

# Empirical investigation on supervised machine learning models predicting equity risk premium

Myong Jong Shin<sup>\*</sup>

December 9, 2022

## Abstract

We examine the predictive performance of supervised machine learning models in forecasting multi-horizon firm-level equity risk premium. We use an extensive collection of individual firms' financial characteristics and US macroeconomic return predictors for data. We forecast excess returns for (1) all individual firms and (2) each group of firms belonging to the same industry sector in the US. We first show an out-of-sample fit for each forecast model. Second, we forecast and evaluate models pairwise to find ones with superior predictive ability. We also estimate model confidence sets collecting models with superior predictive ability. Finally, we test for a model's conditional superior predictive ability, where a model's predictive ability is determined conditionally on a priori chosen variable indicative of the state of the market.

**Keywords:** Big Data, Supervised Machine Learning, Return Predictability, Forecast Evaluation

**JEL Codes:** C52, C55, C58, G17

---

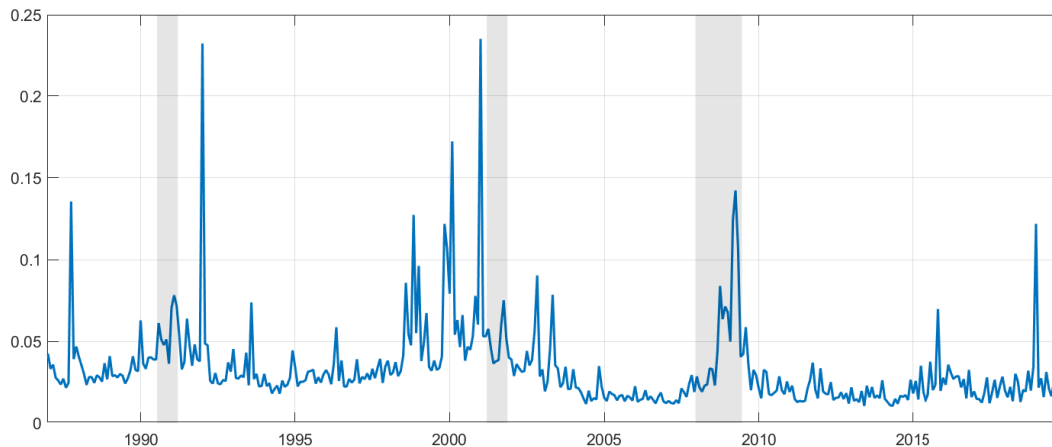
<sup>\*</sup>Department of Economics, Indiana University Bloomington, 100 S Woodlawn Ave, Bloomington, IN 47405. Email: [myonshin@iu.edu](mailto:myonshin@iu.edu)

# 1 Introduction

In the empirical finance literature, predicting equity risk premium, as stock returns above or below the risk-free rate, is a critical and popular research topic. In this article, we conduct a comparative analysis of supervised machine learning models forecasting multi-horizon equity risk premiums for individual firms in the US stock market. The goal is to identify a model or a set of models with better return predictability over others with statistical significance. We do so by employing a set of out-of-sample forecast evaluation tests from highly influential papers in recent years. We study (1) all individual firms collected and (2) subgroups of firms belonging to the same industry sector in the US.

Predicting excess returns is a challenging task. The field has seen many forecasting methods, a combination of forecasting models and predictors, reported having good return predictability. However, they are often sensitive to the choice of periods of the sample. Therefore a forecasting method’s predictability is often time-varying. For example, Figure 1 shows the squared forecast error from predicting monthly excess stock returns with a naive forecast of zeros, henceforth called 0forecast, from January 1987 to December 2019. To forecast a firm  $i$ ’s excess returns, the prediction of 0forecast is  $\hat{r}_{it} = 0, \forall t$ .

Figure 1: Forecast errors from 0forecast model.



Note: Using monthly stock returns for individual firms listed on NYSE, AMEX, and NASDAQ, we calculate squared forecast errors from a naive forecast of zeros. Forecast errors are cross-sectionally averaged for each month from January 1987 to December 2019.

Forecast errors from 0forecast in Figure 1 are significantly large over certain historical events in the US financial market, such as the stock market crash of 1987, the dot com bubble crash of the late 1990s, the

great recession of 2008, and many more. We can expect that under specific periods there will be models that can better forecast excess returns.

Given the noisy nature of the financial markets, we do not expect to find only one model or a fixed set of models that exhibit consistent return predictability across time and are robust to different choices of firms or financial markets. Thus, frequent performance evaluation of the forecasting methods is essential to find our bearings and to adapt our tools for future research in forecasting excess returns.

In their influential paper, [Gu, Kelly, and Xiu \(2020\)](#), henceforth GKX, surveys a set of supervised machine learning models to forecast monthly equity risk premiums for individual firms in the US stock market. Their data is extensive, with a collection of monthly stocks from March 1957 to December 2016. We are highly motivated by their research and contribute in the following ways.

First, we extend their analysis to forecast different multi-horizon returns. In this article, we study monthly, quarterly, semi-annual, and annual returns. The list of good models for predictability is different for each return. Second, we conduct forecast evaluation tests to determine superior predictive ability (SPA) and conditional superior predictive ability (CSPA) for a model and construct model confidence sets. Finally, we check return predictability for not only for the entire set of firms in our data but for their subsets grouped by their respective industry sectors in the US economy. We find models with SPA and CSPA for each industry sector. We find that for some industries, the models with good predictability differ from those for the entire market.

We do not claim that the set of supervised machine learning models used in our research represents a complete list of models employed by financial econometricians for prediction. The main reason behind our choices is that we selected the models used for the forecasting exercise in GKX. Some models are used in GKX but not in our article due to a lack of high-performance computation resources currently available to us. Such models include the grouped LASSO model and feed-forward neural networks. Grouped LASSO uses splines of the predictors of order 1, which are linear splines or higher. With the basis expansion, we model select using the absolute penalty function. Feed-forward neural networks are the simplest form of architecture for neural network models. Its simplicity is often helpful for prediction involving noisy data.

The rest of the paper is organized as follows. Section 2 describes the supervised machine learning models used in our article. Section 3 describes the out-of-sample forecast evaluation tests for SPA and CSPA. Section 4 describes the data used. Section 5 shows the results, and Section 6 concludes.

## 2 Supervised Machine Learning Models

In general terminology used in machine learning literature, prediction problems are divided into unsupervised learning problems and supervised learning problems. Unsupervised learning problems are situations where only the predictors are observable, whereas the latter is a case where we observe both the predictors and the outcome. Thus all of the models in our study are supervised machine learning models. We use them to run regression in Equation 1. In this study, we provide only the general description of each model and its tuning parameters if needed. There are many good resources for an in-depth description of the models' theory and their application to forecasting. For economists using machine learning models for a prediction problem, [Mullainathan and Spiess \(2017\)](#), and [Athey and Imbens \(2019\)](#) provide excellent recent reviews of machine learning methods for economists. Books such as [James, Witten, Hastie, and Tibshirani \(2013\)](#) and [Efron and Hastie \(2021\)](#) also discuss machine learning methods in the context of statistics and computer science.

Individual firm's  $h$ -horizon excess return is calculated as continuously compounded stock returns minus the risk-free rate written as  $\sum_{\tau=0}^{h-1} r_{i,t+\tau}$  where  $h$  indicates the multi-horizon dimension in the number of months. In our study, we forecast for one month ahead ( $h = 1$ ), one quarter ahead ( $h = 3$ ), six months ahead ( $h = 6$ ), and one year ahead ( $h = 12$ ) excess returns. The firms are indexed by  $i$ , and the total number of firms per month  $N_t$  changes for each month  $t = 1, \dots, T$ . All models use a common matrix of covariates  $z_{i,t-1}$ .  $\theta$  is a coefficient vector of size  $P$ . Let  $\mathbb{J}$  be the set of supervised machine models used in this article. All our models are additive prediction error regression models for firm  $i$ 's return written in Equation 1. We forecast returns by the estimated conditional expectation of the regression.

$$\sum_{\tau=0}^{h-1} r_{i,t+\tau} = f_j(z_{i,t-1}; \theta) + \sum_{\tau=0}^{h-1} \epsilon_{i,t+\tau}, \quad j \in \mathbb{J} \quad (1)$$

We denote the different supervised machine learning models by subscript  $j \in \mathbb{J}$ . A firm  $i$ 's  $h$ -horizon excess returns are predicted as  $f_j(z_{i,t-1}, \hat{\theta})$  for model  $j$  as the minimizer of an objective function such as

least squares. For example, when using a partial least squares model for prediction, we denote its forecast as  $f_{pls}(z_{i,t-1}, \hat{\theta})$ . The 0forecast model for forecasting multi-horizon excess returns is  $\sum_{\tau=0}^{h-1} r_{i,t+\tau} = 0$  requiring no estimation, and is denoted with  $f_0$ .

For our out-of-sample scheme, we split our full sample into three mutually exclusive samples by time. First, the training sample, denoted as  $\mathbb{T}_1$ , is used to estimate the model. Second, the validation sample, denoted as  $\mathbb{T}_2$ , is used to choose hyperparameters of the model if needed. With  $\mathbb{T}_1$  and  $\mathbb{T}_2$ , we can make predictions. Lastly, the testing sample, denoted as  $\mathbb{T}_3$ , is used to compare the predictions with the actual excess returns and compute squared forecast errors.

## 2.1 Principal Component Regression

Principal component regression(PCR) uses principal component(PC)  $\omega_k$ . PCs are extracted from the sample covariance matrix of predictors  $Z$ . The solution to the optimization problem in Equation 2 is the eigenvector associated with  $k$ th largest eigenvalue.

$$\begin{aligned} \omega_k &= \arg \max_{\omega} Var(Z\omega), \\ \text{subject to } \omega' \omega &= 1, Cov(Z\omega, Z\omega_k) = 0, \\ k &= 1, 2, \dots, P-1. \end{aligned} \tag{2}$$

For each estimation using PCs from training data, we can use a scree plot of eigenvalues to pick the number of PCs. By using fewer PCs, we can reduce the dimension and improve estimation. Alternatively, we can use the validation data and choose the number of PCs with the smallest mean squared error during the validation period. Both approaches provide a similar number of PCs, and the chosen number of components is used for extracting PCs for prediction during the testing period of our sample. For our data, many PCs are usually needed to explain more than 80% of the covariance matrix. However, using too many PCs erodes the benefit of dimension reduction and makes predictions worse. For our dataset, we find that PCs explaining 60% of the variance of covariates or lower provide good forecasts.

## 2.2 Partial Least Squares Regression

Partial least squares regression(PLS) extracts PLS components  $\omega_k$  by solving Equation 3. Whereas PCR extracts linear combinations of predictors that best explain the covariance matrix of predictors,

PLS extracts linear combinations of predictors that best explain the squared covariance between the multi-horizon excess returns and the matrix of covariates  $Z$ .

$$\begin{aligned} \omega_k &= \arg \max_{\omega} Cov^2\left(\sum_{\tau=0}^{h-1} r_{t+\tau}, Z\omega\right), \\ \text{subject to } \omega' \omega &= 1, \quad Cov(Z\omega, Z\omega_k) = 0 \\ k &= 1, 2, \dots, P-1. \end{aligned} \tag{3}$$

To solve Equation 3, we use the SIMPLS algorithm by [De Jong \(1993\)](#). We use the validation data and choose the number of PLS components with the smallest mean squared error during the validation period. The chosen number of components is then used for prediction during testing.

For both PCR and PLS, the least squares objective function is used to estimate the coefficient vector  $\theta$ , and we make prediction  $f(z_{i,t-1}, \hat{\theta})$ . Equation 4 shows the resulting pooled least squares estimator.

$$\hat{\theta} = \arg \min_{\theta} \sum_{i=1}^{N_t} \sum_{t=1}^T \left( \sum_{\tau=0}^{h-1} r_{i,t+\tau} - f_j(z_{i,t-1}; \theta) \right)^2 \tag{4}$$

## 2.3 LASSO

LASSO stands for least absolute shrinkage and selection operator. We include a penalty function to the least squares objective function for model estimation to find the unique optimizer  $\theta$ . There are many variants of the LASSO with different penalty functions. [Tibshirani \(2011\)](#) provides an excellent review of the different generalizations of the LASSO method. For our research, we use a simple absolute penalty on each slope coefficient  $\theta_k$ , and our LASSO problem can be stated as Equation 5 in lagrangian form with  $\lambda \geq 0$ .

$$\hat{\theta}_{lasso} = \arg \min_{\theta} \left( \sum_{i=1}^{N_t} \sum_{t=1}^T \left( \sum_{\tau=0}^{h-1} r_{i,t+\tau} - f_j(z_{i,t-1}; \theta) \right)^2 + \lambda \sum_{k=1}^p |\theta_k| \right) \tag{5}$$

By constraining the coefficient vector  $\theta$  by  $l1$  norm, LASSO induces *sparsity* where we can have some  $\hat{\theta}_k = 0$ . The optimal tuning parameter  $\lambda$  that controls the amount of regularization needs to be searched over a grid of candidate values. For each estimation of the model using training data, we use the validation data to find an optimal value for  $\lambda$ , and the chosen value is used for forecasts during the testing period.

## 2.4 Ridge

Ridge model includes a square penalty to the least squared objective function that constrains the magnitude of  $\theta_k$ . The optimization problem can be stated as Equation 6 in lagrangian form with  $\lambda \geq 0$ .

$$\hat{\theta}_{ridge} = \arg \min_{\theta} \left( \sum_{i=1}^{N_t} \sum_{t=1}^T \left( \sum_{\tau=0}^{h-1} r_{i,t+\tau} - f_j(z_{i,t-1}; \theta) \right)^2 + \frac{1}{2} \lambda \sum_{k=1}^p \theta_k^2 \right) \quad (6)$$

In contrast to *variable selection* in LASSO, the Ridge induces *shrinkage* of  $\hat{\theta}_k$  near zero. Thus Ridge constrains the magnitude of  $\theta$  from being too large. The optimal tuning parameter  $\lambda$  that controls the regularization amount is determined similarly to LASSO.

## 2.5 Elastic Net

The elastic net model from [Zou and Hastie \(2005\)](#) utilizes the penalty functions of both LASSO and Ridge as their convex combination. Thus the model estimates  $\theta$  through both *variable selection* and *shrinkage*. For our study, the convex combination parameter  $\rho$  is fixed to be 0.5, and the optimal tuning parameter  $\lambda$  is selected using validation data.

$$\hat{\theta}_{enet} = \arg \min_{\theta} \left( \sum_{i=1}^{N_t} \sum_{t=1}^T \left( \sum_{\tau=0}^{h-1} r_{i,t+\tau} - f_j(z_{i,t-1}; \theta) \right)^2 + \lambda(1 - \rho) \sum_{k=1}^p |\theta_k| + \frac{1}{2} \lambda \rho \sum_{k=1}^p \theta_k^2 \right) \quad (7)$$

## 2.6 Random Forest

Random forest is a bootstrap aggregation or ‘bagging’ ([Breiman \(2001\)](#)) of individual regression trees. Random forest predicts using an ensemble of  $B$  number of bootstrapped trees. Depending on the prediction exercise at hand, there are many ways to grow the random forest. For our article, we use the algorithm by [Hastie, Tibshirani, Friedman, and Friedman \(2009\)](#) for growing a random forest. First, using resampled training sample  $\{z_{i,t-1}^b, \sum_{\tau=0}^{h-1} r_{i,t+\tau}^b\}, b = 1, \dots, B$ , each regression tree  $b$  grows branches in top-to-bottom fashion. A branch separates the data from the previous branch using one of the predictor variables. Its goal is to gather observations similar to each other and place them in two disjoint groups of branches. At branch  $C$ , a tree chooses a predictor in  $z_{i,t}^b$  that can split the data that minimize  $l_2$  impurity in Equation 8 where  $|C|$  denotes the number of observations at branch  $C$ .

$$H(\theta, C) = \frac{1}{|C|} \sum_{z_{i,t-1}^b \in C} \left( \sum_{\tau=0}^{h-1} r_{i,t+\tau}^b - \theta \right)^2 \quad (8)$$

We stop splitting branches when the maximum depth of a tree  $L$  is reached. With  $2^L$  number of leaves (terminal nodes), we use 300 regression trees, and each tree has a maximum depth of  $L = 6$ . The prediction of a tree using  $b$ th bootstrap sample is  $f_{tree,b}(z_{i,t-1}^b, \hat{\theta}^b, L)$  with estimated slope  $\hat{\theta}_k^b$  equal to the historical average of returns belonging to the same branch.

$$\hat{\theta}_k^b = \frac{1}{|C_k(L)|} \sum_{z_{i,t-1}^b \in C_k(L)} \sum_{\tau=0}^{h-1} r_{i,t+\tau}^b$$

$$f_{tree,b}(z_{i,t-1}^b, \hat{\theta}^b, L) = \sum_{k=1}^{2^L} \hat{\theta}_k^b \mathbb{1}\{z_{i,t-1}^b \in C_k(L)\}$$

The final random forest prediction bags predictions from individual trees as their average in Equation 9.

$$f_{rf} = \frac{1}{B} \sum_{b=1}^B f_{tree,b}(z_{i,t-1}^b, \hat{\theta}^b, L) \quad (9)$$

### 3 Forecast evaluation

Here, we discuss the statistical tests used to evaluate the performance of forecast models. An out-of-sample fit is a common statistic used in machine learning literature to evaluate forecasts from a model. It is convenient to calculate and is widely used by financial economists, statisticians, and computer science researchers when analyzing a model's predictive performance. However, with the tests used in our article, researchers can make better comparisons across models and avoid errors in multiple hypothesis testing problems. Moreover, the test for CSPA introduces a new way to evaluate forecasts conditional on the state of the economy.

#### 3.1 $R_{OOS}^2$

$R_{OOS}^2$  statistic is the out-of-sample fit during the testing sample period  $\mathbb{T}_3$  for each model. First, we pool forecast errors across time and firms to calculate the mean squared forecast error from model  $j$  and the 0forecast. Then we measure the reduction in mean squared forecast error using model  $j$  relative to 0forecast. Outperforming it on average is shown by a positive  $R_{OOS}^2$  value, and a negative value indicates the relative underperformance of model  $j$ .

$$R_{OOS,j}^2 = 1 - \frac{\sum_{(i,t) \in \mathbb{T}_3} \left( \sum_{\tau=0}^{h-1} r_{i,t+\tau} - f_j(z_{i,t-1}, \hat{\theta}) \right)^2}{\sum_{(i,t) \in \mathbb{T}_3} \left( \sum_{\tau=0}^{h-1} r_{i,t+\tau} \right)^2} \quad (10)$$



We use the definition of  $R_{\text{OOS}}^2$  from GKX. This formula differs from the conventional one for an out-of-sample fit that uses historical mean forecast rather than the 0forecast. The reasons for this choice are twofold. First, this formula for  $R_{\text{OOS}}^2$  provides information regarding a model's out-of-sample fit relative to a model used in our study. Thus the interpretation of the statistic is more meaningful. Second, outperforming historical mean forecast is typically a less challenging task for the models studied here due to the noisy nature of the firm-level return data.

### 3.2 Test for Superior Predictive Ability and Model Confidence Set

We use squared forecast error to measure a prediction's accuracy for all our models<sup>1</sup>. Given  $h$  at time  $t$ , we have  $N_t$  number of errors per model. Let  $\mathbb{L}_{j,t+h-1}$  be the cross-sectional average of squared error from model  $j$  for all firms.

$$\mathbb{L}_{j,t+h-1} = \frac{1}{N_t} \sum_{i=1}^{N_t} [(f_j(z_{i,t-1}, \hat{\theta}) - \sum_{\tau=0}^{h-1} r_{i,t+\tau})^2] \quad (11)$$

We say that the model  $j$  has superior predictive ability (SPA) over model  $j'$  when the squared forecast error of model  $j$  is smaller than model  $j'$  on average.

$$\mathbb{E}(\Delta \mathbb{L}_{j,j',t+h-1}) \leq 0 \quad (12)$$

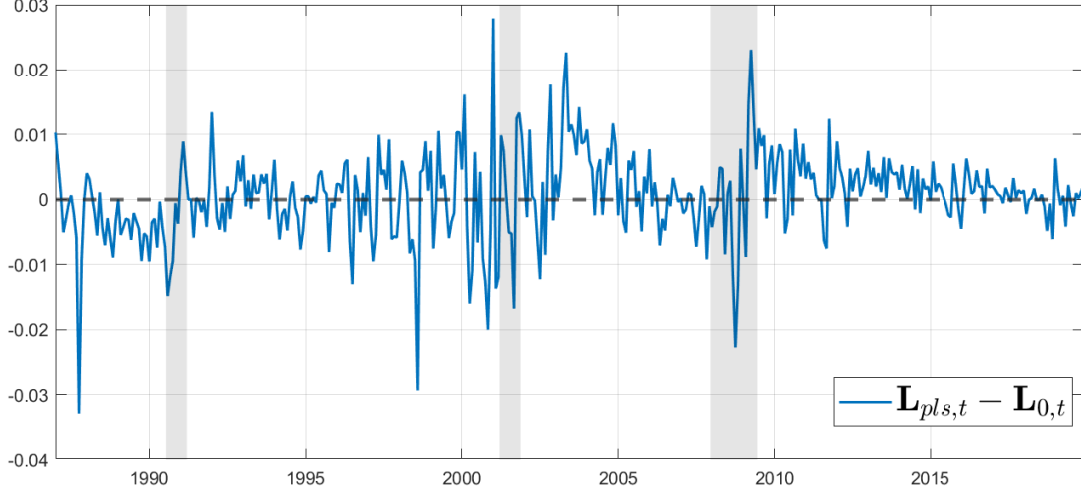
We denote  $\mathbb{L}_{j,t+h-1} - \mathbb{L}_{j',t+h-1}$  as  $\Delta \mathbb{L}_{j,j',t+h-1}$  for simplicity and use them interchangeably. For example, Figure 2 plots the difference between a partial least square model's squared forecast error and that of the 0forecast for monthly forecasts,  $\mathbb{L}_{pls,t} - \mathbb{L}_{0forecast,t}$ . During periods where  $\Delta \mathbb{L}_{pls,0forecast,t} < 0$  on Figure 2, the partial least squares model has a smaller squared error, thus outperforming 0forecast. If  $\mathbb{E} \Delta \mathbb{L}_{pls,0forecast,t} < 0$ , we say that partial least squares has SPA over 0forecast.

Prior to our tests, we checked for the existence of unit-root for the time series  $\Delta \mathbb{L}_{j,j',t+h-1}$  for all pairs of models, as all the tests in this article assume stationary. This does not mean that the data and forecasts need to be stationary. Also, as all our models are additive prediction error regressions in the form of Equation 1, the error terms are additive. We consider the serial correlation of the errors for all our tests using Newey-West standard errors with lag  $h - 1$ .

---

<sup>1</sup>A popular alternative to squared loss is the Huber loss function. It is a hybrid of squared loss for small errors and absolute loss for relatively large errors, thus penalizing them less.

Figure 2:  $\Delta \mathbb{L}_{pls,0forecast,t}$ , January 1987 to December 2019.



Note: Using monthly stock returns for individual firms listed on NYSE, AMEX, and NASDAQ, we calculate squared forecast errors from a partial least squares model and 0-forecast model. Forecast errors are cross-sectionally averaged for each month, and we take the average difference between the two models from January 1987 to December 2019.

The main research questions regarding SPA are twofold. First, for a pair of models  $(j, j') \in \mathbb{J}$ , we test significance for all model pairs under null of  $\mathbb{E}(\Delta \mathbb{L}_{j,j',t+h-1}) \leq 0$ . We test this null using the test of SPA by Hansen (2005). Second, we construct a model confidence set(MCS) from Hansen, Lunde, and Nason (2011) that contains models with SPA over all other models with a chosen coverage probability.

To test the null of superior predictive ability(SPA) of model  $j$  over  $j'$  for all pairwise combinations of models, we use the test for SPA by Hansen (2005). For the difference of mean squared forecast error between model  $j$  and  $j'$  during testing period  $\bar{d}_{j,j',t+h-1} = 1/\mathbb{T}_3 \sum \Delta \mathbb{L}_{j,j',t+h-1}$ , we describe the testing procedure.

1. Studentize the test statistic with Newey-West standard error with lag  $h - 1$  and normalized to 0.

$$T_{SPA} = \max \left[ \frac{\sqrt{\mathbb{T}_3} \bar{d}_{j,j',t+h-1}}{\sqrt{\text{var}(\sqrt{\mathbb{T}_3} \bar{d}_{j,j',t+h-1})}}, 0 \right]$$

2. Bootstrap the studentized test statistic but recenter it to  $\hat{\mu}^c$  where

$$\begin{cases} \hat{\mu}^c = \bar{d}_{j,j',t+h-1}, & \text{if } \frac{\sqrt{\mathbb{T}_3} \bar{d}_{j,j',t+h-1}}{\sqrt{\text{var}(\sqrt{\mathbb{T}_3} \bar{d}_{j,j',t+h-1})}} \leq -\sqrt{2 \log \log \mathbb{T}_3} \\ \hat{\mu}^c = 0, & \text{otherwise} \end{cases}$$

When model  $j$  has the better sample performance, the test statistic is normalized to 0, and we conclude we find no evidence against the null hypothesis, and consequently the null of  $H_0 : \mathbb{E}(\Delta \mathbb{L}_{j,j',t+h-1}) \leq 0$  should not be rejected. The resampled test statistic is recentered using the law of iterated logarithms to generate the bootstrap distribution that conforms with the null hypothesis. We use the stationary bootstrap of [Politis and Romano \(1994\)](#) for implementation.

Next, we construct a model confidence set(MCS) from [Hansen, Lunde, and Nason \(2011\)](#) with a 5% significance level to collect a set of models with SPA. With MCS, we can conduct *all-for-one comparison* of models, where we search for models with better forecast performance over all other models on average jointly. To define MCS, we assume that  $\mu_{j,j',h-1}$ , the unconditional expectation of the difference of mean squared forecast error between model  $j$  and  $j'$  exists and is finite.

$$\mu_{j,j',h-1} = \mathbb{E} \Delta \mathbb{L}_{j,j',t+h-1}, \quad \forall j, j' \in \mathbb{J} \quad (13)$$

For all model pairs, the set of superior models  $\mathbb{M}^*$  is

$$\mathbb{M}^* = \{j \in \mathbb{J} : \mu_{j,j',h-1} \leq 0, \forall j' \in \mathbb{J}\} \quad (14)$$

Using the MCS algorithm in [Hansen, Lunde, and Nason \(2011\)](#), we construct the set of superior models and use the stationary bootstrap for p-values.

### 3.3 Conditional Superior Predictive Ability

We test for CSPA of a model using the testing procedure from [Li, Liao, and Quaadvlieg \(2022\)](#). Testing for CSPA of a model means that we evaluate the forecast model with another for their relative predictive ability conditional on some chosen state variable. An unconditional comparison of the average performance we made for testing SPA and constructing MCS can be considered a special case of testing for CSPA, given a pair of models or a group of multiple models. Because we only compare the average rela-

tive performance of models for SPA and MCS, we integrate out a possible heterogeneity across important periods during the testing sample in the data, such as recession periods. The purpose of CSPA is to discover the state-dependent relative performance among forecast models.

The null hypothesis of CSPA in our article is that the conditional expected squared error of a model  $j$  is no larger than that of all other competing models uniformly across all conditioning states, thus model  $j$  outperforms all competitions uniformly<sup>2</sup>. Given a forecast model  $j \in \mathbb{J}$ , we test the null of CSPA of the model  $j$  over all  $j' \in \mathbb{J}$ , given a priori chosen state variable  $X_t$ . Note that by the law of iterated expectations, the null of CSPA implies the null of model  $j$ 's SPA over all others.

$$H_0 : \mathbb{E}(\Delta_{j,j',t+h-1} | X_t = x_t) \leq 0, \forall j' \in \mathbb{J}, X_t \in \mathbb{X} \quad (15)$$

$\mathbb{X}$  is the support of the state variable. The conditioning state must be a scalar for implementing the test. Table 1 lists the different state variables used in our article and their sources in parenthesis. We report CSPA results using all the state variables for robustness. Our article uses business cycle indicators, financial conditions or stress indicators, and uncertainty indices.

Table 1: List of state variables.

Mnemonic	Description
Business cycle/economic activity indicators	
ADS	Aruoba-Diebold-Scotti index(Philadelphia Fed)
rec_prob	Smoothed U.S. Recession Probabilities(St Louis FRED)
gecon	Global Economic condition indicator ( <a href="#">Baumeister, Korobilis, and Lee (2022)</a> )
Financial conditions/stress indicators	
NFCI	National financial conditions index(Chicago Fed)
ANFCI	Adjusted national financial conditions index(Chicago Fed)
Uncertainty indices	
vxo	Volatility index(CBOE)
mpu	Monetary Policy Uncertainty index ( <a href="#">Husted, Rogers, and Sun (2020)</a> )
epu	Economic Policy Uncertainty(USA); news based index ( <a href="#">Baker, Bloom, and Davis (2016)</a> )
JLN_f	Financial Uncertainty ( <a href="#">Jurado, Ludvigson, and Ng (2015)</a> )
JLN_m	Macroeconomic Uncertainty ( <a href="#">Jurado, Ludvigson, and Ng (2015)</a> )
JLN_r	Real Uncertainty ( <a href="#">Jurado, Ludvigson, and Ng (2015)</a> )

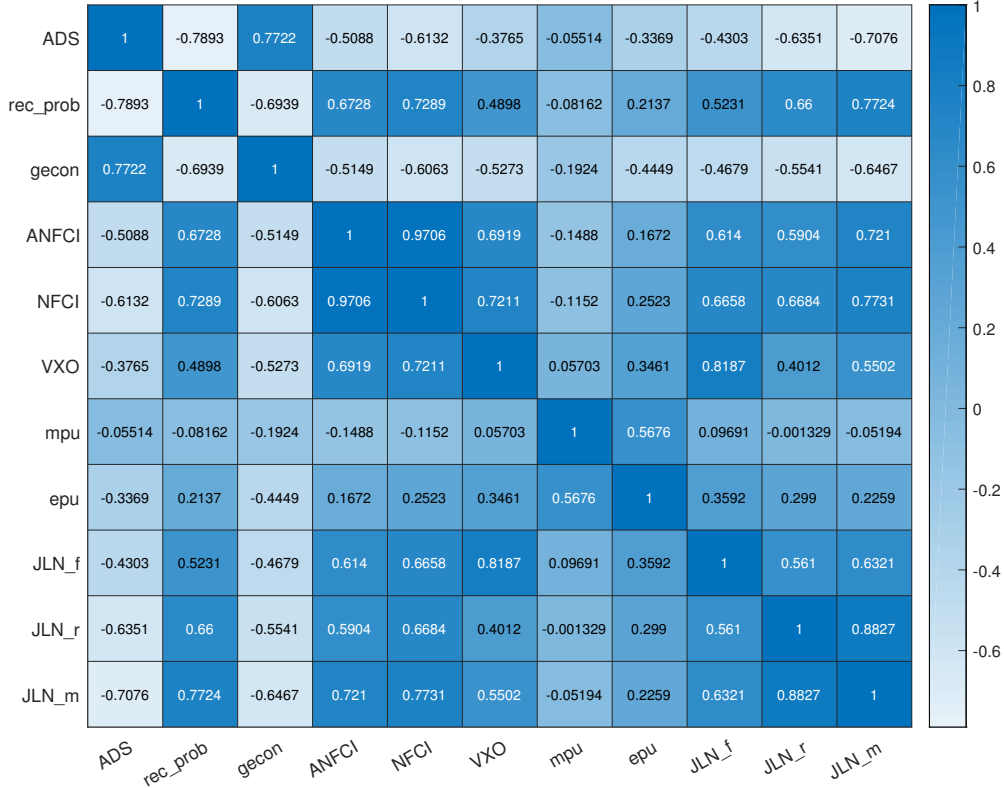
Note: We list all 11 scalar state variable used for testing CSPA from January 1987 to December 2019. JLN for h=6 is not available.

We also display the heatmap of correlation across all the scalar state variable used for testing CSPA from January 1987 to December 2019. For semi-annual return forecasts, uncertainty indices for [Jurado](#),

<sup>2</sup>Unlike the original null hypothesis of CSPA stated in [Li, Liao, and Quaedvlieg \(2022\)](#), the direction of the inequality sign in our article is switched to be coherent with our analysis of testing SPA and constructing MCS.

Ludvigson, and Ng (2015) are not available. We find that some state variables have high correlation with each other. However, this does not affect our test of CSPA as each state variable is used separately as a scalar. Additionally, we argue that a state variable that has strong linear relationship with another is not equivalent to it being redundant, and it may contain information useful for our forecast evaluation.

Figure 3: Correlation among state variables.



Note: We list correlation across all the scalar state variable used for testing CSPA from January 1987 to December 2019. JLN for h=6 is not available.

Our purpose in using CSPA is to find evidence of a uniform conditional dominance of a forecast model. Suppose a model has CSPA over different state variables. In that case, we consider it to be a strong indication that it has superior predictive ability over all other models, and its predictive ability is robust to different states of the economy.

To implement the test, the conditional expectation function  $\mathbb{E}(\Delta_{j,j',t+h-1}|X_t = x_t)$  is approximated via a nonparametric series regression of  $\Delta_{j,j',t+h-1}$  on the basis expansion of the state variable  $X_t$  using Legendre polynomials. To calculate the correct critical value for the test, Li, Liao, and Quaadvlieg (2022) provides an algorithm for implementation. We take serial correlation into account for the test by using Newey-West standard errors.

## 4 Data

To forecast multi-horizon excess returns, we use an extensive collection of firm characteristics and macroeconomic predictors for the aggregate market return. The critical paper that serves as our article’s primary motivation for data construction is GKK. GKK use 94 firm characteristics from [Green, Hand, and Zhang \(2017\)](#), eight macroeconomic predictors from [Welch and Goyal \(2007\)](#), and 74 dummy variables to classify firms into their respective industry sectors. GKK defined industry sectors by a firm’s first two digits of the US Standard Industrial Classification (SIC) Codes. With the data from March 1957 to December 2016, GKK predicted individual monthly excess returns using supervised machine learning models, and checked each model’s fit out-of-sample. For our analysis, we use the data from GKK with more recent observations of predictors included up to December 2019<sup>3</sup>.

Our contribution is to further their work by extending the forecast exercise to predict multi-horizon excess returns and conduct tests in a search for models with SPA and CSPA. Moreover, we forecast multi-horizon excess returns by firm groups categorized by their industry sector. To do so, we do not include SIC dummies from GKK as predictors. We instead forecast excess returns per industry using firm SIC codes and compare the forecast model within. We use the first digit of each firm’s SIC code and categorize the industries in the US into ten different sectors. Table 2 lists the class of firms by their first digit of the SIC code into ten industry sectors. We use the same set of forecast models from studying the entire market to each industry and report the SPA and CSPA test results.

Table 2: Standard Industrial Classification Codes

First digit of SIC	Description
0	Agriculture, Forestry and Fishing
1	Mining and Construction
2,3	Manufacturing
4	Transportation, Communications, Electric, Gas and Sanitary service
5	Wholesale Trade and Retail Trade
6	Finance, Insurance and Real Estate
7,8	Services
9	Public Administration and Nonclassifiable

Let  $sic_k$  denote a set of firms whose first digit of SIC is  $k$ . For example, a set of firms belonging to

<sup>3</sup>The data for predictors in our article is provided by the authors and is readily available at their website but requires access to high-performance computation and ample storage to handle the big data and heavy computation burden.

*Real Estate industry* is labeled with  $sic_6$ . Given  $h$  at time  $t$ , we have  $N_{t,sic_6}$  number of firms in the *Real Estate industry*. For forecast model  $j$ , we can define the cross-sectional average of squared error for each industry  $sic_k$  as  $\mathbb{L}_{sic_k,j,t+h-1}$ . We use this to test for SPA and CSPA of models across industry.

$$\begin{aligned}\mathbb{L}_{sic_k,j,t+h-1} &= \frac{1}{N_{t,sic_k}} \sum_{i=1}^{N_t} \left[ \left( f_j(z_{i,t-1}, \hat{\theta}) - \sum_{\tau=0}^{h-1} r_{i,t+\tau} \right)^2 \cdot \mathbf{1}_{sic_k}(i) \right], \quad k = 0, 1, 2, \dots, 9. \\ N_{t,sic_k} &= \sum_{i=1}^{N_t} \mathbf{1}_{sic_k}(i), \quad N_t = \sum_{k=0}^9 N_{t,sic_k} \\ \mathbf{1}_{sic_k}(i) &= \begin{cases} 1 & , \text{firm } i\text{'s first digit of SIC is } k. \\ 0 & , \text{otherwise} \end{cases}\end{aligned}$$

Monthly firm equity returns are from CRSP for all firms listed in NYSE, AMEX, and NASDAQ. We collect our data from January 1960 to December 2019. Each firm's excess returns are calculated as continuously compounded stock returns minus the risk-free rate, and we use US treasury-bill rate to proxy the risk-free rate. Throughout the period, the number of firms listed per month changes, and Figure 4 displays the changes in the number of stocks in our sample. The proportion of each industry sector in our sample is shown alongside in the form of a pie chart. The proportion for each industry sector  $s$  is calculated as the pooled average across time and cross-section  $\frac{1}{T} \frac{1}{N} \sum_{t=1}^T \sum_{i=1}^N \mathbf{1}_{sic_k}(i)$ . The largest proportion of industry in our data is *Manufacturing* with SIC codes 2 and 3, and the smallest is *Agriculture, Forestry, and Fishing* with SIC code 0.

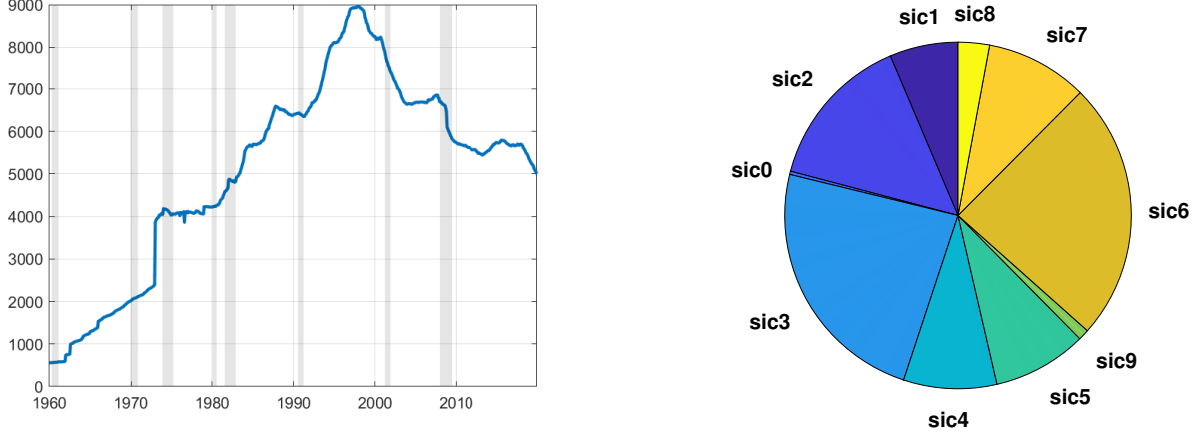
For predictors, we have two groups of covariates. First, we have 94 firm-level characteristics from GKX. Our data contains over 3 million pooled observations for each firm for its excess return and firm characteristics<sup>4</sup>. For firm  $i$  at time  $t$ , let  $c_{i,t}$  be a 94 by 1 vector of firm characteristics with  $i = 1, \dots, N_t$ ,  $t = 1, \dots, T$ .

Second, we have eight macroeconomic predictors from [Welch and Goyal \(2007\)](#) for market equity premium, and they consist of market stock characteristics and bond interest-related predictors. For our research, we use dividend-price ratio, earning-price ratio, book-to-market ratio, net equity expansion, stock variance, treasury-bill rate, term spread, and default spread. At time  $t$ , let  $m_t$  be an eight by one vector of macroeconomic predictors.

<sup>4</sup>Each firm characteristic is winsorized cross-sectionally for 1% and 99%. Refer to the Appendix of [Gu, Kelly, and Xiu \(2020\)](#) for a full description of the firm characteristics used.

Figure 4: Firms from January 1960 to December 2019

(a) Note: We plot  $N_t$ , the number of firms listed on NYSE, AMEX, and NASDAQ each month from January 1960 to December 2019. (b) We show proportion of firms belonging to 10 different industry sectors defined using the first digit of the SIC code. The proportion for each sector is an pooled average.



With  $c_{i,t}$  and  $m_t$ , we construct the vector of predictors  $z_{i,t}$ . Let  $P$  be the total number of predictors used and  $z_{i,t}$  be a  $P$  by one vector. For predictors, we use  $c_{i,t}$  and interaction terms between  $m_t$  and  $c_{i,t}$ .

$$\underbrace{z_{i,t}}_{P \times 1} = \underbrace{[1, m_t]'}_{9 \times 1} \otimes \underbrace{c_{i,t}}_{94 \times 1} \quad (16)$$

By appending a constant in front of vector  $m_t$  and using Kronecker product, we, therefore, have  $P$  equal to  $94 * (8 + 1) = 846$  including the number of interaction between  $c_{i,t}$  and  $m_t$ . With  $z_{i,t}$  and a model  $j$ , we formulate forecast models in the form of Equation 1. Aside from the random forest model, other supervised machine learning models in our research can be interpreted as a beta pricing model with  $\theta$  interpreted as quantity of risk in each individual stock.

For our out-of-sample analysis, we use the rolling scheme. In the beginning, we have the initial 15 years training period from January 1960 to December 1974, and 12 years of validation from January 1975 to December 1986. We then forecast excess returns for the entire year of 1987, from January to December. In the next step, we roll over our training and validation period for one year and set a new 15 years of training period from January 1961 to December 1975, and a new 12 years of validation from January 1976 to December 1987. We then make our forecast for the entire year of 1988. We repeat this



process until the end of the sample. In the end, we have 33 years of testing period from January 1987 to December 2019 that we can compare with observed excess returns and calculate forecast errors.

## 5 Results

First, we forecast multi-horizon returns for all firms in our data using supervised machine models described in Section 2. For returns of horizon  $h$ ,  $h = 1$  is monthly,  $h = 3$  is quarterly,  $h = 6$  is semi-annual, and  $h = 12$  is annual excess returns.

Table 3: All firms:  $R_{OOS}^2$ , MCS results

	h=1		h=3		h=6		h=12	
	r2	MCS	r2	MCS	r2	MCS	r2	MCS
0forecast	-	✓	-		-		-	
Enet	-0.0176		0.0140		0.0214		0.0257	
LASSO	-0.0177		0.0144		0.0217		0.0272	
PCR	-0.0176		0.0020		0.0056		0.0061	
PLS	-0.0171		0.0180		0.0260		0.0324	
Ridge	-0.0648		-0.0590		-0.0942		-0.0837	
RF	-0.0092		0.0351	✓	0.0377	✓	0.0397	✓

Note: Models in MCS is 5% marked with ✓, stationary bootstrap with B=100000 and block length=5000.

Table 3 shows  $R_{OOS}^2$  of each model and the models included in MCS with 5% significance. We observe that as the number of horizon  $h$  increases,  $R_{OOS}^2$  increases for most models. Aside from predicting monthly excess returns, our result shows that the random forest model shows better out-of-sample fit relative to others and is the only one included in the MCS. This suggests that when forecasting multi-horizon returns for all the firms collected in our data, a random forest model should be chosen from the models tested in our article.

The ridge model shows poor out-of-sample fit for all horizons, and we suspect this is due to the sheer number of covariates used that shrinkage alone could not handle. This is further supported by observing that their fit is very similar for the elastic net model and LASSO model, suggesting that elastic net primarily benefited from variable selection, reducing the number of parameters to estimate.

Table 4 displays test results for SPA between model pairs with 5% significance. For readability, we write the model's name with SPA over the other in the table. For example, at  $h = 3$ , between 0forecast and elastic net model, the latter has SPA. We can see that random forest has SPA over almost every model. Dimension reduction technique such as partial least squares and model selection via elastic net model or LASSO model also has SPA over 0forecast and some other models.

Table 4: All firms: SPA results(pairwise)

		0forecast	Enet	LASSO	PCR	PLS	Ridge	RF
h=1	0forecast	-						
	Enet	-	-					RF
	LASSO	-	-	-				RF
	PCR	-	-	-	-			RF
	PLS	-	-	-	-	-		RF
	Ridge	-	-	-	-	-	-	RF
h=3	0forecast	-	Enet	LASSO		PLS		RF
	Enet	-	-	LASSO		PLS		RF
	LASSO	-	-	-		PLS		RF
	PCR	-	-	-	-	PLS		RF
	PLS	-	-	-	-	-		RF
	Ridge	-	-	-	-	-	-	RF
h=6	0forecast	-	Enet	LASSO		PLS		RF
	Enet	-	-			PLS		RF
	LASSO	-	-	-		PLS		RF
	PCR	-	-	-	-	PLS		RF
	PLS	-	-	-	-	-		RF
	Ridge	-	-	-	-	-	-	RF
h=12	0forecast	-	Enet	LASSO		PLS		RF
	Enet	-	-			PLS		RF
	LASSO	-	-	-		PLS		RF
	PCR	-	-	-	-	PLS		RF
	PLS	-	-	-	-	-		RF
	Ridge	-	-	-	-	-	-	RF

Note: For a model  $j$ (in row) that has SPA over model  $j'$ (in column), its name is written.

Next, we report test results for a model's CSPA in Table 5. A  $(\star)$  indicates rejection of the null hypothesis of a model  $j$ 's CSPA over all other competing models uniformly given a conditioning state variable;  $\mathbb{E}(\Delta_{\mathbb{L}_{j,j',t+h-1}}|X_t = x_t) \leq 0, \forall j' \in \mathbb{J}, X_t \in \mathbb{X}$ . Names of the model  $j$  being tested are written on the left column, and the names of each state variable used per a test are written on the top row. For example, in  $h = 12$ , the random forest model has CSPA over all other forecast models uniformly for all our state variables. Uncertainty indices from Jurado, Ludvigson, and Ng (2015) for semi-annual returns are unavailable and thus not reported.

We can see that the random forest model generally has CSPA over all others in almost all horizons and state variables. This complements the results from before and provides more evidence that the random forest model is a good choice of model in our data. The partial least squares model exhibit CSPA except for  $h = 1$ . Elastic net model and LASSO has CSPA in  $h = 6$ , and the principal component regression model and ridge model are rejected the most often, suggesting that others often outperform them under certain states in the testing period.

Next, we repeat our forecast evaluation exercise for predicting firms belonging to the same industry

Table 5: All firms: CSPA results

		ADS	ANFCI	NFCI	rec prob	vxo	gecon	mpu	ept	JLN f	JLN r	JLN m
h=1	0forecast											
	Enet	*	*	*		*	*	*	*	*	*	*
	LASSO	*	*	*		*	*	*	*	*	*	*
	PCR	*	*	*		*	*	*	*	*	*	*
	PLS	*	*	*		*	*	*	*	*	*	*
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF		*			*						
h=3	0forecast	*	*	*	*	*	*	*	*	*	*	*
	Enet	*	*	*		*	*	*	*		*	*
	LASSO										*	*
	PCR	*	*	*	*	*	*	*	*	*	*	*
	PLS										*	
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF											
h=6	0forecast	*	*	*	*	*	*	*	*	NA	NA	NA
	Enet									NA	NA	NA
	LASSO									NA	NA	NA
	PCR	*	*	*	*	*	*	*	*	NA	NA	NA
	PLS									NA	NA	NA
	Ridge	*	*	*	*	*	*	*	*	NA	NA	NA
	RF									NA	NA	NA
h=12	0forecast			*		*		*	*		*	
	Enet	*	*	*	*	*	*			*	*	*
	LASSO	*	*	*	*		*			*	*	*
	PCR	*	*	*	*	*	*	*	*	*	*	*
	PLS									*		
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF											

Note: A model that reject CSPA null with 5% significance marked (\*).

sectors in the US. We define different industry sectors by a firm's first digit of the SIC code, and we have ten sectors. We do not report pairwise SPA test results due to difficulty in the readability of results.

Table 6 reports out-of-sample fit and MCS for each industry sector and horizon. The industries are labeled on the left column as sic#. For example, sic0 means the models' forecasts for firms whose first digit of the SIC code is 0, the *Agriculture, Forestry, and Fishing* sector. For simplicity, we will refer to each industry by the mnemonic used in Table 6.

First, we observe that for industries sic3 to sic 6, the random forest model has an excellent out-of-sample fit and is the sole member of MCS for all horizons. However, for industries sic7 and sic9, it has a worse fit than others and has never been included in MCS. Concurrently, the group whose first digit SIC code is 3, 4, 5, or 6 makes up most of the firms in Figure 4. This change in the model with the best fit could not have been observed had we not divided our firms into different groups.

Table 6: By industry:  $R_{oos}^2$ , MCS results

		h=1		h=3		h=6		h=12	
		$R_{oos}^2$	MCS	$R_{oos}^2$	MCS	$R_{oos}^2$	MCS	$R_{oos}^2$	MCS
sic0	Oforecast	-	✓	-		-		-	
	Enet	-0.0095		0.0024		0.0161		0.0267	
	LASSO	-0.0090	✓	0.0023		0.0149		0.0257	
	PCR	-0.0201		0.0072		0.0119		0.0040	
	PLS	-0.0152		0.0152	✓	0.0230	✓	0.0611	✓
	Ridge	-0.0202		-0.0092		-0.0086		-0.0063	
	RF	-0.0080	✓	0.0126	✓	0		0.0288	
sic1	Oforecast	-	✓	-		-		-	
	Enet	-0.0211		0.0191	✓	0.0142	✓	0.0069	
	LASSO	-0.0212		0.0191	✓	0.0139		0.0084	
	PCR	-0.0242		0.0030		0.0081		0	
	PLS	-0.0209		0.0056		0.0081		0.0128	✓
	Ridge	-0.0326		-0.0369		-0.0417		-0.0691	
	RF	0.0153	✓	0.0424	✓	0.0094		-0.0268	
sic2	Oforecast	-		-		-		-	
	Enet	0.0090	✓	0.0086		0.0069		0.0074	
	LASSO	0.0089		0.0086		0.0065		0.0099	
	PCR	0.0051		0.0023		0.0053		0.0040	
	PLS	0.0088		0.0098		0.0025		0.0016	
	Ridge	-0.0140		-0.0372		-0.0698		-0.0877	
	RF	0.0031		0.0753	✓	0.0580	✓	0.0654	✓
sic3	Oforecast	-		-		-		-	
	Enet	0		0.0175		0.0201		0.0269	
	LASSO	0		0.0175		0.0199		0.0281	
	PCR	-0.0084		0.0061		0.0112		0.0138	
	PLS	0		0.0166		0.0207		0.0295	
	Ridge	-0.0374		-0.0658		-0.1077		-0.0999	
	RF	0.0757	✓	0.0597	✓	0.0620	✓	0.0836	✓
sic4	Oforecast	-		-		-		-	
	Enet	0.0095		0.0115		0		-0.0012	
	LASSO	0.0094		0.0115		0		0.0035	
	PCR	0.0070		0.0048		0.0109		0.0093	
	PLS	0.0086		0.0077		-0.0086		-0.0139	
	Ridge	-0.0275		-0.0554		-0.0952		-0.1158	
	RF	0.3095	✓	0.3034	✓	0.2855	✓	0.2514	✓
sic5	Oforecast	-		-		-		-	
	Enet	0.0154		0.0112		-0.0033		-0.0023	
	LASSO	0.0153		0.0111		-0.0038		0.0032	
	PCR	0.0130		0.0066		0.0139		0.0141	
	PLS	0.0133		0.0092		-0.0136		-0.0204	
	Ridge	-0.0305		-0.0604		-0.1033		-0.1205	
	RF	0.1526	✓	0.1504	✓	0.1582	✓	0.1892	✓
sic6	Oforecast	-		-		-		-	
	Enet	0.0115		0.0138		0.0138		0.0202	
	LASSO	0.0114		0.0137		0.0132		0.0197	
	PCR	0.0041		0.0053		0.0092		0.0084	
	PLS	0.0106		0.0148		0.0162		0.0210	
	Ridge	-0.0225		-0.0565		-0.1090		-0.1401	
	RF	0.6098	✓	0.6210	✓	0.5961	✓	0.5918	✓
sic7	Oforecast	-		-		-		-	
	Enet	0.0151		0.0101		0.0054		0.0075	
	LASSO	0.0150		0.0101		0.0052		0.0108	✓
	PCR	0.0132		0.0050		0.0095	✓	0.0101	✓
	PLS	0.0160	✓	0.0124	✓	0.0036		0.0028	
	Ridge	-0.0133		-0.0520		-0.0968		-0.1175	
	RF	-0.2658		-0.2571		-0.2503		-0.2576	
sic8	Oforecast	-		-		-		-	
	Enet	0.0137		0.0225		0.0344		0.0498	
	LASSO	0.0137		0.0226		0.0347		0.0501	✓
	PCR	0.0183		0.0140		0.0251		0.0333	
	PLS	0.0218		0.0245		0.0416	✓	0.0520	✓
	Ridge	-0.0142		-0.0508		-0.0515		-0.0412	
	RF	0.1521	✓	0.0654	✓	0.0544	✓	0.0544	✓
sic9	Oforecast	-		-		-		-	
	Enet	0.0472		0.0100		-0.0043		0	
	LASSO	0.0472		0.0100	✓	-0.0046		-0.0012	
	PCR	0.0608		0.0081		0.0114	✓	0.0096	✓
	PLS	0.0650	✓	0.0105	✓	0.0097		0.0037	
	Ridge	0.0228		-0.0856		-0.1211		-0.1693	
	RF	-1.0684		-0.1537		-0.1451		-0.0889	

Note: Models included in MCS with 5% significance marked with ✓.

Next, we report test results for a model's CSPA by industry. For each model we count the number of times it has CSPA over all other competing models given a conditioning state variable.

$$\mathbb{E}(\Delta \mathbb{L}_{j,j',t+h-1} | X_t = x_t) \leq 0,$$

$$\forall j' \in \mathbb{J}, X_t \in \mathbb{X}$$

For example, for Agriculture, Forestry, and Fishing sector with SIC code 0 given  $h = 3$  for quarterly returns, we see that partial least squares model(PLS) has CSPA over all other models for all 11 state variables used. In 7, we report the CSPA test result for each state variable.

Table 7: By industry: CSPA results

		sic0	sic1	sic2	sic3	sic4	sic5	sic6	sic7	sic8	sic9
h=1	Oforecast	11	10	3	9	0	2	0	4	2	0
	Enet	1	3	7	9	0	3	0	10	5	11
	LASSO	11	3	7	9	0	3	0	7	5	9
	PCR	1	2	2	0	0	2	0	4	3	9
	PLS	3	3	0	9	0	2	0	11	5	11
	Ridge	1	0	0	0	0	0	0	0	0	0
	RF	8	11	2	8	11	11	11	0	7	1
h=3	Oforecast	9	10	6	3	0	4	0	7	0	0
	Enet	4	10	6	9	0	4	0	9	0	7
	LASSO	4	10	6	8	0	4	0	8	6	9
	PCR	2	1	2	0	0	4	0	2	0	1
	PLS	11	7	6	8	0	4	0	11	8	10
	Ridge	0	0	0	0	0	0	0	0	0	0
	RF	10	6	6	6	11	11	11	0	4	2
h=6	Oforecast	7	8	6	5	0	5	0	6	0	5
	Enet	6	8	5	7	0	3	0	7	0	0
	LASSO	5	7	5	7	0	3	0	7	0	0
	PCR	4	8	5	3	0	4	0	6	0	6
	PLS	5	8	6	7	0	3	0	8	7	7
	Ridge	0	0	0	0	0	0	0	0	0	0
	RF	8	8	8	8	8	8	8	0	8	2
h=12	Oforecast	6	11	4	4	1	4	0	10	0	8
	Enet	0	6	5	8	0	4	0	9	7	1
	LASSO	0	10	5	8	0	4	0	9	7	0
	PCR	0	6	4	2	0	5	0	8	0	11
	PLS	11	11	5	8	0	3	0	11	10	11
	Ridge	0	0	0	0	0	0	0	0	0	0
	RF	11	10	11	10	11	11	11	0	11	9

Note: We list the number of times a model has CSPA over all other models given a state variable with 5% significance. Maximum number of wins is 11 for h=1,3, and 12. Maximum wins for h=6 is 8.

Table 7 shows the CSPA results of models for multi-horizon returns. We see that for *Transportation, Communications, Electric, Gas and Sanitary service, Wholesale Trade & Retail Trade*, and *Finance, Insurance and Real Estate* sectors with SIC code 4, 5, and 6, respectively, the random forest model has CSPA over other models for 11 state variables. However, we also see industry sectors that the random forest model predicts poorly. In *Services* sector with SIC code 7, the model has rejects the null of CSPA for all 11 state variables. Therefore, we see no single model or a fixed group of models that have CSPA across all industries and horizons and this provides evidence of heterogeneity in different industry sectors

during our testing period.

Except for *Agriculture, Forestry and Fishing* industry for predicting monthly returns, the ridge model rejects the CSPA null for all industries and horizons, similar to previous forecast evaluation results. Therefore, when using an extensive collection of predictors, we argue that reducing the number of parameters to increase estimation quality is more critical than shrinkage. We suspect a repeated forecast exercise using a smaller set of predictors may show changes in the performance of the ridge model. We leave this for future research.

## 6 Conclusion

We investigate the predictive ability of supervised learning models predicting individual firm excess returns. Our main contribution is the forecast evaluation of models for their relative predictive ability. We evaluate models forecasting all individual firm returns and each group of firms belonging to different industry sectors in the US. First, using tests of SPA, CSPA, and constructing MCS, we find that partial least squares and random forest models perform better than others when forecasting for all firms in our data. Second, when models forecast returns for each industry sector, no single model has SPA, CSPA over other models and is always included in MCS. We argue that this is due to the heterogeneity of firms across different industry sectors.

# References

- ATHEY, S., AND G. W. IMBENS (2019): “Machine learning methods that economists should know about,” *Annual Review of Economics*, 11, 685–725.
- BAKER, S. R., N. BLOOM, AND S. J. DAVIS (2016): “Measuring economic policy uncertainty,” *The quarterly journal of economics*, 131(4), 1593–1636.
- BAUMEISTER, C., D. KOROBILIS, AND T. K. LEE (2022): “Energy markets and global economic conditions,” *Review of Economics and Statistics*, 104(4), 828–844.
- BREIMAN, L. (2001): “Random forests,” *Machine learning*, 45(1), 5–32.
- DE JONG, S. (1993): “SIMPLS: an alternative approach to partial least squares regression,” *Chemometrics and intelligent laboratory systems*, 18(3), 251–263.
- EFRON, B., AND T. HASTIE (2021): *Computer Age Statistical Inference, Student Edition: Algorithms, Evidence, and Data Science*, vol. 6. Cambridge University Press.
- GREEN, J., J. R. HAND, AND X. F. ZHANG (2017): “The characteristics that provide independent information about average US monthly stock returns,” *The Review of Financial Studies*, 30(12), 4389–4436.
- GU, S., B. KELLY, AND D. XIU (2020): “Empirical asset pricing via machine learning,” *The Review of Financial Studies*, 33(5), 2223–2273.
- HANSEN, P. R. (2005): “A test for superior predictive ability,” *Journal of Business & Economic Statistics*, 23(4), 365–380.
- HANSEN, P. R., A. LUNDE, AND J. M. NASON (2011): “The model confidence set,” *Econometrica*, 79(2), 453–497.
- HASTIE, T., R. TIBSHIRANI, J. H. FRIEDMAN, AND J. H. FRIEDMAN (2009): *The elements of statistical learning: data mining, inference, and prediction*, vol. 2. Springer.
- HUSTED, L., J. ROGERS, AND B. SUN (2020): “Monetary policy uncertainty,” *Journal of Monetary Economics*, 115, 20–36.
- JAMES, G., D. WITTEN, T. HASTIE, AND R. TIBSHIRANI (2013): *An introduction to statistical learning*, vol. 112. Springer.
- JURADO, K., S. C. LUDVIGSON, AND S. NG (2015): “Measuring uncertainty,” *American Economic Review*, 105(3), 1177–1216.
- LI, J., Z. LIAO, AND R. QUAEDVLIEG (2022): “Conditional superior predictive ability,” *The Review of Economic Studies*, 89(2), 843–875.
- MULLAINATHAN, S., AND J. SPIESS (2017): “Machine learning: an applied econometric approach,” *Journal of Economic Perspectives*, 31(2), 87–106.
- POLITIS, D. N., AND J. P. ROMANO (1994): “The stationary bootstrap,” *Journal of the American Statistical association*, 89(428), 1303–1313.
- TIBSHIRANI, R. (2011): “Regression shrinkage and selection via the lasso: a retrospective,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(3), 273–282.
- WELCH, I., AND A. GOYAL (2007): “A comprehensive look at the empirical performance of equity premium prediction,” *The Review of Financial Studies*, 21(4), 1455–1508.
- ZOU, H., AND T. HASTIE (2005): “Regularization and variable selection via the elastic net,” *Journal of the royal statistical society: series B (statistical methodology)*, 67(2), 301–320.

## 7 Appendix

### 7.1 Additional test results

We forecast multi-horizon returns test for CSPA using different state variables. The tables below report results where  $h=1$  is monthly returns,  $h=3$  is quarterly returns,  $h=6$  is semi-annual returns, and  $h=12$  is annual returns.



Table 8: By industry: h=1, CSPA results

		ADS	ANFCI	NFCI	rec prob	vxo	gecon	mpu	epu	JLN f	JLN r	JLN m
sic0	0forecast											
	Enet	*	*	*	*	*	*		*	*	*	*
	LASSO											
	PCR	*	*	*	*	*	*		*	*	*	*
	PLS	*	*	*		*	*			*	*	*
	Ridge	*	*	*	*	*	*		*	*	*	*
sic1	RF		*			*		*				
	0forecast									*		
	Enet	*	*	*		*		*	*	*		*
	LASSO	*	*	*		*		*	*	*		*
	PCR	*	*	*		*		*	*	*	*	*
	PLS		*	*	*	*		*	*	*	*	*
sic2	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF											
	0forecast	*		*	*	*	*			*	*	*
	Enet			*		*				*		*
	LASSO			*		*				*		*
	PCR	*	*	*	*	*	*			*	*	*
sic3	PLS			*		*				*		*
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF	*	*	*	*	*	*			*	*	*
	0forecast					*				*		
	Enet					*				*		
	LASSO					*				*		
sic4	PCR	*	*	*	*	*	*	*	*	*	*	*
	PLS			*	*	*				*		*
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF					*				*		
	0forecast	*	*	*	*	*	*	*	*	*	*	*
	Enet	*	*	*	*	*	*	*	*	*	*	*
sic5	LASSO	*	*	*	*	*	*	*	*	*	*	*
	PCR	*	*	*	*	*	*	*	*	*	*	*
	PLS	*	*	*	*	*	*	*	*	*	*	*
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF					*				*		
	0forecast	*	*	*	*	*	*	*	*	*	*	*
sic6	Enet	*	*	*	*	*	*	*	*	*	*	*
	LASSO	*	*	*	*	*	*	*	*	*	*	*
	PCR	*	*	*	*	*	*	*	*	*	*	*
	PLS	*	*	*	*	*	*	*	*	*	*	*
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF					*				*		
sic7	0forecast	*		*		*			*	*	*	*
	Enet					*				*		
	LASSO	*			*		*			*		
	PCR	*		*		*	*			*	*	*
	PLS			*		*				*		
	Ridge	*	*	*	*	*	*	*	*	*	*	*
sic8	RF	*	*	*	*	*	*	*	*	*	*	*
	0forecast	*	*	*	*	*		*		*	*	*
	Enet	*	*	*		*				*		*
	LASSO	*	*	*		*				*		*
	PCR	*	*	*		*		*	*	*		*
	PLS	*	*	*		*				*		*
sic9	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF	*	*	*	*	*	*	*	*	*	*	*
	0forecast	*	*	*	*	*	*	*	*	*	*	*
	Enet			*						*		
	LASSO										*	*
	PCR											

Note: A model that reject CSPA null with 5% significance marked (\*).

Table 9: By industry: h=3, CSPA results

		ADS	ANFCI	NFCI	rec prob	vxo	gecon	mpu	epu	JLN f	JLN r	JLN m
sic0	0forecast				*							*
	Enet	*	*	*		*				*	*	*
	LASSO	*	*	*		*				*	*	*
	PCR	*	*	*		*		*	*	*	*	*
	PLS											
	Ridge	*	*	*	*	*	*	*	*	*	*	*
sic1	RF		*									
	0forecast									*		
	Enet									*		
	LASSO									*		
	PCR	*	*	*		*	*	*	*	*	*	*
	PLS	*			*		*			*		
sic2	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF	*	*	*	*	*	*			*		*
	0forecast			*		*	*			*		*
	Enet			*		*	*			*		*
	LASSO			*		*	*			*		*
	PCR	*	*	*	*	*	*	*		*		*
sic3	PLS			*		*	*			*		*
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF	*	*	*	*	*	*			*		*
	0forecast		*	*		*		*	*	*	*	*
	Enet					*				*		*
	LASSO					*				*		*
sic4	PCR	*	*	*	*	*	*	*	*	*	*	*
	PLS	*	*	*	*	*	*	*	*	*	*	*
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF	*	*	*	*	*	*	*	*	*	*	*
	0forecast	*	*	*	*	*	*	*	*	*	*	*
	Enet	*	*	*	*	*	*	*	*	*	*	*
sic5	LASSO	*	*	*		*	*			*		*
	PCR	*	*	*		*	*			*		*
	PLS	*	*	*		*	*			*		*
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF	*	*	*	*	*	*	*	*	*	*	*
	0forecast	*	*	*	*	*	*	*	*	*	*	*
sic6	Enet	*	*	*	*	*	*	*	*	*	*	*
	LASSO	*	*	*	*	*	*	*	*	*	*	*
	PCR	*	*	*	*	*	*	*	*	*	*	*
	PLS	*	*	*	*	*	*	*	*	*	*	*
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF	*	*	*	*	*	*	*	*	*	*	*
sic7	0forecast	*				*			*	*		
	Enet					*				*		
	LASSO					*			*	*		
	PCR	*	*	*	*	*	*	*	*		*	
	PLS					*						
	Ridge	*	*	*	*	*	*	*	*	*	*	*
sic8	RF	*	*	*	*	*	*	*	*	*	*	*
	0forecast	*	*	*	*	*	*	*	*	*	*	*
	Enet	*	*	*	*	*	*	*	*	*	*	*
	LASSO		*			*			*	*	*	*
	PCR	*	*	*	*	*	*	*	*	*	*	*
	PLS					*			*	*	*	*
sic9	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF	*	*	*	*	*	*	*	*	*	*	*
	0forecast	*	*	*	*	*	*	*	*	*	*	*
	Enet					*		*	*	*		*
	LASSO			*	*	*	*	*	*	*	*	*
	PCR	*		*	*	*	*	*	*	*	*	*
sic0	PLS					*			*	*	*	*
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF	*	*	*	*	*	*	*	*	*	*	*
	0forecast	*	*	*	*	*	*	*	*	*	*	*
	Enet					*		*	*	*		*
	LASSO					*			*	*		*

Note: A model that reject CSPA null with 5% significance marked (\*).

Table 10: By industry: h=6, CSPA results

		ADS	ANFCI	NFCI	rec prob	vxo	gecon	mpu	epu
sic0	0forecast	★							
	Enet		★	★					
	LASSO		★	★	★				
	PCR	★	★	★		★			
	PLS		★	★	★				
	Ridge	★	★	★	★	★	★	★	★
	RF								
sic1	0forecast								
	Enet								
	LASSO					★			
	PCR								
	PLS								
	Ridge	★	★	★	★	★	★	★	★
	RF								
sic2	0forecast			★		★			
	Enet			★		★	★		
	LASSO			★		★	★		
	PCR			★		★	★		
	PLS			★		★			
	Ridge	★	★	★	★	★	★	★	★
	RF								
sic3	0forecast		★			★			★
	Enet					★			
	LASSO					★			
	PCR		★	★	★	★		★	
	PLS					★			
	Ridge	★	★	★	★	★	★	★	★
	RF								
sic4	0forecast	★	★	★	★	★	★	★	★
	Enet	★	★	★	★	★	★	★	★
	LASSO	★	★	★	★	★	★	★	★
	PCR	★	★	★	★	★	★	★	★
	PLS	★	★	★	★	★	★	★	★
	Ridge	★	★	★	★	★	★	★	★
	RF								
sic5	0forecast			★		★	★		
	Enet	★	★	★		★	★		
	LASSO	★	★	★		★	★		
	PCR		★	★		★	★		
	PLS	★	★	★		★	★		
	Ridge	★	★	★	★	★	★	★	★
	RF								
sic6	0forecast	★	★	★	★	★	★	★	★
	Enet	★	★	★	★	★	★	★	★
	LASSO	★	★	★	★	★	★	★	★
	PCR	★	★	★	★	★	★	★	★
	PLS	★	★	★	★	★	★	★	★
	Ridge	★	★	★	★	★	★	★	★
	RF								
sic7	0forecast					★		★	
	Enet					★			
	LASSO					★			
	PCR					★		★	
	PLS								
	Ridge	★	★	★	★	★	★	★	★
	RF	★	★	★	★	★	★	★	★
sic8	0forecast	★	★	★	★	★	★	★	★
	Enet	★	★	★	★	★	★	★	★
	LASSO	★	★	★	★	★	★	★	★
	PCR	★	★	★	★	★	★	★	★
	PLS					★			
	Ridge	★	★	★	★	★	★	★	★
	RF								
sic9	0forecast			★	★	★			
	Enet	★	★	★	★	★	★	★	★
	LASSO	★	★	★	★	★	★	★	★
	PCR			★		★			
	PLS					★			
	Ridge	★	★	★	★	★	★	★	★
	RF	★	★	★	★	★	★		

Note: A model that reject CSPA null with 5% significance marked (★).

Table 11: By industry: h=12, CSPA results

		ADS	ANFCI	NFCI	rec prob	vxo	gecon	mpu	epu	JLN f	JLN r	JLN m
sic0	0forecast	*			*		*		*			*
	Enet	*	*	*	*	*	*	*	*	*	*	*
	LASSO	*	*	*	*	*	*	*	*	*	*	*
	PCR	*	*	*	*	*	*	*	*	*	*	*
	PLS											
	Ridge	*	*	*	*	*	*	*	*	*	*	*
sic1	RF											
	0forecast											
	Enet		*	*	*						*	*
	LASSO				*							
	PCR	*	*	*						*		*
	PLS											
sic2	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF						*					
	0forecast	*	*	*		*				*	*	*
	Enet		*	*		*				*	*	*
	LASSO		*	*		*				*	*	*
	PCR		*	*		*	*			*	*	*
sic3	PLS		*	*		*				*	*	*
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF									*		
	0forecast		*	*		*	*	*		*		*
	Enet			*		*				*		
	LASSO			*		*				*		
sic4	PCR		*	*	*	*	*	*		*	*	*
	PLS			*		*				*		
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF									*		
	0forecast	*	*	*	*	*	*	*	*	*	*	*
	Enet	*	*	*	*	*	*	*	*	*	*	*
sic5	LASSO	*	*	*	*	*	*	*	*	*	*	*
	PCR	*	*	*	*	*	*	*	*	*	*	*
	PLS	*	*	*	*	*	*	*	*	*	*	*
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF									*		
	0forecast	*		*		*		*		*	*	*
sic6	Enet	*	*	*	*	*	*	*	*	*	*	*
	LASSO	*	*	*	*	*	*	*	*	*	*	*
	PCR	*	*	*	*	*	*	*	*	*	*	*
	PLS	*	*	*	*	*	*	*	*	*	*	*
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF									*		
sic7	0forecast	*		*		*		*		*	*	*
	Enet	*	*	*	*	*	*	*	*	*	*	*
	LASSO	*	*	*	*	*	*	*	*	*	*	*
	PCR	*	*	*	*	*	*	*	*	*	*	*
	PLS	*	*	*	*	*	*	*	*	*	*	*
	Ridge	*	*	*	*	*	*	*	*	*	*	*
sic8	RF	*	*	*	*	*	*	*	*	*	*	*
	0forecast	*	*	*	*	*	*	*	*	*	*	*
	Enet	*	*	*	*	*	*	*	*	*	*	*
	LASSO	*	*	*	*	*	*	*	*	*	*	*
	PCR	*	*	*	*	*	*	*	*	*	*	*
	PLS	*	*	*	*	*	*	*	*	*	*	*
sic9	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF					*				*		
	0forecast	*		*	*	*	*	*	*	*	*	*
	Enet	*	*	*	*	*	*	*	*	*	*	*
	LASSO	*	*	*	*	*	*	*	*	*	*	*
	PCR	*	*	*	*	*	*	*	*	*	*	*
sic0	PLS	*	*	*	*	*	*	*	*	*	*	*
	Ridge	*	*	*	*	*	*	*	*	*	*	*
	RF					*				*		
	0forecast	*		*	*	*	*	*	*	*	*	*
	Enet	*	*	*	*	*	*	*	*	*	*	*
	LASSO	*	*	*	*	*	*	*	*	*	*	*

Note: A model that reject CSPA null with 5% significance marked (\*).