCA), as described by Esbensen et al. (2002), reduces the dimension in a data set by reducing a large set of variables into a smaller one while preserving most of the original set's information. Dimension reduction attempts to exchange a little accuracy for simplicity by reducing the number of variables, which typically results in a loss of accuracy.

According to the literature, data augmentation techniques like PCA make it simpler and faster to examine the data set for ML methods, which could improve the performance of the ML approach (Chepurko et al., 2020; Cunningham, 2008; Waqar et al., 2017; Wong et al., 2016).

PCA consists of multiple steps; we first standardize the data so that each variable contributes equally to the analysis, after which we compute the covariance matrix of the data set. Next, we compute eigenvalues and eigenvectors of the covariance matrix to identify the principal components (PC), with which we reduce the dimension of your data set by only retaining a certain amount of PC's, which are selected through a criterion. More specifically, let $\mathbf{X} = (X_1, X_2, \ldots, X_k)$ be a $n \times k$ matrix representing our data set, where n is the number of observations, and k is the number of variables. We standardize each predictor X_i ($\forall i \in (1, \ldots, k)$) as

$$Z_i = \frac{X_i - \mu_i}{\sqrt{\sigma_{ii}}}, \tag{1}$$

where Z_i represents the standardized predictor i, μ_i represents the mean of predictor X_i and $\sqrt{\sigma_{ii}}$ represents the standard deviation of X_i. Then we obtain our standardized $n \times k$ dataset $\mathbf{Z} = (Z_1, Z_2, \ldots, Z_k)$ of which we can compute the $k \times k$ covariance matrix $\mathbf{\Sigma}$. Next, we solve the characteristic equation

$$\det(\mathbf{\Sigma} - \lambda \mathbf{I}) = 0, \tag{2}$$

where $\mathbf{I}$ is the $k \times k$ identity matrix. By solving the characteristic equation we find the eigenvalues λ_i and eigenvectors v_i of the covariance matrix. After which we pair each eigenvalue λ_i with an eigenvector v_i such that

$$(\Sigma - \lambda_i I) v_i = 0. \tag{3}$$

We now have the eigenvalues λ_i and the corresponding eigenvectors v_i, with which we compute the $k \times k$ matrix $\mathbf{V} = [v_1, v_2, \ldots, v_k]$ whose columns are the eigenvectors. Without loss of generality, this matrix $\mathbf{V}$ is ordered in such a way that the first column corresponds to the eigenvector with the largest eigenvalue. In addition, we can calculate the $k \times k$ diagonal matrix $\mathbf{\Lambda}$ whose elements are the associated eigenvalues λ_i. Consequently, we can compute a diagonal $k \times k$ matrix $\mathbf{\Sigma}_{Var}$ where the element on the diagonal in column i represents the variance explained by principal component i. The computation of $\mathbf{\Sigma}_{Var}$ is given by

$$\mathbf{\Sigma}_{Var} = \frac{\mathbf{\Lambda}}{tr(\mathbf{\Lambda})}, \tag{4}$$

where $tr(\mathbf{\Lambda})$ denotes the trace of matrix $\mathbf{\Lambda}$. Finally, we derive the Principal Components by multiplying our matrix of standardized data $\mathbf{Z}$ by $\mathbf{V}$

$$\mathbf{P} = \mathbf{ZV}. \tag{5}$$

The resulting $n \times k$ matrix $\mathbf{P}$ has the principal components as columns, where the first column corresponds to the principal component which explains the most variance of the data set $\mathbf{Z}$. We can now reduce the dimensions of our $n \times k$ data set by only retaining a selection of principal components. With regards to matrix $\mathbf{P}$, this means that we remove some of its columns if we

aim to reduce the dimension of our data, resulting in a $n \times g$ matrix $\mathbf{G}$ that represents our reduced data set, with $g < k$.

3.1.1 The selection of Principal components through 10-fold Cross-Validation

The selection of principal components to retain can be chosen through various criteria. Commonly used criteria within the literature are information criteria such as AIC and BIC (Bai and Ng, 2002), kaiser's rule (Dunteman, 1989), variance-explained thresholding (Dunteman, 1989), and the elbow test (Dunteman, 1989). However, Ferré (1995) shows that kaiser's rule, variance-explained thresholding, and the elbow test often fail to select the correct amount of principal components properly. In addition, Guo et al. (2023) show that no information criterion can consistently identify latent factors when factor strength is too weak. Therefore, we use k-fold cross-validation to select the number of principal components to retain and forecast. This procedure has support within the literature (Eastment and Krzanowski, 1982; Thomaz and Giraldi, 2010). Furthermore, we choose to perform k-fold cross-validation with k=10, where we restrict the maximum number of factors at five, which is in line with Huang et al. (2022). Using ten folds has proven to result in a low bias in a predicting context (Molinaro et al., 2005). The cross-validation procedure is extensively described by Berrar (2019).

3.2 sPCA

Scaled principle component analysis, or sPCA, is a dimension reduction technique closely related to PCA and is described by Huang et al. (2022). This method addresses a major problem that PCA faces: Scaled PCA does not ignore the target variable when reducing the dimension of a data set. It incorporates information on the target variable by putting more weight on predictors with a higher forecasting power, which is done by regressing each predictor individually onto the target variable and scaling this predictor by the predictive slope of the regression. This leads to better detection of the signals of latent factors by sPCA, consequently leading to more unbiased forecasts than the forecast of PCA. Before discussing sPCA we first introduce some mathematical notation. Let $\mathbf{X} = (X_1, X_2, \ldots, X_k)$ be a $n \times k$ matrix representing our data set, where n is the number of observations and k is the number of predictors and let y_t be the target variable that we forecast. Then we follow Huang et al. (2022) by first standardizing our data set $\mathbf{X}$ to $\mathbf{Z}$ in a similar manner as described in Subsection 3.1. Then we run, $\forall i \in (1, \ldots, k)$, the predictive regression

$$y_{t+1} = \alpha_i + \beta_i Z_{i,t} + \epsilon_{i,t+1}, \tag{6}$$

where y_{t+1} represents the target variable in period t+1, $Z_{i,t}$ represents the standardized predictor i in period t and $\epsilon_{i,t+1}$ represents the error term in period $t+1$. Subsequently, we store, $\forall i \in (1, \ldots, k)$, the estimated coefficients $\hat{\beta}_i$ within a vector, and we winsorize the vector to diminish the effect of extreme values. Next, we scale our data set $\mathbf{Z_t}$ with the vector of predictive slopes. Finally, we follow the same steps PCA performs on the standardized data set $\mathbf{Z_t}$, after which we obtain the scaled principal components, defined as sPCA factors.

3.3 (Scaled) Principal Component Regression

Principal component regression (PCR) is a technique often used in forecasting and is proposed by Stock and Watson (2002). It combines PCA and linear regression by using principal components as predictors in a linear regression to forecast a target variable. We follow the model proposed by Huang et al. (2022) to make one-step ahead forecasts with PCA and sPCA factors, computed as

$$y_{t+1} = \alpha^{PCA} + \pi^{PCA} g_t^{PCA}, \tag{7}$$

$$y_{t+1} = \alpha^{sPCA} + \pi^{sPCA} g_t^{sPCA}, \tag{8}$$

where $(\alpha^{PCA}, \pi^{PCA})$ and $(\alpha^{sPCA}, \pi^{sPCA})$ are the respective slopes of the two predictive regressions. Furthermore, g_t^{PCA} and g_t^{sPCA} represent the vector of PCA and sPCA which contain the first r_1 PCA factors and the first r_2 sPCA factors respectively. The value of r_i, (i=1,2), is computed through 10-fold Cross-Validation and has a maximum value of five, as described in Section 3.1.1.

3.4 Random Forest

In our research, we also use random forest (RF) to forecast the stock market. RF is a machine learning technique introduced by Breiman (2001) and belongs to the ensemble methods. The first step of RF is to perform bootstrap sampling on the data. More specifically, we create k bootstrap data samples with replacement. Unfortunately, this means that the dependence structure across series and the time series structure itself is ignored, which is a limitation in our research. However, RF still can capture nonlinear relationships in the data and remain predictively accurate (Athanasopoulos et al., 2011; Breiman, 2001) when ignoring this structure. Moreover, we use an expanding window while constructing our forecasts, which, although it does not address the time series dependence structure directly, partially addresses it because a model is always trained on past data. Therefore, it respects the temporal order of the time series data. Additionally, the model can still adapt to changes in the underlying data over time because we estimate and tune the model in each iteration. After we perform bootstrap sampling, we have k samples that are the same size as our data set. Their observations are randomly chosen from the original data, allowing for repetition. By allowing for repetition, our bootstrap samples contain, in theory, approximately 63.2% of the data, which means that we have an out-Of-bag sample that contains roughly 36.8% of the data. For the theoretical derivation of the bootstrap and out-of-bag sample size, we refer to Appendix A. Via the k bootstrap samples, we can create the so-called 'forest'. This forest consists of k decision trees, where k corresponds with the hyperparameter $ntree$. Furthermore, each tree corresponds to a unique bootstrap sample, where at each split in the decision tree, a subset of all predictors is randomly chosen at the nodes. The size of this subset corresponds to the hyperparameter $mtry$. Due to the randomness of choosing a subset of predictors at each split, there is an increase in diversity between all trees, making RF more robust and less at risk for overfitting.

The construction of our forecast $\hat{y}_{t+1}$ starts by making a prediction with each unique decision tree. Within an individual decision tree, we randomly choose a subset of predictors at each

node and select an optimal partition (split point) of the bootstrap sample for this subset. The optimal split point is determined by minimizing an impurity measure, such as Gini impurity or the variance of the child nodes. The usage of Gini impurity has support within the literature (Kim et al., 2018; Qiu et al., 2017). However, we opt to minimize the variance in the child nodes to determine the optimal split point due to the good results of Lahouar and Slama (2017) and Wang et al. (2022), who use this criterion in the context of both forecasting and the usage of augmented RF (PCA-RF). Hence, we utilize the R package *randomForest* to implement RF, which determines the split point based on minimizing the mean squared error in the child nodes. This results in two child nodes, onto which we recursively perform the same steps as the first node. We follow Arsham et al. (2022) and use node purity and minimum node size as stopping criteria for building extra nodes. Subsequently, when the decision tree is fully grown, we give the observation x_t to our tree, and the value of the leaf node (node that has no child nodes) that is reached gives the prediction $f_i(x_t)$ of the decision tree i. We repeat this process for all k different decision trees. Finally, the computation of our prediction $\hat{y}_{t+1}$ is given by

$$\hat{y}_{t+1} = \frac{1}{ntree} \sum_{i=1}^{ntree} f_i(x_t) \tag{9}$$

where $f_i(x_t)$ represents the prediction made by the i-th decision tree when given observation x_t, and $ntree$ corresponds to the amount of trees in our forest. More specifically, we average the predictions made by all decision trees to make our final prediction of y_{t+1}.

3.4.1 Hyperparameter tuning

Implementing the RF method through the R package *randomForest* does not automatically tune the model. Hence we optimize our model manually by tuning the hyperparameters $ntree$ and $mtry$. The results of Probst et al. (2019) and Salles et al. (2015) show that tuning hyperparameters using the OOB sample enhances model performance. The OOB sample consists of observations that are not used within a specific bootstrap sample, therefore it differs for each tree. We utilize the sample as test data, with which we can tune certain parameters to improve forecasting performance. The test error, which is crucial in tuning our hyperparameters, is approximated quite well by the OOB error (Salles et al., 2015). Therefore, we use the OOB error to tune our hyperparameters $ntree$ and $mtry$. The default values of these hyperparameters are 500 for $ntree$ and $\frac{p}{3}$ for $mtry$, where p represents the number of predictors. We opt to tune the hyperparameter $ntree$ first since RF models are generally insensitive to the value of $mtry$ (Breiman, 2001), as opposed to $ntree$, which is a crucial hyperparameter because too many trees make the model inefficient, while just the right number of trees stabilize the error (Probst et al., 2019). We begin by estimating a RF model with the default values, after which we determine the value of $ntree$, which minimizes the OOB error. After that, we construct a RF model with the optimized $ntree$ hyperparameter and still the default value for $mtry$. Then we can determine the optimal value of $mtry$ by minimizing the OOB error for various values of $mtry$. Finally, with the optimal values for both hyperparameters, we construct our tuned RF model with which we make our prediction.

3.4.2 RF augmentation

In addition to the standard RF method, described in Section 3.4, this research also augments the RF method by PCA and sPCA. We follow the steps described in Sections 3.1 and 3.2 to apply PCA or sPCA on our data set and reduce its dimension. Consequently, we follow the steps described in Section 3.4 to utilize RF on the augmented data. Thereafter, we obtain the PCA-RF or sPCA-RF method, depending on the factors used, while reducing the dimension of the data.

3.5 Forecasting

Our research forecasts the returns and volatility of the S&P-500 and NASDAQ-100, which are described in Section 2. The forecasts are 1-month ahead, and we consider the model

$$y_{t+1} = h(F_t) + u_{t+1}, \tag{10}$$

where y_{t+1} is the target variable in month $t+1$, F_t represents either a vector of the 123 macroeconomic variables in the FRED-MD or (s)PCA factors, and the h in the formula stands for the specific model/function used for forecasting our target variables. More specifically, h can represent PCR, sPCR, RF, PCA-RF, and sPCA-RF, which are described in Subsections 3.3, 3.4, and 3.4.2. Lastly, u_{t+1} is a zero-mean random error. Consequently, we have the following model for forecasting our target variables, which we use in our research:

$$\hat{y}_{t+1} = \hat{h}(\hat{F}_t) \tag{11}$$

3.5.1 Expanding window

When estimating the models described in Subsections 3.3, 3.4 and 3.4.2 and Equation (11), we utilize an expanding window, this means that we recursively estimate and update all the coefficients within our model as we predict more observations. We have an in-sample period of 120 observations, corresponding to 10 years, and the rest of the data is used as an out-of-sample. More specifically, we train our models with the observations up to time t and then predict y_{t+1} with the data of F_t. Consequently, meaning that for each prediction, we re-estimate the coefficients as our 'training data' expands. For RF, this means we keep tuning the hyperparameters (*ntree* and *mtry*), which makes our model more robust. With regards to PCR and sPCR, this consequently means that we recalculate the sPCA and PCA factors before each prediction to update our model. Lastly, Table 2 shows the exact in- and out-of-sample periods per target variable. The VXN has a relatively short out-of-sample period, corresponding to little training

Table 2: In- and Out-Of-Sample periods of the target variables.

	In-sample period	Out-Of-Sample period
US500	February 1970 - January 1980	February 1980 - December 2019
NDX	October 1985 - September 1995	October 1995 - December 2019
VIX	February 1990 - January 2000	February 2000 - December 2019
VXN	November 2003 - October 2013	December 2013 - December 2019

and few predictions. This could lead to inconsistent results in Section 4, as few observations negatively influence the interpretability of the results. For example, Davydenko and Fildes (2013) shows that traditional evaluation metrics of forecasting models such as MAE and MSE can be misleading when forecasting with small data sets.

3.6 Forecasting evaluation

3.6.1 MAE and MSE

Two commonly used forecast evaluation metrics are the mean squared error (MSE) and the mean absolute error (MAE). The MAE and MSE can be computed as

$$MSE = \frac{1}{n} \sum_{t=121}^{n} (y_t - \hat{y}_t)^2, \tag{12}$$

$$MAE = \frac{1}{n} \sum_{t=121}^{n} |y_t - \hat{y}_t|, \tag{13}$$

where y_t represents the actual value of the target variable, $\hat{y}_t$ represents the predicted value and n is the number of observations. Even though Hyndman and Koehler (2006) advocates for both evaluation metrics, they also discuss some strengths and weaknesses. The MSE penalizes large errors more, which is a good thing when that is desirable, but it also makes the evaluation metric sensitive to outliers. On the other hand, the MAE does not penalize large errors more and gives each error the same weight due to taking the absolute value of the errors, which can be desirable. The downside of taking the absolute value of errors is removing the consideration of over- and under-prediction of the target variable.

3.6.2 Subsample performance: CSSED

To evaluate relative subsample performance between two forecasts we use the cumulative sum of squared error differentials (CSSED), which is proposed by Welch and Goyal (2008). The CSSED between forecasts i and j is calculated as

$$CSSED_{i,j} = \sum_{t=121}^{n} (e_{i,t}^2 - e_{j,t}^2), \tag{14}$$

where n is the number of observations and $e_{i,t}$ is the error of forecast i of observation t. Subsequently, we plot the value of $CSSED_{i,j}$ through time, after which we can observe and evaluate the relative subsample performance between forecasts i and j. When the value of CSSED is positive and rising, the plot shows that the line has a positive slope, which indicates that forecast j is more accurate than forecast i.

3.7 Comparing two forecasts: Modified Diebold-Mariano test

After constructing forecasts with various models, we compute the values of our evaluation metrics, described in Section 3.6, with which we can compare forecasts. However, a 'better' value of an evaluation metric does not necessarily indicate a significant out-performance. This means we

cannot make definite conclusions about the accuracy of different forecasts through evaluation metrics alone. Consequently, we perform a proper test to do this. One of the most common methods for testing equal prediction accuracy is the Diebold-Mariano (DM) test (Diebold and Mariano, 1995). However, this test is seriously oversized for moderate numbers of sample observations, and even though the test is more versatile than any alternative test of equality of forecast performance, this problem grows bigger as the forecast horizon increases (Harvey et al., 1997). Harvey et al. (1997) describes a modified DM-test, which is closely related to the original DM-test, but it alleviates the oversized test problem and performs better in all cases (Harvey et al., 1997). Their results specifically show a drastic improvement in test performance at the smallest sample sizes, which is interesting for our research because our target variable, Nasdaq-100 volatility, only has a small sample size. Therefore, we choose to perform a modified version of the DM-test, as described by Harvey et al. (1997).

We first introduce notation before discussing the modified DM-test. Let $e_{i,t}$ be the error of forecast i at time t which is computed as

$$e_{i,t} = \hat{y}_{i,t} - y_{i,t} \qquad \text{for} \quad i = 1, 2. \tag{15}$$

Furthermore, we introduce a loss function d_t and average loss $\bar{d}$, computed as

$$d_t = e_{it}^2 - e_{jt}^2 \qquad \text{and} \qquad \bar{d} = \frac{1}{n}\sum_{t=1}^{n} d_t, \tag{16}$$

where n represents the amount of observations made. Next we calculate the variance of our loss function $\hat{V}(d_t)$ and μ, the expected value of d_t, as follows

$$\hat{V}(d_t) = \frac{1}{n}\sum_{t=1}^{n}(d_t - \bar{d})^2 \qquad \text{and} \qquad \mu = E[d_t], \tag{17}$$

where $\mathrm{E}[d_t]$ represents the expected value of d_t. Furthermore, the hypothesis that we test is equal prediction accuracy, that is $\mathrm{E}[d_t]$=0. Hence we are formally testing

$$H_0 : \mu = E[d_t] = 0 \qquad vs. \qquad H_a : \mu = E[d_t] \neq 0. \tag{18}$$

The test statistic of the modified DM-test is similar to the DM test statistic described by Diebold and Mariano (1995). The modified DM-test statistic DM^* is computed as

$$DM^* = \sqrt{\frac{n+1-2h+n^{-1}h(h-1)}{n}} * DM \overset{\mathrm{h=1}}{=} \sqrt{\frac{n-1}{n}} * \frac{\bar{d}-\mu}{\sqrt{\frac{V(d_t)}{n}}} \overset{\mathrm{H_0}}{=} \sqrt{\frac{n-1}{n}} * \frac{\bar{d}}{\sqrt{\frac{V(d_t)}{n}}} \sim t(n-1), \tag{19}$$

where h represents the forecast horizon (h-step ahead forecast), which is equal to 1 in our research, and $t(n-1)$ is the Student's t distribution with $(n-1)$ degrees of freedom. Finally, for a formal proof and further description of the DM^*-test statistic and its distribution, we refer to Harvey et al. (1997).

4 Results

4.1 In-sample results

We perform PCA and SPCA on our data to determine what type of macro variables are most important within the principal components and, consequently, our forecasts. The variance explained by each of the first five principal components is shown in Table 3.

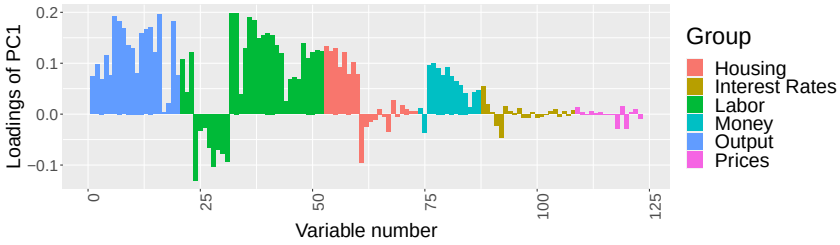
Table 3: Variance explained by the first five factors of each respective method

		sPCA			
	PCA	S&P-500 returns	S&P-500 volatility	Nasdaq-100 returns	Nasdaq-100 volatility
1st	0.15	0.18	0.26	0.18	0.40
2nd	0.07	0.15	0.17	0.10	0.07
3rd	0.07	0.08	0.10	0.09	0.06
4th	0.05	0.06	0.07	0.06	0.04
5th	0.04	0.04	0.05	0.05	0.04

Note: When the eigenvalues are normalized to have sum of one, this table also reports the 1st to 5th eigenvalues in a descending order for the covariance matrixes of the (scaled) macro variables, where the scaling parameter is the predictive slope of the variable on the forecasted target (as described in Section 3.2).

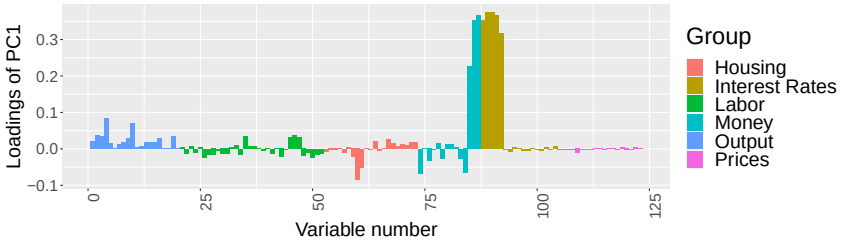
We see that there is clear a difference in variance explained by the principal components of PCA and sPCA. The first few principal components of each sPCA factor explain more variance than that of PCA. Whereas sPCA, regarding the two volatility target variables, explains the most variance with a single PC. This is consistent with the results of Huang et al. (2022), where the factors of sPCA explain more variance with fewer principal components. Due to the differences in variance explained by the PC's, we look further into the factors by investigating the loadings of all the first factors. Figure 1 shows the loadings of the first PCA factor. We see that

Figure 1: Loadings of the first PCA factor



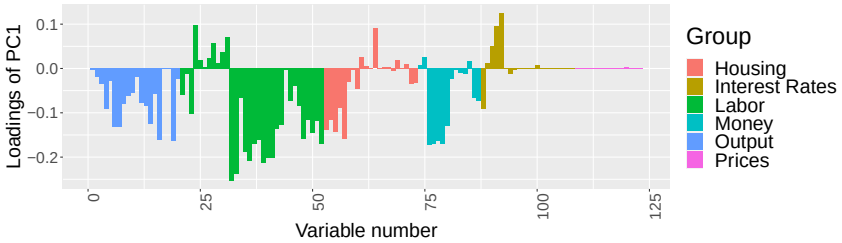
macro variables corresponding with output, labor, housing, and money contribute significantly to the PCA factor. Consequently, apart from the variables related to interest rates and prices, we see that many variables are important in the first PCA factor and that no individual group is the most important for this PC. In terms of learning which specific kind of macro variables is good for predicting target variables, we do not learn much. Next, we look at Figure 2, which shows the loadings of the first sPCA factor regarding S&P-500 returns. In this sPCA factor, we can see two groups of macro variables that are the most important for the PC: money and interest rates. More specifically, securities in bank credit at all commercial banks, the effective federal funds rate, and macroeconomic variables regarding the T-bill and treasury rate

Figure 2: Loadings of the first sPCA factor with regards to S&P-500 returns



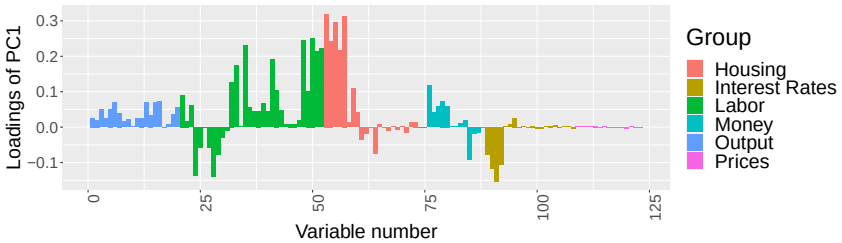
contribute significantly to this PC. The fact that these specific two groups are the most important could be since the return of a S&P-500 stock is closely related to variables corresponding with money and interest rates. For example, if the interest rate declines, buying stocks becomes more attractive, as opposed to other buying bonds or saving, increasing the demand and price of stocks. Furthermore, we look at Figure 3 that shows the loadings of the first sPCA factor regarding S&P-500 volatility. Figure 3 shows that nearly all groups are of importance for the

Figure 3: Loadings of the first sPCA factor with regards to S&P-500 volatility



first factor, with loadings of housing, labor, output, money, and interest rates all having variables with significant loadings. Variables belonging to these groups can all affect consumer spending behaviour and, therefore, volatility in the market, which in this case means an effect on the stock volatility of the S&P-500. For example, labor-related variables, such as unemployment rates or wage growth, can clearly affect consumer sentiment and spending when the unemployment rate is high or wage growth is low. Next, we look at Figure 4, which shows the loading of the first sPCA factor regarding Nasdaq-100 returns. We see that mainly labor and housing-related variables

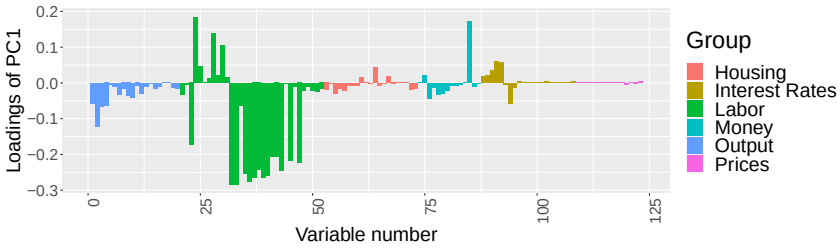
Figure 4: Loadings of the first sPCA factor with regards to Nasdaq-100 returns



are important in this factor. However, variables related to money and interest rates also play a minor role. Even though money and interest rates still contribute, the biggest contributors to this factor are variables related to labor and housing, which is different than for the sPCA factor with regards to S&P-500 returns (shown in Figure 2). This could be explained due to the fact that the Nasdaq-100 consists of different types of companies than the S&P-500. More specifically, it

consists of technology and innovative companies, which can significantly be affected by changes in the labor market because tech companies often rely on highly skilled labor. Furthermore, the housing market affects consumer spending and confidence in the economy. Technological and innovative companies, some of which are start-ups, are often very growth-oriented and rely on external financing for growth. This external financing is sensitive to consumer spending and confidence in the economy, which could be why housing-related variables contribute a lot to this factor. Lastly, we look at Figure 5, which shows the loadings of the first sPCA factor concerning Nasdaq-100 volatility. When looking at Figure 5 and comparing it to the loadings of

Figure 5: Loadings of the first sPCA factor with regards to Nasdaq-100 volatility



the sPCA factor regarding Nasdaq-100 returns (Figure 4), we observe a similarity, which is that we see that this sPCA factor is significantly influenced by labor-related variables, which can be explained through the same reasoning as to why labor-related variables influence Nasdaq-100 returns. However, we also see two key differences within the sPCA factor regarding Nasdaq-100 volatility: Housing-related variables do not contribute significantly, and there is a significant contribution of an individual money-related variable to the PC. The first difference can be explained due to the fact that housing-related variables often relate to the general health of an economy, thus directly influencing returns. The effect on volatility might be more indirect and thereby quieted. Furthermore, the individual money-related variable is securities in bank credit at all commercial banks. A drastic change of securities in bank credit might signal upcoming instability of the economy, which in turn leads to panic on the market, and thereby, it has a direct effect on volatility but a lesser direct effect on returns.

4.2 Out-of-sample results

In this subsection, we investigate and evaluate our forecasts through various evaluation metrics and other characteristics. We have five models with which we construct forecasts for four target variables. In Tables 4 through 7, we show the relative MSE, MAE, and runtime of these forecasts while taking PCR as the benchmark. Runtime indicates the duration of running each model and constructing the forecast. In addition, we also report the mean variance explained, which represents the average percentage of variance explained by the chosen number of PC's in each iteration. Next, to evaluation metrics and characteristics, we include the p-values of modified DM-tests to draw accurate conclusions while comparing the forecasts. Lastly, we also report the number of predictions made, denoted by n. Table 4 shows the evaluation of the forecasts of S&P-500 returns. When looking at Table 4, there are five main takeaways. First, when looking at relative MSE and MAE, we see that the forecast made with the PCR model is the least accurate compared to all other models and, more specifically, it shows when looking at the p-values of the

Table 4: Relative forecasting performance and Diebold-Mariano test results for all models with regard to S&P-500 returns, using PCR as benchmark model

$n = 479$	PCR	sPCR	RF	PCA-RF	sPCA-RF
MSE	1.00	0.98	0.96	0.28	0.30
MAE	1.00	0.99	0.98	0.53	0.54
Variance Explained	0.28	0.34	-	0.27	0.26
Runtime	1.00	1.11	61.62	1.59	2.73
P-value DM* Test (vs. PCR)	-	0.034^{**}	0.040^{**}	$< 0.001^{***}$	$< 0.001^{***}$
P-value DM* Test (vs. sPCR)	-	-	0.309	$< 0.001^{***}$	$< 0.001^{***}$
P-value DM* Test (vs. RF)	-	-	-	$< 0.001^{***}$	$< 0.001^{***}$
P-value DM* Test (vs. PCA-RF)	-	-	-	-	0.297

Note: Statistical significance for $\alpha = 0.1$, 0.05 and 0.01 is denoted by *, ** and *** respectively. Furthermore, the values of MSE, MAE and runtime are the relative values while taking PCR as benchmark.

DM* test versus PCR, that this forecast is significantly outperformed by all other models on a 5% significance level and outperformed by PCA-RF and sPCA-RF on a 1% significance level. In addition, we also see that the forecast of sPCR is significantly outperformed by both PCA-RF and sPCA-RF on a 1% significance level. Secondly, we look at the p-values of the DM* test versus RF and observe that the forecasts of PCA-RF and sPCA-RF significantly outperform RF on a 1% significance level. The reason for PCA-RF and sPCA-RF outperforming the forecast of RF could be due to three different reasons: high dimensionality, multicollinearity among predictors, and noise within the data. High dimensionality possibly leads to the usage of many redundant variables and, consequently, a less accurate and more complex model. Furthermore, multicollinearity among predictors can cause instability in estimating the relationship between each predictor and the target variable, making it difficult to construct a model accurately. Lastly, even though noise reduction is not the main strength of (s)PCA, it still addresses the problem, leading to the model responding better to unseen data. We address all these problems by combining RF with PCA or sPCA. Not only do we enhance forecasting performance by combining RF with PCA or sPCA, it also greatly reduces the computation time of RF, resulting in a much faster construction of forecasts. A decrease in runtime is because both PCA and sPCA reduce the number of predictors, and therefore RF has fewer nodes and smaller tree depth within the constructed forest, which leads to a lower computation time. Next, our third takeaway is that when looking at the DM* test versus PCA-RF, we see that PCA-RF does not significantly outperform the forecast of sPCA-RF. This suggests that the differences between PCA and sPCA are insignificant when combining the methods with a ML method while forecasting. Moreover, our fourth takeaway is that constructing forecasts for S&P-500 using PCA or sPCA without a ML method leads to sPCR significantly outperforming PCR (while restricting the number of factors chosen at 5). This is consistent with the results of Huang et al. (2022). Lastly, our fifth main takeaway is that the variance explained by the chosen number of principal components says little about forecasting performance. For example, PCR nearly has the same percentage of variance explained as both PCA-RF and sPCA-RF (0.28 vs. 0.27), but all other models significantly outperform the forecast. Next, we look at Table 5, which shows the evaluation of the forecasts of S&P-500 volatility. We observe that while forecasting the S&P-500 volatility, the forecasts made by RF, PCA-RF, and sPCA-RF significantly outperform the forecasts of PCR and sPCR on a 1% significance level. In addition, we see that the forecasts of PCA-RF

Table 5: Relative forecasting performance and Diebold-Mariano test results for all models with regard to S&P-500 volatility, using PCR as benchmark

$n = 239$	PCR	sPCR	RF	PCA-RF	sPCA-RF
MSE	1.00	0.95	0.47	0.19	0.24
MAE	1.00	$1.00^{(!)}$	0.66	0.40	0.42
Variance Explained	0.41	0.41	-	0.37	0.37
Runtime	1.00	1.24	19.38	1.08	2.51
P-value DM* Test (vs. PCR)	-	0.154	$< 0.001^{***}$	$< 0.001^{***}$	$< 0.001^{***}$
P-value DM* Test (vs. sPCR)	-	-	$< 0.001^{***}$	$< 0.001^{***}$	$< 0.001^{***}$
P-value DM* Test (vs. RF)	-	-	-	$< 0.001^{***}$	$< 0.001^{***}$
P-value DM* Test (vs. PCA-RF)	-	-	-	-	0.435

Note: Statistical significance for $\alpha = 0.1$, 0.05 and 0.01 is denoted by *, ** and *** respectively. Furthermore, the values of MSE, MAE and runtime are the relative values while taking PCR as benchmark. (!): This value is lower than 1.00 when rounded to three decimals.

and sPCA-RF also outperform the RF forecast but do not significantly outperform each other. Furthermore, the variance explained still says little about forecasting performance. This is all in line with the results of Table 4; however, a difference we now observe is that the forecast of sPCR does not significantly outperform the forecast of PCR. A possible explanation could be that the loadings of the sPCA factor do not mainly consist of 1 or 2 groups, as opposed to the first sPCA factor with regards to S&P-500, which singles out variables related to money and interest rates as shown in Section 3 (Figures 2 and 3. This means that the characteristics of the sPCA loadings are similar to the PCA factors, whose loadings also consist of various groups (Figure 1 in Section 3). Next, we see that the forecast of RF significantly outperforms the benchmark and sPCR on a 1% significance level when forecasting S&P-500 volatility, as opposed to only on a 5% level when forecasting S&P-500 returns. This could be explained since RF performs well with high volatility ((Medeiros et al., 2021)). Lastly, we observe that the runtime of RF is greatly reduced when RF is augmented with PCA or sPCA. Subsequently, in Table 6, we show the results of the relative forecasting performance and modified DM tests concerning Nasdaq-100 returns. We

Table 6: Relative forecasting Performance and Diebold-Mariano test results for all models with regard to Nasdaq-100 returns, using PCR as benchmark

$n = 291$	PCR	sPCR	RF	PCA-RF	sPCA-RF
MSE	1.00	0.98	0.96	0.27	0.29
MAE	1.00	0.99	0.99	0.52	0.53
Variance Explained	0.16	0.22	-	0.18	0.18
Runtime	1.00	1.15	25.16	1.28	2.57
P-value DM* Test (vs. PCR)	-	0.137	0.231	$< 0.001^{***}$	$< 0.001^{***}$
P-value DM* Test (vs. sPCR)	-	-	0.472	$< 0.001^{***}$	$< 0.001^{***}$
P-value DM* Test (vs. RF)	-	-	-	$< 0.001^{***}$	$< 0.001^{***}$
P-value DM* Test (vs. PCA-RF)	-	-	-	-	0.648

Note: Statistical significance for $\alpha = 0.1$, 0.05 and 0.01 is denoted by *, ** and *** respectively. Furthermore, the values of MSE, MAE and runtime are the relative values while taking PCR as benchmark.

see that the forecast of PCR is only significantly outperformed by the forecasts of PCA-RF and sPCA-RF but not by those of sPCR and RF. RF not significantly outperforming PCR and sPCR might be because the relationship between Nasdaq-100 returns and our macroeconomic predictors is linear; this leads to a problem because RF is a non-linear model. Moreover, we see that the forecasts of PCA-RF and sPCA-RF do not significantly outperform each other, but

they outperform the RF forecast on a 1% significance level. Lastly, we observe that runtime decreases when augmenting RF with PCA or sPCA and that variance explained says little about forecasting performance while forecasting Nasdaq-100 returns. Next, we look at Table 7, which shows the results of the relative forecasting performance and modified DM tests with regard to Nasdaq-100 volatility. We see that the forecast of PCR is significantly outperformed by the

Table 7: Forecasting Performance and Diebold-Mariano test results for all models with regard to Nasdaq-100 volatility, using PCR as benchmark

$n = 74$	PCR	sPCR	RF	PCA-RF	sPCA-RF
MSE	1.00	1.08	0.65	0.23	0.21
MAE	1.00	1.01	0.77	0.43	0.41
Variance Explained	0.47	0.44	-	0.46	0.27
Runtime	1.00	1.11	18.37	1.68	2.79
P-value DM* Test (vs. PCR)	-	0.076^{*}	0.013^{**}	$< 0.001^{***}$	$< 0.001^{***}$
P-value DM* Test (vs. sPCR)	-	-	0.01^{**}	$< 0.001^{***}$	$< 0.001^{***}$
P-value DM* Test (vs. RF)	-	-	-	$< 0.001^{***}$	$< 0.001^{***}$
P-value DM* Test (vs. PCA-RF)	-	-	-	-	0.285

Note: Statistical significance for $\alpha = 0.1$, 0.05 and 0.01 is denoted by *, ** and *** respectively. Furthermore, the values of MSE, MAE and runtime are the relative values while taking PCR as benchmark.

forecasts of all other models. More specifically, the forecast of sPCR outperforms on a 10% significance level, the forecast of RF outperforms on a 5% level, and the forecasts of PCA-RF and sPCA-RF on a 1% level. In addition, we see that the forecasts of PCA-RF and sPCA-Rf do not significantly outperform each other but do significantly outperform the forecasts of RF and sPCR on a 1% significance level. Lastly, we see that runtime drastically decreases when PCA-RF and sPCA-RF augment RF, and that variance explained says little about forecasting performance. For example, PCR and PCA-RF almost have the same average percentage of variance explained, but they perform significantly different.

4.2.1 Subsample performance

Due to the forecasts of PCA-RF and sPCA-RF consistently outperforming all other models on a 1% significance level, we examine subsample performance to investigate whether there are specific periods that contribute to the outperformance. Through means of the CSSED, as described in Section 3.7, we construct Figures 7 through 9, where we display the loss differential d_t through time for all the models compared with our benchmark model PCR. We see that in Figure 6, the CSSED value is consistently growing, which is, for the most part, also true for the other figures. However, there are some periods in Figures 7, 8 and 9 where the line suddenly becomes steep, thus indicating a period where the forecasts of PCA-RF and sPCA-RF outperform the forecasts made by PCR even more. Figure 7 shows, for PCA-Rf and sPCA-RF, an increase in CSSED of nearly 3000 in a 6-month time period, starting in September 2001. An explanation for this is the September 11 attacks (9/11), which significantly impacted the stock market. The exchanges were closed for four training sessions. By the time they reopened, the prices on the stock market had decreased drastically due to increased economic uncertainty, which led to increased market volatility and, thus, increased S&P-500 volatility. Next to PCA-RF and sPCA-Rf, we also see that the RF model responded to these events better than PCR.

Figure 6: Cumulative sum of squared forecast error differentials: S&P-500 returns

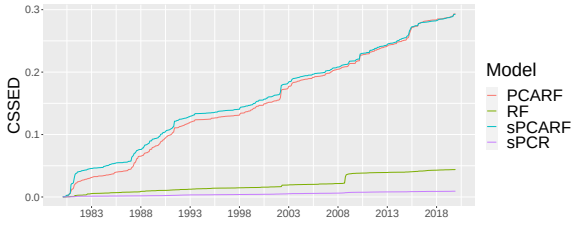


Figure 7: Cumulative sum of squared forecast error differentials: S&P-500 volatility

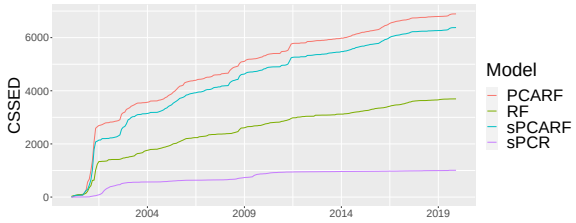


Figure 8: Cumulative sum of squared forecast error differentials: Nasdaq-100 returns

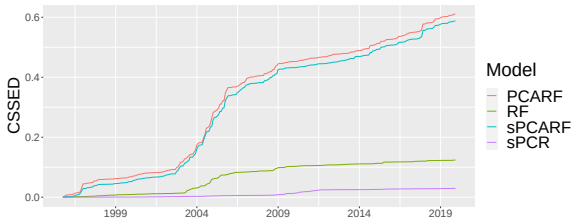
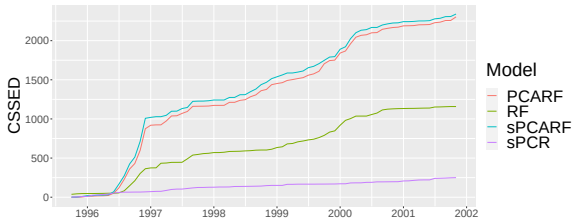


Figure 9: Cumulative sum of squared forecast error differentials: Nasdaq-100 volatility



These results are consistent with current literature that RF models relatively perform well when there is high volatility (Medeiros et al., 2021). Secondly, Figure 8 shows a relatively big increase in CSSED between 2002 and 2007, which could be a repercussion of the Dot-com Bubble Burst that occurred between 2000 and 2002, which was characterized by the rapid devaluation of tech stocks. In the years after this burst, there was a period of growth (2002-2007) where investor confidence increased, which resulted in positive returns, especially in the tech sector due to the rise of internet-based businesses. This event affects the relationship between our predictors and Nasdaq-100 returns, which probably means that this relationship becomes nonlinear, favoring RF models such as PCA-RF and sPCA-RF instead of PCR. Lastly, we observe in Figure 9 that in 1996, the CSSED of PCA-Rf and sPCA-RF quickly grows from nearly 0 to 1000. This could be explained by the fact that the start of the Internet boom was in 1996. The sudden growth of this sector led to a price increase of stocks related to the tech sector, which in turn could lead to a sudden increase in Nasdaq-100 volatility. This increase in volatility is handled better by the (augmented) RF models compared to the PCR and sPCR models. Overall we see that even though crises or big economic events drastically, but briefly, contribute to the significant forecasting outperformance of PCA-RF and sPCA-RF, the outperformance is also consistent without major events as the lines in Figures 6 through 9 continuously have positive slopes.

5 Conclusion

In this research, we perform PCA and sPCA on a macroeconomic data set of predictors and subsequently utilize RF to estimate the models PCA-RF and sPCA-RF, with which we construct forecasts of the simple returns and volatility of the S&P-500 and Nasdaq-100 stock indices. Subsequently, we investigate the effects on forecasting performance compared to regular RF, PCR,

and sPCR, which leads to our central research question:

How does the sPCA-RF method perform while forecasting stock indices, compared to PCA-RF and RF?

Our results show that forecasts of both PCA-RF and sPCA-RF consistently outperform PCR, sPCR, and RF on a 1% significance level while forecasting both the returns and volatility of S&P-500 and Nasdaq-100. However, the forecasts made by sPCA-RF never significantly outperform PCA-RF forecasts. We also see that the augmented variants of RF are computationally less expensive than RF but more expensive than PCR and sPCR. Nonetheless, the significant gain in forecasting accuracy when using PCA-RF or sPCA-RF, is a great trade for the loss in computation time when compared to PCR and sPCR. Therefore, to answer our main research question, the forecasts of sPCA-RF and PCA-RF significantly outperform those of RF on a 1% significance level, but sPCA-RF does not significantly outperform PCA-RF when forecasting stock indices. More specifically, the outperformance of PCA-RF and sPCA-RF with regards to PCR, sPCR, and RF is higher in times of crises that affect the target variables. However, also in stable economic times, both models consistently outperform the other models when constructing forecasts for our target variables. This implies that parties who aim to forecast the returns and volatility of the S&P-500 or Nasdaq-100, such as investors and hedge funds, should utilize either of the relatively computationally inexpensive PCA-RF or sPCA-RF models, to construct the most accurate forecasts. In addition to the results concerning our research question, we also see that when forecasting stock indices, the percentage of variance explained by a chosen number of principal components in a forecasting model says little about the forecasting performance of that model. This implies that variance explained is a bad criterion for determining the number of PC's to retain when reducing the dimension of a data set in a forecasting context.

For further research, it is interesting to see how the results would differ if one made different choices regarding used data, criteria for retaining a number of PC's, and which ML method to augment with PCA and sPCA. In our research, we use data from FRED-MD. However, this data has one slight drawback: A forward-looking bias. Therefore it would be interesting to extend our research by using the ALFRED-MD database, which consists of the actual data known at some point in the past which is the most realistic when constructing forecasts. Next, we utilize cross-validation to select a number of PC's to retain while reducing the dimension of our data, while the maximum of allowed factors is five. It is interesting to see if results change when one opts to use a different way than CV to select the number of PC's or a different maximum factor restriction. Furthermore, our RF models ignore both the time series structure and the dependence structure across series, which is a limitation of our research. Even though this is partly addressed by utilizing an expanding window, it would be interesting to see if our results change when the dependence and time series structures are incorporated in the RF, PCA-RF, and sPCA-RF models. Lastly, we only look at RF in this research, but it would be interesting to investigate if the forecasting performance of other ML methods changes when combining said method with PCA or sPCA.

References

Arak, M. and Mijid, N. (2006). The vix and vxn volatility measures: Fear gauges or forecasts? *Derivatives Use, Trading & Regulation*, 12:14–27.

Arsham, A., Rosenberg, P., and Little, M. (2022). Effects of stopping criterion on the growth of trees in regression random forests. *New Engl. J. Stat. Data Sci.*

Athanasopoulos, G., Hyndman, R. J., Song, H., and Wu, D. C. (2011). The tourism forecasting competition. *International Journal of Forecasting*, 27(3):822–844.

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

Berrar, D. (2019). Cross-validation.

Boivin, J. and Ng, S. (2006). Are more data always better for factor analysis? *Journal of Econometrics*, 132(1):169–194.

Breiman, L. (2001). Random forests. *Machine learning*, 45:5–32.

Chen, S.-S. (2009). Predicting the bear stock market: Macroeconomic variables as leading indicators. *Journal of Banking & Finance*, 33(2):211–223.

Chepurko, N., Marcus, R., Zgraggen, E., Fernandez, R. C., Kraska, T., and Karger, D. (2020). Arda: automatic relational data augmentation for machine learning. *arXiv preprint arXiv:2003.09758*.

Corrado, C. J. and Miller, Jr, T. W. (2005). The forecast quality of cboe implied volatility indexes. *Journal of Futures Markets: Futures, Options, and Other Derivative Products*, 25(4):339–373.

Cunningham, P. (2008). Dimension reduction. *Machine learning techniques for multimedia: Case studies on organization and retrieval*, pages 91–112.

Davydenko, A. and Fildes, R. (2013). Measuring forecasting accuracy: The case of judgmental adjustments to sku-level demand forecasts. *International Journal of Forecasting*, 29(3):510–522.

Diebold, F. X. and Mariano, R. S. (1995). Com paring predictive accu racy. *Journal of Business and Economic Statistics*, 13(3):253–263.

Dunteman, G. H. (1989). *Principal components analysis*, volume 69. Sage.

Eastment, H. and Krzanowski, W. (1982). Cross-validatory choice of the number of components from a principal component analysis. *Technometrics*, 24(1):73–77.

Esbensen, K. H., Guyot, D., Westad, F., and Houmoller, L. P. (2002). *Multivariate data analysis: in practice: an introduction to multivariate data analysis and experimental design*. Multivariate Data Analysis.

Ferré, L. (1995). Selection of components in principal component analysis: a comparison of methods. *Computational Statistics & Data Analysis*, 19(6):669–682.

Giot, P. (2002). The information content of implied volatility indexes for forecasting volatility and market risk. *Available at SSRN 362440*.

Guo, X., Chen, Y., and Tang, C. Y. (2023). Information criteria for latent factor models: A study on factor pervasiveness and adaptivity. *Journal of Econometrics*, 233(1):237–250.

Harvey, D., Leybourne, S., and Newbold, P. (1997). Testing the equality of prediction mean squared errors. *International Journal of forecasting*, 13(2):281–291.

Heij, C., Heij, C., de Boer, P., Franses, P. H., Kloek, T., van Dijk, H. K., et al. (2004). *Econometric methods with applications in business and economics*. Oxford University Press.

Huang, D., Jiang, F., Li, K., Tong, G., and Zhou, G. (2022). Scaled pca: A new approach to dimension reduction. *Management Science*, 68(3):1678–1695.

Hyndman, R. J. and Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International journal of forecasting*, 22(4):679–688.

Khaidem, L., Saha, S., and Dey, S. R. (2016). Predicting the direction of stock market prices using random forest. *arXiv preprint arXiv:1605.00003*.

Kim, Y., Hardisty, R., Torres, E., and Marfurt, K. J. (2018). Seismic facies classification using random forest algorithm. In *2018 SEG International Exposition and Annual Meeting*. OnePetro.

Lahouar, A. and Slama, J. B. H. (2017). Hour-ahead wind power forecast based on random forests. *Renewable energy*, 109:529–541.

Liu, D. and Sun, K. (2019). Random forest solar power forecast based on classification optimization. *Energy*, 187:115940.

Lu, X., Ma, F., Xu, J., and Zhang, Z. (2022). Oil futures volatility predictability: New evidence based on machine learning models. *International Review of Financial Analysis*, 83:102299.

Luan, J., Zhang, C., Xu, B., Xue, Y., and Ren, Y. (2020). The predictive performances of random forest models with limited sample size and different species traits. *Fisheries Research*, 227:105534.

Maysami, R. C., Howe, L. C., and Hamzah, M. A. (2004). Relationship between macroeconomic variables and stock market indices: Cointegration evidence from stock exchange of singapore's all-s sector indices. *Jurnal pengurusan*, 24(1):47–77.

McCracken, M. W. and Ng, S. (2016). Fred-md: A monthly database for macroeconomic research. *Journal of Business & Economic Statistics*, 34(4):574–589.

Medeiros, M. C., Vasconcelos, G. F., Veiga, Á., and Zilberman, E. (2021). Forecasting inflation in a data-rich environment: the benefits of machine learning methods. *Journal of Business & Economic Statistics*, 39(1):98–119.

Mikołajczyk, A. and Grochowski, M. (2018). Data augmentation for improving deep learning in image classification problem. In *2018 international interdisciplinary PhD workshop (IIPhDW)*, pages 117–122. IEEE.

Molinaro, A. M., Simon, R., and Pfeiffer, R. M. (2005). Prediction error estimation: a comparison of resampling methods. *Bioinformatics*, 21(15):3301–3307.

Nti, K. O., Adekoya, A., and Weyori, B. (2019). Random forest based feature selection of macroeconomic variables for stock market prediction. *American Journal of Applied Sciences*, 16(7):200–212.

Peres-Neto, P. R., Jackson, D. A., and Somers, K. M. (2003). Giving meaningful interpretation to ordination axes: assessing loading significance in principal component analysis. *Ecology*, 84(9):2347–2363.

Pilinkus, D. (2009). Stock market and macroeconomic variables: evidences from lithuania. *Ekonomika ir vadyba*, (14):884–891.

Pojarliev, M. and Polasek, W. (2001). Applying multivariate time series forecasts for active portfolio management. *Financial Markets and Portfolio Management*, 15(2):201.

Probst, P., Wright, M. N., and Boulesteix, A.-L. (2019). Hyperparameters and tuning strategies for random forest. *Wiley Interdisciplinary Reviews: data mining and knowledge discovery*, 9(3):e1301.

Qiu, X., Zhang, L., Suganthan, P. N., and Amaratunga, G. A. (2017). Oblique random forest ensemble via least square estimation for time series forecasting. *Information Sciences*, 420:249–262.

Raudys, S. J., Jain, A. K., et al. (1991). Small sample size effects in statistical pattern recognition: Recommendations for practitioners. *IEEE Transactions on pattern analysis and machine intelligence*, 13(3):252–264.

Rossi, A. G. (2018). Predicting stock market returns with machine learning. *Georgetown University*.

Salles, T., Gonçalves, M., Rodrigues, V., and Rocha, L. (2015). Broof: exploiting out-of-bag errors, boosting and random forests for effective automated classification. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pages 353–362.

Stock, J. H. and Watson, M. W. (2002). Forecasting using principal components from a large number of predictors. *Journal of the American statistical association*, 97(460):1167–1179.

Thomaz, C. E. and Giraldi, G. A. (2010). A new ranking method for principal components analysis and its application to face image analysis. *Image and vision computing*, 28(6):902–913.

Wang, D., Cui, X., and Niu, D. (2022). Wind power forecasting based on lstm improved by emd-pca-rf. *Sustainability*, 14(12):7307.

Waqar, M., Dawood, H., Guo, P., Shahnawaz, M. B., and Ghazanfar, M. A. (2017). Prediction of stock market by principal component analysis. In *2017 13th International conference on computational intelligence and security (CIS)*, pages 599–602. IEEE.

Wei, X. and Ouyang, H. (2023). Forecasting carbon price using double shrinkage methods. *International Journal of Environmental Research and Public Health*, 20(2):1503.

Welch, I. and Goyal, A. (2008). A comprehensive look at the empirical performance of equity premium prediction. *The Review of Financial Studies*, 21(4):1455–1508.

Wild Ali, A. B. (2021). Prediction of employee turn over using random forest classifier with intensive optimized pca algorithm. *Wireless Personal Communications*, 119(4):3365–3382.

Wong, S. C., Gatt, A., Stamatescu, V., and McDonnell, M. D. (2016). Understanding data augmentation for classification: when to warp? In *2016 international conference on digital image computing: techniques and applications (DICTA)*, pages 1–6. IEEE.

Yin, L., Li, B., Li, P., and Zhang, R. (2023). Research on stock trend prediction method based on optimized random forest. *CAAI Transactions on Intelligence Technology*, 8(1):274–284.

Ziane, A., Necaibia, A., Sahouane, N., Dabou, R., Mostefaoui, M., Bouraiou, A., Khelifi, S., Rouabhia, A., and Blal, M. (2021). Photovoltaic output power performance assessment and forecasting: Impact of meteorological variables. *Solar Energy*, 220:745–757.

Appendices

A Derivation out-of-bag sample size

As described in Section 3.4, we perform bootstrap sampling on our data, where each sample has observations randomly chosen from the original data while allowing for repetition. Consequently, certain observations are not chosen within a specific bootstrap sample, resulting in an OOB sample. We can derive the percentage of observations not picked when filling the bootstrap sample with observations of the original data. Let there be N observations in the training data set, then the probability of not picking an observation in a random draw is given by

$$\frac{N-1}{N}. \tag{20}$$

We use sampling with replacement, so the probability of not picking an observation in a random draw N times is then given by

$$(\frac{N-1}{N})^N. \tag{21}$$

Which in limit of large N is equal to

$$\lim_{n\to\infty}(1-\frac{1}{N})^N = e^{-1} = 0.368. \tag{22}$$

We conclude that when N becomes large, the probability of not picking an observation that is already in the bootstrap sample is 36.8%. Consequently, an out-of-bag sample consists approximately of 36.8% of the training data and the corresponding bootstrap sample roughly consists of 63.2% of the original training data, as N becomes large.

B Replication

B.1 Data

Regarding the replication of Huang et al. (2022), we have four target variables. Namely: U.S. Inflation, industrial production growth, change in unemployment rate, and S&P-500 volatility. We obtain the data on U.S. inflation and unemployment rate through the FRED-MD website. These variables correspond with the following names on the site: *CPIAUCSL* and *UNRATE*, respectively. Moreover, we obtain the industry production levels from the data released with the paper by Huang et al. (2022). Furthermore, we are not able to retrieve the data used by Huang et al. (2022) for the S&P-500 volatility because the origins of that data are undisclosed. The Erasmus Data Service Centre could not help us in our search for the correct data either, so we decide not to include the S&P-500 volatility target variable in our replication.

Additionally, the time series *CPIAUCSL* and *UNRATE* of the FRED-MD website differ from the time series used by Huang et al. (2022). The value difference starts from January 2016 up until and including December 2019. We believe this is due to an adjustment of the variables within the database that happened after the start of the research by Huang et al. (2022). This could lead to minor differences concerning *CPIAUCSL* and *UNRATE* in the replication results shown in Section 4.

Finally, we perform a data transformation on our remaining three target variables to obtain the required values of our target variables. The time series *CPIAUCSL*, *UNRATE*, and *IP levels* do not represent the required percentual changes, so we take the log difference of the variables as transformation, computed as

$$\Delta \log(X_t) = \log(X_t) - \log(X_{t-1}) \tag{23}$$

where X_t and X_{t-1} represents a target variable at periods t and t-1 respectively.

B.2 Results

In this Subsection, we present our replication results of the paper by Huang et al. (2022). In Table 8 we show the replication of Table 3 of the paper.

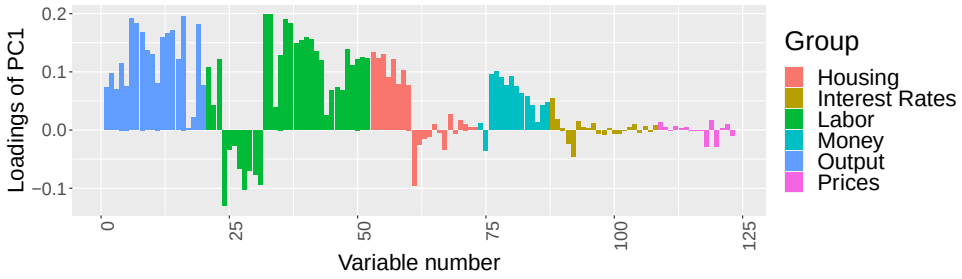
Table 8: Eigenvalues of the Covariance Matrixes of the Raw and Scaled Macro Variables.

		sPCA			
	PCA	Inflation	IP	Unemploy	Volatility(!)
1st	0.147	0.191	0.326	0.362(**)	0.22
2nd	0.074	0.143	0.105(*)	0.105	0.10
3rd	0.070	0.132	0.069	0.070	0.08
4th	0.054	0.085(*)	0.067	0.059	0.08
5th	0.043	0.063	0.060	0.042	0.06
6th	0.035(*)	0.033	0.032	0.030	0.04
7th	0.030	0.030	0.028	0.026	0.03
8th	0.024	0.025	0.020	0.022	0.03
9th	0.021	0.024	0.018	0.020	0.03
10th	0.020	0.020	0.018	0.019	0.02
11th	0.020	0.018	0.017	0.017	0.02
12th	0.020	0.015	0.015	0.016	0.02
13th	0.017	0.015(*)	0.014	0.014	0.02
14th	0.017	0.012	0.014	0.013	0.02
15th	0.016	0.011	0.014	0.012	0.02

Note: When the eigenvalues are normalized to have sum of one, this table also reports the 1st to 5th eigenvalues in descending order for the covariance matrixes of the (scaled) macro variables, where the scaling parameter is the predictive slope of the variable on the forecasted target (as described in Section 3.2). Moreover, (!) indicates that the values of this column are copied from Huang et al. (2022).
(*): When rounding these values to two decimals instead of three, the replicated values correspond with Huang et al. (2022).
(**): This value differs 0.029 from the value of the paper.

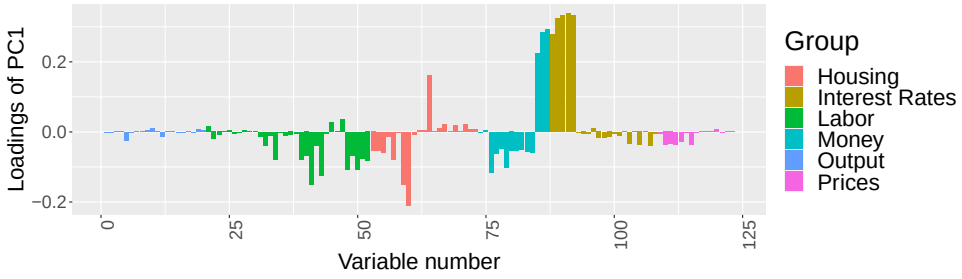
We observe that nearly all the values match the results of the paper; however, there are some exceptions. In our Table, we round to 3 decimals instead of the two decimals used in Huang et al. (2022). Consequently, the values in the Table with an asterisk ($*$) seem to differ from the results of the paper, but this is not the case. In addition, the first eigenvalue of sPCA Unemploy column has two asterisks ($**$) since it differs by 0.029 from the paper. Section B.1 mentions that the time series of $UNRATE$, which represents the unemployment rate, differs slightly from the time series used by Huang et al. (2022). This could explain the difference in results concerning the first eigenvalue in the sPCA Unemploy column. Next, we observe the loadings of our PCA and sPCA factors shown in Figures 10 through 13. We start by looking at Figure 10, which shows the loadings of the first PCA factor.

Figure 10: Loadings of the first PCA factor



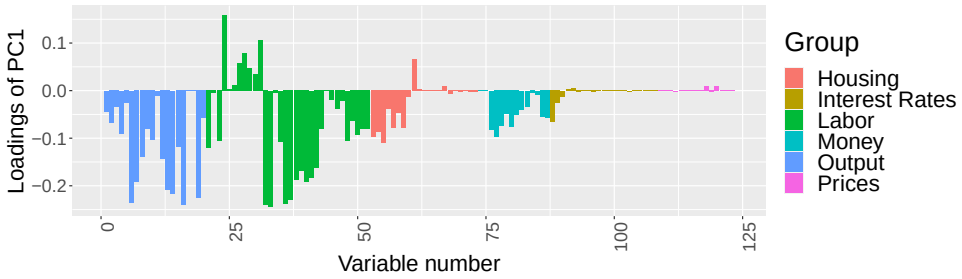
We see similar results as Huang et al. (2022). However, the sign of our loadings appears to differ (the absolute value is the same, but the non-absolute value differs). More specifically, the loadings are multiplied by -1. This does not matter since the sign of the loading does not change the interpretation of the contribution of a variable with regards to the principal component (Peres-Neto et al., 2003). It only changes the direction of the components in the PC feature space. Next, we look at Figure 11, which shows the loadings of the first sPCA factor concerning inflation.

Figure 11: Loadings of the first sPCA factor with regard to inflation



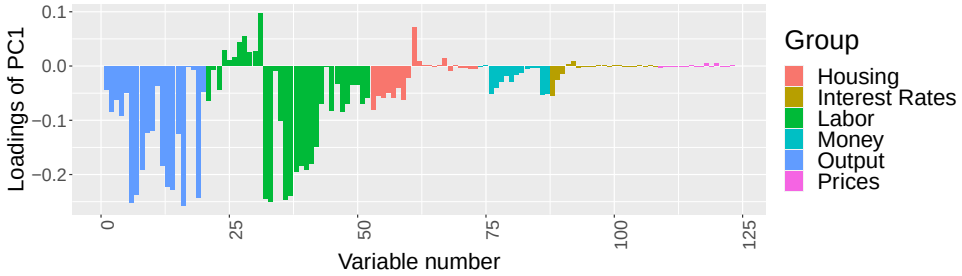
The loadings of this factor are similar to the loadings shown by Huang et al. (2022), in the sense that the same type and same variables contribute significantly to the PC. However, there still is a difference in signs of the loadings when looking at the loadings of labor-related, housing-related, and price-related variables. But, as mentioned before, this does not influence the interpretation and meaning of the PC. This sign could also result from using a more recent version of the data regarding inflation, as opposed to Huang et al. (2022). Next, we look at Figure 12, which shows the loadings of the first sPCA factor with regard to industrial production growth. We see that the same variables contribute significantly to the sPCA factor concerning

Figure 12: Loadings of the first sPCA factor with regard to industrial production growth



industrial production growth. However, we still see that the sign of these loadings differs from the results of Huang et al. (2022). Lastly, we look at Figure 13 that shows the loadings of the sPCA factor with regard to unemployment rate change.

Figure 13: Loadings of the first sPCA factor with regard to unemployment rate change



We see that the loadings of the first sPCA factor, with regard to unemployment rate change, are similar to the loadings of the factor reported by Huang et al. (2022). This time the factor loadings do not differ that much reported by Huang et al. (2022). The only difference is that the signs of the 4th through 9th labor variables are different, but their contribution is similar to the results of Huang et al. (2022).