Empirical investigation on supervised machine learning models predicting equity risk premium

Myong Jong Shin*

September 18, 2023

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 variables as return predictors. We forecast excess returns for (1) all individual firms and (2) specific groups of firms from different industry sectors 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

*Department of Economics, Indiana University Bloomington, 100 S Woodlawn Ave, Bloomington, IN 47405. Email: myonshin@iu.edu

1 Introduction

In empirical finance literature, predicting the equity risk premium—the difference between stock returns and the risk-free rate—is a pivotal and frequently explored research topic. In this paper, we conduct a comparative analysis of popular supervised machine learning models each forecasting multi-horizon equity risk premiums for companies listed on major stock markets in the US. The goal of this research is test with statistical significance the identified model or a set of models with better return predictability over other models. We do so by employing a set of novel out-of-sample forecast evaluation tests from highly influential papers in forecast evaluation literature. We study for (1) individual firm-level risk premiums and (2) risk premium of individual groups of firms categorized according to the US standard industrial classification.

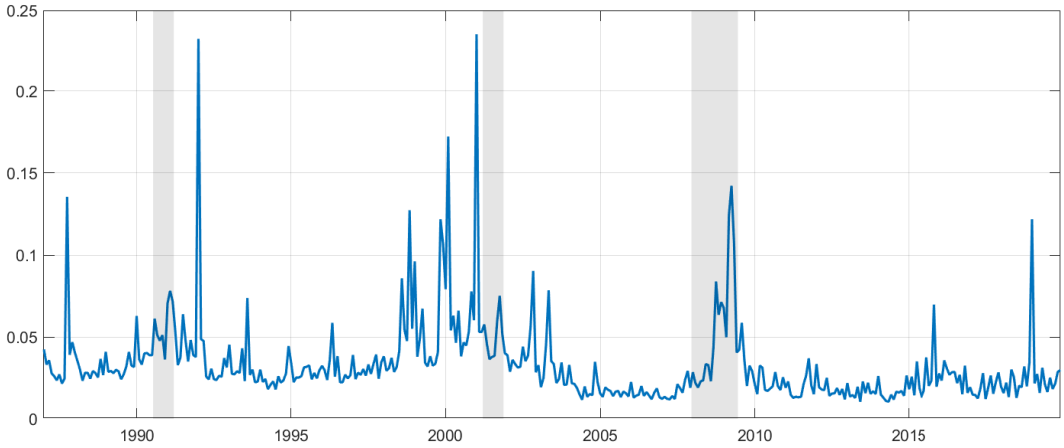
Predicting excess risk premium is a notoriously challenging task. There has been many forecasting methods having good return predictability measured by some loss function such as squared errors. However, they are often sensitive to the periods of samples studied, and hyperparameters that are chosen based on the observed predictive performance during model estimation. Therefore a forecasting method's predictability is often time-varying. We will refer equity risk premium also as stock returns, excess returns, or simply, returns, for convenience and use them interchangeably throughout the paper.

When attempting to identify a model or a set of models that has return predictability over a chosen benchmark model, we argue that a rigorous examination using statistical tests is critical to choosing the right model for return forecasting. Rather than computing simple summary statistics such as a ratio of mean squared errors two models over the out-of-sample periods, we employ forecast evaluation tests that relies on asymptotic theories that provide valid critical values for inference.

For example, the out-of-sample fit statistic, R^2_{OOS}, can be used during machine learning prediction exercise to evaluate the forecasts. 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 paper, researchers can make better comparisons between models with statistical significance.

The main objective of this paper is to use a set of statistical tests novel to return forecasting literature to evaluate the forecasting models and to provide the forecasters a reliable collection of practical models backed by evidence gathered from this paper.

Figure 1: Forecast errors from *0forecast* model.



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

For example, figure (1) shows the cross-sectional average of squared forecast errors from predicting monthly excess returns with a naive forecast of zeros from January 1987 to December 2019. We refer this naive approach *0forecast*, and it can be interpreted as the time series of cross-sectional averages of individual firms' uncentered variance from January 1987 to December 2019. To forecast any firm i's excess returns, the prediction of *0forecast* is $\hat{r}_{it} = 0, \forall i, t$. The cross-sectional average of forecast error is then taken at each time t to create a time series of forecast errors.

It is clear from figure (1) that *0forecast* does not forecast well over certain historical events in the US financial market. Examples of such time periods include the stock market crash of 1987, the dot com bubble crash of the late 1990s, the great recession of 2008, and many more.

Although *0forecast* may seem an absurd choice of modeling and should never be used, we will show in 6 that when we evaluate models forecasting monthly returns, *0forecast* is among the set of best predictive models tested. For this reason, while *0forecast* is not the model we recommend for forecasting, we

will use it as a benchmark to measure some of the models used in our paper for their marginal predictive improvement over *0forecast*.

There are a plethora of models that have been used to predict excess returns of firms or the market excess returns. For a tractable but relevant study, we limit ourselves to exploring only the more recently developed models that have not been thoroughly evaluated for their relative predictive performance in the empirical finance literature. In particular, we set the scope of the models searched in this paper to be of *supervised machine learning models* that have been used to predict excess returns in papers published in renowned academic journals. These include a wide collection of models that are already used widely in economics and statistics literature, such as least squares or least squares with a regularization term. A more novel set of models among them includes variations of tree method that are widely used in academics and industry for classification problems as well as forecasting using regression. We will refer *supervised machine learning models* as ML models and use them interchangeably, and unsupervised machine learning models are beyond the scope of this paper.

For our analysis, we benchmark a recently published paper Gu, Kelly, and Xiu (2020), henceforth GKX, for the choice of data and ML models used. GKX surveys a set of supervised machine learning models to forecast monthly equity risk premiums for individual firms in the US stock market. Their models include simple linear models such as principal components, partial least squares, as well as nonlinear models such as nonparametric basis expansions with regularization, gradient boosted trees, and random forest. Their data is extensive with a collection of monthly stocks from firms listed on NYSE, AMEX, and NASDAQ from March 1957 to December 2016. For forecast evaluation, GKX reports each model's out-of-sample goodness-of-fit, and determine a model's predictive ability over another model by calculating the ratio of their mean squared errors(MSE). If the ratio is greater than 1, the MSE of the model at the denominator is deemed to have better predictive ability.

Here we state our main contributions of our paper. First, we extend the analysis of GKX to forecasting not only monthly excess returns of the market and individual firms, but also the returns that materializes later than a month. We refer to these returns as *multi-horizon returns*. In our study, we forecast monthly, quarterly, semi-annual, and annual excess returns. Second, we conduct forecast evaluation tests to determine whether if a model, or a set of models has better predictive ability compared to

the rest of the models in this horse race with statistical significance. To do so, we will test for what is known in the literature as *superior predictive ability*, and *conditional superior predictive ability*. Finally, we extend the analysis of return predictability not only for individual firms or the market but subsets of firms where they are grouped according to their respective industry sectors in the US economy. This was motivated to observe weather if when forecasting excess returns, the choice of models should vary with respect to their industry sectors.

In summary, our findings underscore that the preferred models for return prediction can vary based on the forecasting target and the specific tests applied. We assessed the models' ability to predict average multi-horizon excess returns of individual firms in the stock market. Both the partial least squares model and the random forest model generally outperformed others when tested for their *superior predictive ability* and *conditional superior predictive ability*. However, when focusing on excess returns for firms within specific industry sectors, the predictive ability of models varied by sector, without any single model or a set of models dominating the rest.

We do not claim that the set of supervised machine learning models used in our research represents a complete list of ML models that econometricians should consider when modeling models for return forecasting. The main reason behind our choice of ML models for the horse race is that we selected the models used for the forecasting exercise in GKX so that both GKX and our paper can complement each other.

Some models that are used in GKX are omitted in our paper due to a lack of high-performance computation resources and memory storage currently available to us at the time of writing this paper. 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 and select groups of splines for forecasting using the L_1 absolute penalty function. On the other hand, the 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) provides a literature review of classic asset pricing models and the recent advent of machine learning in understanding the financial markets. Sec-

tion (3) describes the supervised machine learning models used in our article. Section (4) describes the out-of-sample forecast evaluation tests for superior predictive ability and conditional superior predictive ability. Section (5) describes the data used. Section (6) shows the test results and findings, and section (7) concludes our paper.

2 Literature Review

Here we provide a literature review of related research to our paper. First, our paper is in line with the long list of academic papers regarding understanding the dynamics of market equity risk premium and forecasting it as accurately as possible with asset pricing models. At its core, gauging an asset's risk premium in the future is a forecasting problem. The risk premium of an asset is the condition expectation of the future return minus the risk-free rate. The functional form of the condition expectation must be approximated and many researchers have assume a linear form.

The foundation of our equity risk premium prediction problem is grounded on the classical empirical asset pricing research. There is a plethora of seminal papers in the literature that we owe our theoretical background to. Just to name a few, we find the relationship between our ML models to the classical *Capital Asset Pricing Model (CAPM)* of Sharpe (1964). Provided we have access to a zero-risk asset with return R_f, the *Security Market Line* relationship between an asset and the market return of CAPM can be written as a linear relationship between the expected return of an asset and the return of the wealth portfolio β_{asset,R^W}.

$$\mathbb{E}(R^{\text{asset}}) = R_f + \beta_{R^W}(\mathbb{E}(R^W) - R_f) \tag{1}$$

By using the US treasury bill rate and stock market portfolio as proxies for R_f and R^W, we can estimate the market beta β_{R^W} equation (1) via regression.

The *3-factor model* of Fama and French (1993) further extends equation (1). While CAPM prices the asset using a single market factor $\mathbb{E}(R^W)$, *3-factor model* adds two more factors to CAPM in order to capture other sources of potential stock return variability observed in the data. The additional factors

are also linear to the return of the asset.

$$\mathbb{E}(R^{\text{asset}}) = R_f + \beta_{R^W}(\mathbb{E}(R^W) - R_f) + \beta_{SMB}\mathbb{E}(SMB) + \beta_{HML}\mathbb{E}(HML) \tag{2}$$

SMB(Small Minus Big) stands for the difference in returns between small-cap and large-cap stocks. As small stocks tend to have higher average returns than large stocks, by including *SMB*, the model attempts to account for this size effect. *HML*(High Minus Low) represents the difference in returns between stocks with high book-to-market ratio and those with low book-to-market ratio, which is known in the data to be positive on average.

Following the modeling style of Sharpe (1964) and Fama and French (1993), many other influential papers extends the pricing equation by adding more factors, such as the *5-factor model* of Fama and French (2015). These models, while providing a robust theoretical foundation, have limitations in their ability to account for the complexity and non-linearity observed in real-world financial markets. The linear relationship, which was imposed mainly for simplicity, can be relaxed for including non-linear relationship between the factors and the asset's excess return. We seek to use ML models to model the non-linear relationship to produce better predictions.

The application of machine learning in the field of empirical asset pricing has garnered substantial attention in recent years. Machine learning algorithms like decision trees, support vector machines, and neural networks have emerged as powerful tools for modeling complex relationships and have shown promising results in predicting asset prices, offering an alternative to traditional pricing models.

Furthermore, ML models can also be used to incorporate a very large collection of factors in the pricing equation. Over time, the academic community has compiled an impressive array of predictors, of which many researchers believe can forecast returns. However, these predictors often do not perform as well as advertised when the time scope of the forecast change. Also most good predictors share similarities and exhibit high correlations that may cause rank deficiency in the covariate matrix. The conventional prediction techniques such as least squares also falter when the number of predictors is near or exceeds the number of observations due to rank deficiency. Machine learning is a suitable tool on for handling these problems by selecting predictors with regularization, and reducing their dimensions for making

better quality predictions.

Machine learning techniques are increasingly adopted in forecasting stock returns. These techniques present an alternative to traditional methods, offering flexible, data-driven modeling capable of capturing complex, non-linear relationships, even when utilizing a plethora of predictors.

For instance, Rapach, Strauss, and Zhou (2013) used the elastic net method of Zou and Hastie (2005) to investigate the monthly stock returns of various countries to identify leading-lagging relationships between countries' stock returns. Rapach, Strauss, Tu, and Zhou (2019) use *Least Absolute Shrinkage and Selection Operator*, or LASSO, to analyze industry return predictability based on the information in lagged industry returns. Freyberger, Neuhierl, and Weber (2020) applied the *adaptive group LASSO*, a method that heavily penalizes groups of covariates with slopes closer to 0. They use this to select predictors and estimate how those selected affect the expected returns nonparametrically. Chinco, Clark-Joseph, and Ye (2019) used LASSO to predict one-minute returns of stocks listed on the *New York Stock Exchange* between January 2005 and December 2012, relying on the cross-section of lagged portfolio returns and specific firm characteristics as predictors.

Here, We begin by highlighting some foundational papers in the out-of-sample forecast evaluation tests that are closely related to our work. A comprehensive discussion on the forecast evaluation tests utilized in our paper is deferred and will be elaborated upon in section (4). Clark and McCracken (2013) provided a comprehensive survey of the extant literature and is an excellent source for those new to the research field of forecast evaluation.

The test of equal predictability between forecast models from Diebold and Mariano (1995) is arguably the most well-known test in the forecast evaluation literature. For a given parametric regression model j and j' pair at time t, they are considered to have *equal predictive ability* if the expectation of the difference in their loss equals zero.

$$\mathbb{E}(\Delta\mathbb{L}_{j,j',t}) = 0 \tag{3}$$

Let $\Delta\mathbb{L}_{j,j',t}$ be a time series of loss differential between the two models. For example, this might be a

difference between squared forecast errors from them. Assuming that $\Delta\mathbb{L}_{j,j',t}$ is weakly stationary and exhibit short memory, the models possess *equal predictive ability* under the null hypothesis. The assumptions yield the standard central limit theorem of $\Delta\mathbb{L}_{j,j',t}$, and the test uses the standard normal critical values for testing. West (1996) later advanced the theoretical basis of the test, identifying conditions where parameter estimation error does not affect normality result of Diebold and Mariano (1995) and how to estimate its asymptotic variance.

One instance where the asymptotic normality is not valid for testing *equal predictive ability* for two competing parametric forecasting models is when they are *nested*. This means a set of predictors from one model is a subset of the predictors from the other model. Using an example from Clark and McCracken (2013), consider a case where two nested linear models are being compared whose parameters are estimated via least squares. Define a k by 1 be a vector of predictors from model 2 as $x_{2,t} = (x'_{1,t}, x'_{22,t})'$, and let $x_{1,t}$ be a vector of predictors from model 1. k is equal to $k_1 + k_{22}$, where the length of each vectors $x_{1,t}$ and $x_{22,t}$ are k_1 and k_{22} respectively, with $x_{22,t}$ being the set of predictors for model 2 but not in model 1. To make predictions for y at time t for $t + \tau$, the two models can be written as

$$y_{t+\tau} = x'_{i,t}\beta_i + u_{i,t+\tau}, \text{ for } i = 1, 2$$

such that model 2 with $x_{1,t}$ *nests* model 1 with additional predictors that are not in model 1. Thus, under the null of equal predictability between models, $\beta_2 = (\beta'_1, \beta'_{22})' = (\beta'_1, 0)'$. Therefore, under the null, both models are identical and the forecast errors from both of the models are thus exactly the same.

We can categorize the research that followed to address this nested model problem into two approaches. One approach is by developing population-level predictive ability tests between nested models. Most notable works in this approach are Clark and McCracken (2001) and McCracken (2007). The main characteristic of their theory is that the parameters of the forecast models are allowed to converge in probability to their limit regardless of the choice of window-scheme for out-of-sample testing. Their out-of-sample tests use critical values from a nonstandard limit distribution dependent on nuisance parameters such as the ratio of size of testing sample to the entire sample. Hansen and Timmermann (2015) has shown later that their test statistics are asymptotically equivalent to a simple linear combination of wald statistics. The critical values can also be bootstrapped following the algorithm developed by Clark

and McCracken (2012). Other related papers include Clark and West (2006), Patton and Timmermann (2012), and Clark and West (2007).

On the other hand, Giacomini and White (2006) provides an alternative framework to test finite-sample predictive ability between two competing models. The parameters of the forecast models are only estimated in *finite* sample and thus it does not suffer from the nested model problem in population. This is implemented by requiring that the forecasts be constructed using rolling scheme with a finite size of sample used for training when making out-of-sample predictions. The test allows for both nested and non-nested model comparisons. The forecasts can also be based on estimators that are not linear parametric, such as bayesian, non-parametric, or semi-parametric.

ML models' forecasts in our paper can also be evaluated for their finite-sample predictive ability using the framework of Giacomini and White (2006), as the models are not restricted to be parametric linear. For this reason, all of the out-of-sample tests we conduct in section (4) will also be based on a rolling scheme and will be interpreted as testing the models for their predictive ability given a finite-sample that rolls over after each forecast is made.

While the test of finite-sample predictive ability of Giacomini and White (2006) only concerns over a pair of models, in reality a forecaster may wish make the same evaluation using more than two models. White (2000) first proposed the framework of comparing predictive ability for multiple models with the *reality check* test.

The null hypothesis of the test is that, out of the set of alternative models being considered, none can outperform a simple benchmark model. The benchmark model has to be chosen a priori, and the test is based on a bootstrap-based procedure. By incorporating the rolling window scheme, we can interpret the *reality check* test as testing for finite-sample predictive ability of multiple models. As we have multiple ML models for comparison, our choice of test statistics in this paper also is in line with this. However, the tests used in this paper build upon the work of White (2000) by improving the power of the test and the benchmark model not having to be chosen a priori. We discuss the details in section (4). Related testing procedures for multiple hypothesis testing not used in this paper includes, Romano and Wolf (2005), Clark and McCracken (2012), and Romano, Shaikh, and Wolf (2008).

3 Supervised Machine Learning Models

In this section we describe what ML models were used for our forecasting exercise. In our paper, we provide only the general description of each ML model used and document its tuning parameters if needed. For more in-depth discussion of the models studied here, 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.

In machine learning literature, prediction problems can be divided into two cases; 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.

At its core, unsupervised machine learning is a type of algorithm that a machine can recognize and learn patterns from given data without explicit supervision given by a target variable. In contrast, for supervised learning, we provide the algorithm with both input data and the correct output. In our paper, the input data will be our set of predictors and the output would be the realized historical excess return of firms.

The unsupervised learning only involves input data. The machine tries to learn the underlying structure or distribution in the data without any labeled responses to guide the learning process. Such techniques include clustering where we group data points into clusters based on their similarity such as their distance. Techniques such as *Principal Component Analysis* (PCA) can reduce the dimensionality, or the number of predictors in the data, by creating the linear combination from the set of predictors that can best explain their covariance matrix.

We note that although PCA is generally considered an unsupervised learning, we include them among our ML models for prediction. This is because we use PCAs as generated regressors to predict excess

returns using least squares and employ scree plots to determine how many components we want to put in. Also, as we will see in 5, there is another ML model that we use that has closely related to PLS. Therefore we include PCA as part of our forecasting horse race. To avoid confusion, we will refer this method as principal component regression, or PCR.

We can write all of our supervised machine learning models' prediction problem via the regression in equation (4). Let θ be the coefficients vector of size P and $\mathbb{J}$ be the set of supervised machine models used in this article.

$$\sum_{\tau=0}^{h-1} r_{i,t+\tau} = f_j(z_{i,t-1};\theta) + \sum_{\tau=0}^{h-1} \epsilon_{i,t+\tau}, \;\; j \in \mathbb{J} \tag{4}$$

Individual firm's h-horizon excess return is calculated as its continuously-compounded stock returns minus the risk-free rate, and we denote it as $\sum_{\tau=0}^{h-1} r_{i,t+\tau}$ where h is the number of months it takes the return to materialize. We denote the different supervised machine learning models by using the subscript $j \in \mathbb{J}$. In our paper, we forecast for one month ahead ($h = 1$), a 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 $t = 1, \cdots T$ due to newly enlisted and delisted firms in the stock market.

All ML models use a common set of predictors $z_{i,t-1}$ as inputs for the models to make predictions. This is done to make proper comparisons between models' forecasts that stems from the difference in the functional form of each ML model alone. The are lagged by one month period relative to the monthly excess returns. As a result of using compounding returns, all of our models have error terms as a sum of errors for firm i's return in monthly frequency and is written in equation (4). We forecast returns by the estimated conditional expectation of the regression. The functional form of $f_j(z_{i,t-1};\theta)$ in 4 is assumed by each ML model's specification. For example, PCR assumes $f_j(z_{i,t-1};\theta)$ to be estimated by a chosen number of principal components. A firm i's h-horizon excess returns are predicted as $f_j(z_{i,t-1},\hat{\theta})$ for each model j. For example, when using a partial least squares model for prediction, we denote its forecast as $f_{pls}(z_{i,t-1},\hat{\theta})$. Likewise, 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.

3.1 Principal Component Regression

Principal component regression uses principal components(PCs) that are generated from the set of predictors Z during training period. Each PC is denoted as ω_k with the subscript k. PCs are the solution to the optimization problem in equation (5). The corresponding solution to (5) is known to be the eigenvectors of the sample covariance matrix and their associated eigenvalues shows how much of the variation in the sample covariance matrix the kth eigenvector explained. Principal components are orthogonal by design thus the combined variations explained by both ω_1 and ω_2 is simply the sum of their eigenvalues.

$$\begin{aligned} \omega_k &= \arg\max_{\omega} Var(Z\omega), \\ \text{subject to} \quad \omega'\omega = 1, \;& \; Cov(Z\omega, Z\omega_k) = 0, \\ & k = 1, 2, \cdots, P-1. \end{aligned} \tag{5}$$

Using too many PCs erodes the benefit of dimension reduction and makes predictions worse. Therefore, to select the number of PCs to use for forecasting from training data, we can use a scree plot of eigenvalues to pick the number of PCs ordered from the largest eigenvalue to the smallest. 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. We find that both approaches suggest a similar number of PCs to use fore forecasting. For our data, we find that generally, PCs explaining approximately 60% of the covariance matrix of covariates or lower provide good predictions. Therefore we report the results from the PCR specification that explains 60% of the covariance matrix at section (6).

3.2 Partial Least Squares Regression

Partial least squares regression (PLS) extracts PLS components ω_k by solving equation (6) to run least squares regression with a chosen number of PLS components. This regression can be related to PCR in that whereas principal components are the linear combinations of predictors that best explain the covariance matrix of predictors, PLS components are the linear combinations of predictors that best

explain the squared covariance between the multi-horizon excess returns and the matrix of covariates Z.

$$\omega_k = \arg\max_{\omega} Cov^2(\sum_{\tau=0}^{h-1} r_{t+\tau}, Z\omega), \tag{6}$$
$$\text{subject to} \quad \omega'\omega = 1, \;\; Cov(Z\omega, Z\omega_k) = 0$$
$$k = 1, 2, \cdots, P-1.$$

To solve equation (6), we use the *SIMPLS* algorithm by De Jong (1993) to generate PLS components. Then, we use the validation data to choose the number of PLS components that results in the smallest mean squared error during the validation time period. The chosen number of components is then used for forecasting during testing.

For both PCR and PLS, the least squares loss function is used to estimate the coefficient vector θ, and we make prediction with it using $f(z_{i,t-1}, \hat{\theta})$. The equation (7) shows the resulting pooled least squares estimator.

$$\hat{\theta} = \arg\min_{\theta} \sum_{i=1}^{N_t} \sum_{t=1}^{T} \big(\sum_{\tau=0}^{h-1} r_{i,t+\tau} - f_j(z_{i,t-1}; \theta) \big)^2 \tag{7}$$

3.3 LASSO

LASSO stands for *least absolute shrinkage and selection operator* and its name was first coined by Tibshirani (1996). The *least absolute shrinkage* refers to the penalty function added to the least squares loss function for finding the unique optimizer θ. The penalty term constrains the magnitude of the estimate θ by its L_1 norm. By constraining the magnitude of the coefficient vector θ by $l1$ norm, LASSO induces *sparsity* where we can have some $\hat{\theta}_k = 0$. The optimal tuning parameter λ 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 λ, and the chosen value is used for forecasts during the testing period.

By adding the penalty we give a bias to the model's prediction and therefore lose the *best linear unbiased estimator*, or *BLUE*, property of least squares. However it allows us to reduce the variance of the data and also perform model selection my dropping off certain predictors entirely.

For our research, we use a simple absolute penalty on each slope coefficient scalar θ_k, and our LASSO problem can be stated as equation (8) in the lagrangian form with $\lambda \geq 0$.

$$\hat{\theta}_{lasso} = \arg\min_{\theta} \Big(\sum_{i=1}^{N_t} \sum_{t=1}^{T} \big(\sum_{\tau=0}^{h-1} r_{i,t+\tau} - f_j(z_{i,t-1}; \theta) \big)^2 + \lambda \sum_{k=1}^{p} |\theta_k| \Big) \tag{8}$$

The absolute shrinkage penalty is widely used in economics, and there are many variants of the LASSO method with different penalty functions. For example, the adaptive LASSO of Zou (2006) further penalizes the the individual scalar θ_k using weights to scale down some slope estimates with higher values such that the model penalizes the slopes of *theta* more uniformly. For in-depth discussion of the topic, Tibshirani (2011) provides an excellent review of the different generalizations of the LASSO method.

3.4 Ridge

The ridge model also shares the idea of penalizing the coefficient vector θ of the least squares loss function. However, in contrast to LASSO, ridge adopts an L_2 norm as its penalty to the least squared objective function that constrains the magnitude of θ_k. This will not let the slopes of predictors drop off entirely but instead shrink their magnitude towards zero and a tuning parameter λ can be chosen to determine the rate of shrinkage. Similar to LASSO, λ needs to be searched over a grid of candidate values. For each estimation of the model using training data, we again use the validation data to find an optimal value for λ, and the chosen value is used for forecasts during the testing period.

The optimization problem can be stated as equation (9) in lagrangian form with $\lambda \geq 0$.

$$\hat{\theta}_{ridge} = \arg\min_{\theta} \Big(\sum_{i=1}^{N_t} \sum_{t=1}^{T} \big(\sum_{\tau=0}^{h-1} r_{i,t+\tau} - f_j(z_{i,t-1}; \theta) \big)^2 + \frac{1}{2} \lambda \sum_{k=1}^{p} \theta_k^2 \Big) \tag{9}$$

The benefit of the ridge penalty can be easily seen with a simple example of linear regression. When the covariate matrix of the regression X is not full column rank, the least squares estimator will not give unique estimator as $X'X$ is not invertible. This can also be interpreted in that in the spectral decomposition of $X'X$, some of its eigenvalues are zero due to rank deficiency. By adding the ridge penalty which are positive values by design, the eigenvalues of the $X'X$ becomes all positive thus the matrix become invertible and the penalized least squares estimator gives a unique estimate.

3.5 Elastic Net

The elastic net model from Zou and Hastie (2005) adds the penalty function to the least squares loss function by the convex combination of L_1 and L_2 norm constraint. Thus the model estimates θ through both *variable selection* and *shrinkage* of LASSO and ridge respectively. For our study, the convex combination parameter ρ is fixed to be 0.5 by choice, as it was the choice from GKX also, and the optimal tuning parameter λ is selected using validation data.

$$\hat{\theta}_{enet} = \arg\min_{\theta} \Big(\sum_{i=1}^{N_t} \sum_{t=1}^{T} \big(\sum_{\tau=0}^{h-1} r_{i,t+\tau} - f_j(z_{i,t-1}; \theta) \big)^2 + \lambda(1-\rho) \sum_{k=1}^{p} |\theta_k| + \frac{1}{2}\lambda\rho \sum_{k=1}^{p} \theta_k^2 \Big) \tag{10}$$

3.6 Random Forest

The random forest model is a bootstrap aggregation or 'bagging' (Breiman (2001)) of individual regression trees. In other words, the model makes predictions using an ensemble of B number of bootstrapped trees where the re-sampling is done with replacements and in batches. This means that same observations can be picked again during bootstrap and the set of predictors to sample from are also selected in random batches.

The choice of using random batches is to first make sure that all relevant features fore forecasting are checked for branching the regression trees. If the same set of predictor data were used for each regression tree, they would most like be very similar to each other given that they will choose the same list of predictors for branching. Secondly the choice of using random batches serves as a regularization to the loss minimization, in that it smooths out the surfaces of the loss function that helps the gradient decent algorithm to descend the loss faster to the optimal level.

We use the algorithm by Hastie, Tibshirani, Friedman, and Friedman (2009) for growing the random forest. First, using re-sampled training sample $\{z^b_{i,t-1}, \sum_{\tau=0}^{h-1} r^b_{i,t+\tau}\}, b = 1, \cdots, B$, each binary regression tree b grows two branches per branching in a greedy algorithm fashion. This means that when branching, it is determined solely based on minimizing the squared loss at the local node without considering other branches for global optimization. A branch separates the data by using the predictor in the batch that can minimize the squared loss. Its goal is to categorize 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 in two to minimize squared loss in equation (11) where $|C|$ denotes the number of observations at branch C. The best prediction $\hat{\theta}$ will be the average excess returns associated with the predictor observation at branch C.

$$H(\theta, C) = \frac{1}{|C|} \sum_{z_{i,t-1}^b \in C} (\sum_{\tau=0}^{h-1} r_{i,t+\tau}^b - \theta)^2 \tag{11}$$

We stop splitting branches when the maximum depth of a tree L is reached. We use 300 regression trees, and each tree has a maximum depth of $L = 6$. The prediction of a tree using bth 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 excess returns belonging to the same branch[1].

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

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

Finally, the bags of predictions from individual trees are averaged to produce the prediction from the random forest in equation (12).

$$f_{rf} = \frac{1}{B} \sum_{b=1}^{B} f_{tree,b}(z_{i,t-1}^b, \hat{\theta}^b, L) \tag{12}$$

4 Testing the predictive performance of ML models

Here, we introduce the statistical tests used to evaluate the predictive performance of ML models from section (3). When comparing the accuracy of the forecasts from different models, researchers outside of econometrics field usually rely on simple summary statistics. For example, the out-of-sample fit statistic of Campbell and Thompson (2008), otherwise known as R^2_{OOS}, is often calculated to decide which model has the best predictive performance in out-of-sample. A higher R^2_{OOS} relative to other models are interpreted as the model having a better fit out-of-sample, or equivalently having a smaller mean squared error. Therefore, the model with the highest R^2_{OOS} is considered the best model to use.

[1]The choice of using historical average for branching is a standard choice in training random forests. However there have been other designs. For example, Goulet Coulombe (2020) modifies the standard random forest to fit an ensemble of trees which have a *linear model* in each terminal node, rather than a constant for forecasting inflation.

However, the argument of using sample statistics to determine the predictive ability of a model has little to no statistical background in that the judgment relies on their numerical values without any proper hypothesis testing procedure. In the econometrics literature, there has been many seminal papers published in recent years that allows us to test the null with the correct size and power for comparing the accuracy of the forecasts from different models. These tests have not yet been used in research fields outside of theoretical econometrics. Our contribution to the field of empirical finance is therefore introducing the new tools from econometrics literature for a better forecast evaluation of models.

R^2_{OOS} is a naive but simple approach that is used widely used in empirical finance. For example, in GKX, R^2_{OOS} is calculated using the *0forecast* as the benchmark for the ML models to beat. Other papers such as Atanasov, Møller, and Priestley (2020), Jondeau, Zhang, and Zhu (2019), and Rapach, Ringgenberg, and Zhou (2016) use the historical mean as the benchmark to beat instead. The benefit of this approach is that using the test from Clark and West (2007) the R^2_{OOS} can be tested with normal critical values. However, this can only be done with nested linear models, where nested meaning the challenger model must be linear and contains a constant. Because ML models do not generally agree with this assumption, we do not test the significance of R^2_{OOS} in our paper.

GKX uses *0forecast* as the benchmark to assess predictive performance for individual excess stock return forecasts using R^2_{OOS}. This choice was made such that the benchmark is harder to beat for the challenger ML models. Empirically it is well known that the *0forecast* often outperforms historical mean model by a large margin when used to predicting excess returns in the stock market. This is because the historical mean of a stock return is often too noisy that it is easy to beat its forecasting performance.

4.1 R^2_{OOS}

R^2_{OOS} of Campbell and Thompson (2008) shows the percentage of a model's fit out-of-sample relative to the choice of a benchmark model during the testing sample period $\mathbb{T}_3$. It can vary between -1 and 1 and we calculate this statistic for each models studied in our paper. We use *0forecast* as the benchmark similar to GKX to provide our ML models a meaningful benchmark to beat. Outperforming historical mean model is typically a less challenging task for the challenger models studied here.

$$R^2_{\text{OOS},j} = 1 - \frac{\sum_{(i,t)\in\mathbb{T}_3} \left(\sum_{\tau=0}^{h-1} r_{i,t+\tau} - f_j(z_{i,t-1}, \hat{\theta}) \right)^2}{\sum_{(i,t)\in\mathbb{T}_3} \left(\sum_{\tau=0}^{h-1} r_{i,t+\tau} \right)^2} \tag{13}$$

First, we pool forecast errors for each firm to calculate the mean squared forecast error from model j and the *0forecast*. We then measure the out-of-sample fit by the reduction in mean squared error given by model j relative to *0forecast* by (13). Outperforming the benchmark on average is reflected by a positive R^2_{OOS} value, and a negative value indicates the relative underperformance of model j. This is not a statistical test but given its popularity among researchers, we report them in our paper alongside with the test results.

4.2 The test for superior predictive ability and the model confidence set

We now introduce the first statistical test used for evaluating our ML model's predictive ability, the test for superior predictive ability (SPA). The test was proposed by Hansen (2005) as a way to improve the test proposed by White (2000) called *the reality check for data snooping* or simply RC test. They both share the same framework for comparing multiple forecasting models and proposing a test for SPA given a priori chosen benchmark model. Whereas the RC test is based on wald-type test with the entire covariance matrix having to be estimated, the test for SPA studentized the test statistic and only require the estimation of the diagonal term of the covariance matrix. This choice showed increase the power relative to the RC test shown by simulation.

We use squared loss as our choice of loss function to measure a prediction's accuracy for all our ML models. All of the statistical tests we use in our paper can be used using a different choice of loss function. A popular alternative to squared loss is the huber loss function. It is a hybrid of squared loss and absolute loss where for small prediction errors the squared loss is applied, and the absolute loss is used for relatively large prediction errors. The threshold to defining the boundary between two loss function is determined as a hyperparameter. We defer exploring the sensitivity of the test results to the choice of loss function to future research.

Given h number of months for the return to be realized at time $t+h$, N_t is the number of cross-sectional prediction errors at time t. Let $\mathbb{L}_{j,t+h-1}$ be the cross-sectional average of squared forecast error

between the forecast made by model j as $f_j(z_{i,t-1}, \hat{\theta}), i = 1, 2, \cdots, N_t$ and the observed multi-horizon return $\sum_{\tau=0}^{h-1} r_{i,t+\tau}$ out-of-sample.

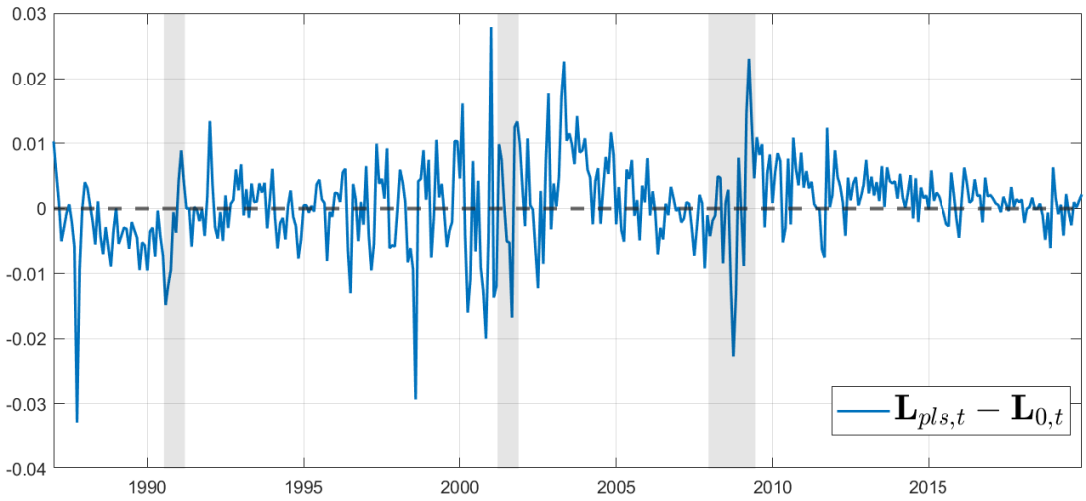
$$\mathbb{L}_{j,t+h-1} = \frac{1}{N_t} \sum_{i=1}^{N_t} \left[\left(f_j(z_{i,t-1}, \hat{\theta}) - \sum_{\tau=0}^{h-1} r_{i,t+\tau} \right)^2 \right] \tag{14}$$

The test for SPA defines that the model j has superior predictive ability over another model j' when the squared forecast error of model j is smaller than that of model j' on average. 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.

$$\mathbb{E}(\Delta\mathbb{L}_{j,j',t+h-1}) < 0 \tag{15}$$

To illustrate the idea of the SPA, we use figure (2) to plot the difference between the *partial least square* model's cross-sectional average of the squared forecast error and that of the *0forecast* for monthly forecasts with $h = 1$. Their difference is $\mathbb{L}_{pls,t} - \mathbb{L}_{0forecast,t}$ or equivalently, $\Delta\mathbb{L}_{pls,0forecast,t}$. During periods

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 0forecast model. Forecast errors from individual firms are cross-sectionally averaged for each month, and we take the average difference between the two models from January 1987 to December 2019.

where $\Delta\mathbb{L}_{pls,0forecast,t} < 0$ on figure (2), the partial least squares model has a smaller squared error thus outperforming 0forecast. The part of the graph where $\Delta\mathbb{L}_{pls,0forecast,t} < 0$ is below the 0 dotted line. If

$\mathbb{E}(\Delta\mathbb{L}_{pls,0forecast,t}) < 0$, we say that partial least squares has SPA over 0forecast.

Prior to using the test of SPA as well as all other statistical tests used in our paper, we check for the existence of unit-root for the time series $\Delta\mathbb{L}_{j,j',t+h-1}$ for all pairs of models. This is done as all the tests in this papers assume stationary of $\Delta\mathbb{L}_{j,j',t+h-1}$. We note that this does not mean that the data and forecasts themselves also need to be stationary.

Additionally, as all our models are of additive prediction error regressions in the form of equation (4), the error terms are additive so the tests need to use the correct form of standard error for valid inference. We address the serial correlation among the additive errors for all our tests using Newey-West standard errors from Newey and West (1987) with lag choice of $h-1$.

Now we formally define the test of SPA. Given a pair of models $(j, j') \in \mathbb{J}$, we test significance for all model pairs under null hypothesis that model j has SPA over model j'. We also include *0forecast* model for the analysis.

$$H_0 : \mathbb{E}(\Delta\mathbb{L}_{j,j',t+h-1}) \leq 0 \tag{16}$$

Let T_{SPA} be the test statistic for test of SPA, and $\bar{d}_{j,j',t+h-1}$ the difference between mean squared forecast error of model j and j' during testing period. With $\bar{d}_{j,j',t+h-1} = 1/\mathbb{T}_3 \sum \Delta\mathbb{L}_{j,j',t+h-1}$, we describe the testing procedure.

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

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

2. Bootstrap T_{SPA} with recentering to $\hat{\mu}^c$ where

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

When model j has the better predictive performance, $\bar{d}_{j,j',t+h-1}$ is negative. The test statistic is normalized to 0 and we conclude that we found no evidence against the null hypothesis of $H_0 : \mathbb{E}(\Delta\mathbb{L}_{j,j',t+h-1}) \leq 0$ and it 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 algorithm of Politis and Romano (1994) for bootstrap implementation.

In addition to testing the SPA of models in a pairwise manner, we construct the model confidence set, or the MCS, from Hansen, Lunde, and Nason (2011). MCS provides a method to identify a model of a set of models that has SPA to all other models, with a chosen level of confidence. The procedure of constructing the MCS ensures that, under the null hypothesis that all models have equivalent predictive abilities, the probability of incorrectly excluding at least one model from the MCS is no greater than the chosen significance level. We construct MCS based on all the ML models studied with a 5% significance level to collect a set of models with SPA for an *all-for-one comparison*.

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.

$$\mu_{j,j',h-1} = \mathbb{E}(\Delta\mathbb{L}_{j,j',t+h-1}), \quad \forall j, j' \in \mathbb{J} \tag{17}$$

For all model pairs, the set of superior models $\mathbb{M}^*$ is defined as a set of models that has SPA to all other models $\mathbb{J}$

$$\mathbb{M}^* = \{j \in \mathbb{J} : \mu_{j,j',h-1} \leq 0, \forall j' \in \mathbb{J}\} \tag{18}$$

Using the MCS algorithm in Hansen, Lunde, and Nason (2011), we construct the set of superior models and use the block bootstrap for critical values as the asymptotic distribution is not standard with nuisance parameters.

1. Choose the length of the block for the bootstrap and the significance level for the MCS.
2. Calculate the t-statistic $t_{j,j'} = \frac{\bar{d}_{j,j',t+h-1}}{\hat{\text{var}}(\bar{d}_{j,j',t+h-1})}, \forall (j, j') \in \mathbb{J}$.

3. Test the null hypothesis $H_{0,mcs} : \mu_{j,j',h-1} = 0, \forall j, j' \in \mathbb{J}$ using the test statistic

$$T_{max,\mathbb{J}} = \max_{(j,j') \in \mathbb{J}} |t_{j,j'}|$$

4. Compare it with the $(1 - \alpha)$ quantile of the bootstrap distribution as critical value. If the test statistic exceeds the critical value, eliminate that model from the set $\mathbb{J}$.

5. Repeat the model elimination starting from step 3, by now using the smaller set of models $\mathbb{J}'$ that does not contain the previously eliminated models. Sequentially eliminate the models inside the set until the null hypothesis is not rejected.

6. The remaining models at the end of the algorithm form the desired MCS.

The MCS procedure does not require a benchmark to be specified, which is very useful in applications without an obvious benchmark. The MCS procedure thus has the advantage that it can be employed for model selection. Additionally, the asymptotic familywise error rate (FWE), which is the probability of making one or more false rejections, is bounded by the $1 - \alpha$ level.

A drawback of test of SPA is that its null hypothesis is a composite hypothesis as defined in equation (16). The null is defined by inequality constraints for each model pair, which affect the asymptotic distribution of the SPA test statistic. The binding inequality constraints create a nuisance parameter problem and this makes it difficult to control the size of the test that in turn induces a loss of power. For this reason, we only use the test of SPA for pairwise comparison of models.

4.3 Conditional Superior Predictive Ability

Finally, we test for the conditional superior predictive ability, or CSPA, of a model using the testing procedure from Li, Liao, and Quaedvlieg (2022) (LLQ). Evaluating the CSPA of a model means that we test the superior predictive ability of a model to other models, *conditional* on a chosen state variable. The conditioning state variable can be chosen by the researcher dependent on what one wishes to test.

The purpose of CSPA is to test the state-dependent predictive performance among our ML models. The benefit of using conditional evaluation approach of models is particularly relevant when the forecast models perform similar on average, but behave differently conditionally. In other words, models'

predictability can vary when forecasts are made conditional on certain economic states. In the example given by LLQ, one can track forecast performance between models through business cycles by setting the conditioning state to be a cyclical indicator such as GDP growth or unemployment rate. This is not possible to test when models are compared only for their overall average performance.

The idea of conditional forecast evaluation has been explored in previous research papers before LLQ. The first paper to explore this idea was Giacomini and White (2006) and they propose to test the conditional equal predictive ability of models. The null hypothesis is that conditional expected loss functions of different forecast methods are the same across all conditioning states given a choice of a state variable. For this, the functional form conditional expected loss function needs to be approximated. They impose the conditional expected loss to be finite-dimensional giving a simple wald-type two-sided test statistic.

When testing for SPA of constructing MCS, we only compare the average relative performance of models during the testing period. This approach flatten certain pockets of periods where the predictive performance of a model do not agree with the outcome of the hypothesis tests for superior predictive ability. This is because when defining the unconditional expectation of the difference of mean squared forecast error between models from equation (17), we implicitly have integrated out all heterogeneity in the models' average loss during the testing period in the data.

We can consider the previous test for evaluating SPA of models to be a special case of CSPA where *unconditional* tests of superior predictive ability were made. By setting a vector of ones as the conditioning state variable, we can use CSPA test also to test the model's superior predictive ability or construct the MCS. However, we do not use the test for CSPA for that purpose of unconditional evaluation. It is recommended that when testing for the model's predictive ability unconditionally, testing the superior predictive ability by Hansen (2005) or constructing the MCS of Hansen, Lunde, and Nason (2011) are better choices as they would be a more powerful approach.

The null hypothesis of CSPA in our paper 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[2]. Given a forecast model $j \in \mathbb{J}$, we test the null of CSPA

[2]Unlike the original null hypothesis of CSPA stated in LLQ, the direction of the inequality sign in our article is switched to be coherent with our analysis of testing SPA and constructing MCS.

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. We have,

$$H_0 : \mathbb{E}(\Delta\mathbb{L}_{j,j',t+h-1}|X_t = x_t) \leq 0, \ \ \forall j' \in \mathbb{J}, X_t \in \mathbb{X} \tag{19}$$

where $\mathbb{X}$ is the support of the state variable. The null hypothesis of CSPA states that a chosen model j performs equally well as all other competing models, not only on average, but also through the ups-and-downs of the economy. This imposes a much more stringent requirement on the model in that it must weakly dominate all other models uniformly across the state space spanned by the conditioning state variable.

The conditioning state must be a scalar for the CSPA test, and we have chosen them to be either (1) business cycle/economic activity indicators, (2) financial conditions/stress indicators, and (3) uncertainty indices of the US economy. Table (1) lists the different state variables used in our paper as well as their sources in parenthesis. We report our CSPA test results using all the state variables individually using their respective mnemonic, for robustness of the test results.

Table 1: List of state variables.

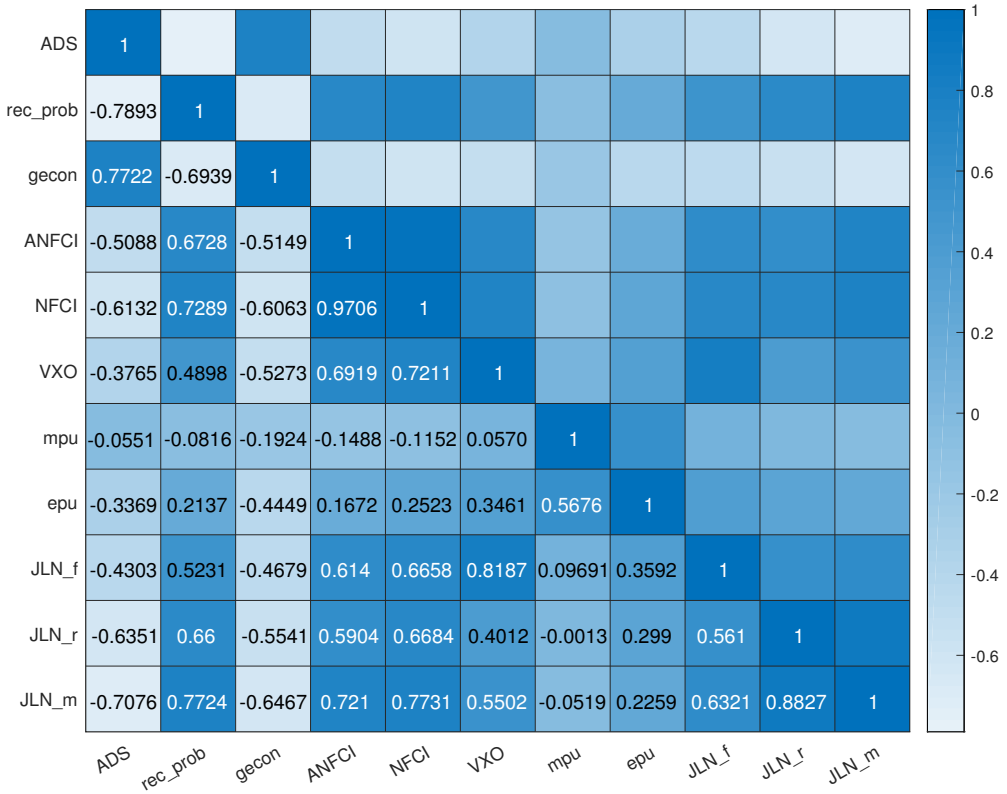
Mnemonic	**Description**
	Business cycle/economic activity indicators
ADS	Aruoba-Diebold-Scotti index(Philadelphia Fed)
rec_prob	Smoothed U.S. Recession Probabilities(St Louis FRED)
gecon	Global Economic condition indicator (Baumeister, Korobilis, and Lee (2022))
	Financial conditions/stress indicators
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 (Husted, Rogers, and Sun (2020))
epu	Economic Policy Uncertainty(USA); news based index (Baker, Bloom, and Davis (2016))
JLN_f	Financial Uncertainty (Jurado, Ludvigson, and Ng (2015))
JLN_m	Macroeconomic Uncertainty (Jurado, Ludvigson, and Ng (2015))
JLN_r	Real Uncertainty (Jurado, Ludvigson, and Ng (2015))

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. The mnemonic is also used for figure (3).

The figure (3) is the heatmap of correlations across all the conditioning state variable used for testing

CSPA from January 1987 to December 2019[3]. 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 for testing. 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 useful information for our forecast evaluation. Suppose a model has shown CSPA for all conditioning state variables. In that case, we consider it to be a strong indication that its predictive ability is robust to different states of the economy.

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.

To implement the test, the conditional expectation function $\mathbb{E}(\Delta\mathbb{L}_{j,j',t+h-1}|X_t = x_t)$ is approximated via a nonparametric series regression of $\Delta\mathbb{L}_{j,j',t+h-1}$ 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 Quaedvlieg (2022) provides an algorithm for implementation. An in-depth discussion of the algorithm is beyond the scope of this paper, and we simply utilize LLQ's test for the forecast evaluation of our ML models[4]. We take

[3]For semi-annual return forecasts, uncertainty indices for Jurado, Ludvigson, and Ng (2015) are not available.
[4]Please refer to *Algorithm 1* in Li, Liao, and Quaedvlieg (2022) for the full description.

serial correlation into account for the test by using Newey-West standard errors with the lag $h-1$ similar to our previous tests.

5 Data

To forecast multi-horizon excess returns, we leverage an extensive collection of (1) firm characteristics that have been used for return prediction in the literature, (2) US macroeconomic predictors used for the stock market return prediction, and (3) the interaction terms between (1) and (2). The seminal paper that serves as the foundation for our paper's data construction is that of GKX. GKX employed 94 firm characteristics from Green, Hand, and Zhang (2017), eight macroeconomic predictors from Welch and Goyal (2007), and 74 dummy variables to classify each firm based on their respective industry sectors. They defined these industry sectors using the first two digits of the US Standard Industrial Classification (SIC) Codes. Using data spanning March 1957 to December 2016, GKX predicted individual monthly excess returns using various ML models and evaluated the out-of-sample fit of each model. In our analysis, we utilize the data from GKX but also augmented with more recent observations of predictors and returns up to December 2019[5].

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

We build upon their work by not only extending the forecasting exercise to predict not only monthly excess returns of individual firms but also their multi-horizon excess returns. Furthermore, we repeat our forecasting exercise and testing for firm groups delineated by their industry sector code. Rather than considering the detailed SIC codes in multiple digits, we focus on the first digit of each firm's SIC code only, which allows us to classify US industries into ten distinct sectors. Table (2) presents these classifi-

[5]The predictor data for our article was provided by the authors and is available on their website. However, accessing it necessitates high-performance computing capabilities and substantial storage due to the volume of the data and computational intensity during model estimation.

cations, and we employ the same set of forecasting models as well as the statistical testing methodology to find SPA and CSPA of models within each industry.

Let sic_k denote a set of firms whose first digit of SIC is equal to k. For example, a set of firms belonging to *real estate industry* is labeled as sic_6 as the industries with $k = 6$ belong to *finance, insurance and real estate* sector of the economy. Given h number of months for the return to be realized at time t, we have N_{t,sic_6} number of firms in this *real estate industry*. For a particular ML model j, we define the cross-sectional average of squared forecast error for each industry as $\mathbb{L}_{sic_k,j,t+h-1}$ in equation (20). The sample average $\mathbb{L}_{sic_k,j,t+h-1}$ during the testing period will be used when testing for SPA and CSPA of models for individual firms well as for each industry.

$$\mathbb{L}_{sic_k,j,t+h-1} = \frac{1}{N_{t,sic_k}} \sum_{i=1}^{N_t} \left[\left(f_j(z_{i,t-1}, \hat{\theta}) - \sum_{\tau=0}^{h-1} r_{i,t+\tau} \right)^2 \cdot \mathbf{1}_{sic_k}(i) \right], \quad k = 0, 1, 2, \cdots, 9. \tag{20}$$

$$N_{t,sic_k} = \sum_{i=1}^{N_t} \mathbf{1}_{sic_k}(i), \quad N_t = \sum_{k=0}^{9} N_{t,sic_k}$$

$$\mathbf{1}_{sic_k}(i) = \begin{cases} 1 & , \text{firm } i\text{'s first digit of SIC is } k. \\ 0 & , otherwise \end{cases} \tag{21}$$

Monthly firm equity returns data of firms are collected from the database of the Center for Research in Security Prices, or CRSP. We consider firms listed in the *New York Stock Exchange*(NYSE), the *American Stock Exchange*(AMEX or also known as NYSE American), and the *National Association of Securities Dealers Automated Quotations*(NASDAQ). Each of these exchange markets has its own listing requirements and operates under the regulatory purview of the *U.S. Securities and Exchange Commission*(SEC). They all play pivotal roles in the U.S. financial system, enabling companies to raise capital by issuing shares to the public and providing venues for investors to buy and sell their shares. Our analysis is specifically focused on the firms listed on these exchanges.

We collect the panel data for their excess returns, as their firm-level characteristics, such as valuation ratios and stock price trends, from January 1960 to December 2019. All characteristics are used in the empirical finance literature for forecasting stock returns of firms[6]. Each firm's excess returns is calculated as the continuously-compounded stock returns minus the risk-free rate. For the latter, we use the US

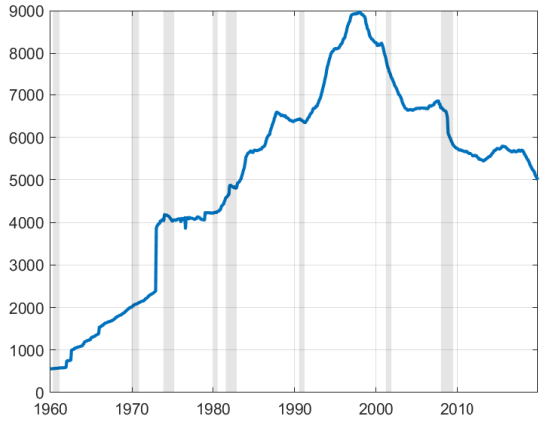
[6]Refer to the appendix of Gu, Kelly, and Xiu (2020) for the full list of references for each firm-level characteristic.

treasury bill rate as a proxy.

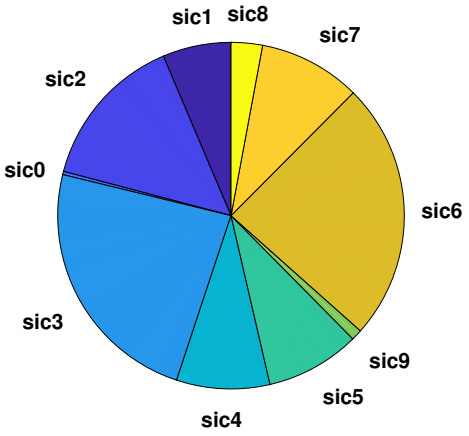
For each month, the number of firms listed on the stock exchange market changes due to new firms entering the market and some existing ones being delisted. Figure (4a) displays the changes in the number of stocks N_t in our data from January 1960 to December 2019. This is needed when we take the cross-sectional average of squared forecast error of a model. Figure (4b) illustrates the proportion that each industry sector represents in our sample from January 1960 to December 2019. The proportion for each industry sector k is calculated as the pooled average of each industry sector. For a firm i, using the indicator function in equation (21), the pooled average is defined as $\frac{1}{T}\frac{1}{N}\sum_{t=1}^{T}\sum_{i=1}^{N_t}\mathbf{1}_{sic_k}(i)$. The largest proportion of industry in our sample is *Manufacturing* with SIC codes 2 and 3, while the smallest is *Agriculture, Forestry, and Fishing* with SIC code 0.

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 the data collected, we define the set of predictors used for all our ML models. First, we have in total 94 firm-level characteristics from our data with over 3 million pooled observations[7] from January 1960 to December 2019. For firm i at time t, let $c_{i,t}$ be a 94 by 1 vector of firm characteristics with $i = 1, \cdots N_t$, $t = 1, \cdots T$. Second, we incorporate eight macroeconomic predictors from Welch and Goyal (2007). In there research the predictors were used for predicting aggregate market equity premium. They

[7]Each firm characteristic is winsorized monthly at the 1% and 99% percentiles the distribution.

consist of various market stock characteristics and bond interest-related variables. 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, m_t is an eight by one vector of these macroeconomic predictors.

With $c_{i,t}$ and m_t defined, we construct the vector of predictors $z_{i,t}$ that are used for model estimation. We append a constant in addition to the vector m_t and use the kronecker product between $\begin{bmatrix}1,\ m_t\end{bmatrix}'$ and $c_{i,t}$ to define $z_{i,t}$ in equation (22). Let P be the total number of predictors used and $z_{i,t}$ be a P by one vector. P is equal to $94 * (8 + 1) = 846$ which includes the interaction terms between $c_{i,t}$ and m_t.

$$\underbrace{z_{i,t}}_{P\times 1} = \underbrace{\begin{bmatrix}1,\ m_t\end{bmatrix}'}_{9\times 1} \otimes \underbrace{c_{i,t}}_{94\times 1} \tag{22}$$

For our out-of-sample analysis, we employ a rolling scheme. Initially, we use a 15-year training period from January 1960 to December 1974 for parameter estimation, followed by a 12-year validation period from January 1975 to December 1986 to choose model hyperparameters, if necessary. We then forecast excess returns for the entirety of 1987 given the multi-horizon h. Subsequently, we roll our training and validation periods forward by one year, establishing a new 15-year training period from January 1961 to December 1975, and a 12-year validation period from January 1976 to December 1987. The forecasting then proceeds for the full year of 1988. This process is repeated until we reach the end of the sample.

Figure (5) illustrates our out-of-sample scheme. We divide our full sample into three mutually exclusive subsets, based on their observed time, at the beginning of the forecasting exercise. We roll forward by 12-months after each iteration.

In figure (5), the topmost line represents the initial division of the sample into training, validation, and testing. The training sample from January 1960 to December 1974, denoted as $\mathbb{T}_1$, is colored in *orange* dots. The 12-year validation period from January 1975 to December 1986 denoted as $\mathbb{T}_2$ is used to selecting hyperparameters of a model and is colored as *blue* dots. With $\mathbb{T}_1$ and $\mathbb{T}_2$, we make a prediction for a point colored in *red*. The prediction is then compared with the actual excess returns to calculate loss. The second topmost line of figure (5) shows the repeated iteration but with rolled over samples by 12-months. By adding 12 new sample points and removing 12 oldest ones, we maintain the same sample

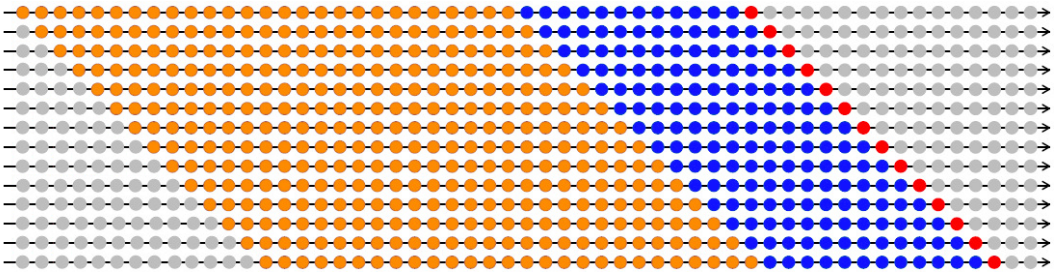


Figure 5: Out-of-sample diagram using a rolling window

Note: The training sample is used to estimate the model and is colored as orange dots. The validation sample is used to choose hyperparameters of models and is colored as blue dots. Finally we make a prediction for a point colored in red, that is used to compare the predictions with the actual excess returns for loss calculation.

size for training and validation, and we use it to predict the next red dot. This process continues until the final observation in our data is reached.

Ultimately, we used a 33-year testing period that spanned from January 1987 to December 2019 denoted $\mathbb{T}_3$, and $T = \mathbb{T}_1 + \mathbb{T}_2 + \mathbb{T}_3$. We make return predictions based on the number of months to realization h. In the subsequent section, we will compare our forecasts with the observed excess returns from this testing period and calculate the squared error loss for each of our models.

6 Results

In this section, we evaluate the predictive ability of ML models using the statistical tests outlined in section (4). We forecast for multi-horizon excess returns using ML models described in section (3). For each horizon, we use h to denote the number of months until the return is realized. Specifically, $h = 1$ represents monthly, $h = 3$ is quarterly, $h = 6$ is semi-annual, and $h = 12$ indicates annual excess returns.

First, we evaluate the forecast performance of ML models by their ability to predict individual firms' multi-horizon excess returns on average. For a single forecast model j, this entails predicting each firms' excess returns out-of-sample, resulting in an unbalanced panel of N_t by $\mathbb{T}_3$ pooled squared loss. We then take the simple average per row of the panel, producing a column vector. This is the time series of the cross-sectional averages of the squared loss from the predicting individual firm's returns. This is $\mathbb{L}_{j,t+h-1}$ in equation (14).

Table (3) displays the out-of-sample fit statistic R^2_{OOS} for each forecast model; *0forecast*, elastic net, LASSO, PCR, PLS, ridge, and random forest. As described in section (4), R^2_{OOS} is calculated with *0forecast* as a benchmark model so the R^2_{OOS} for *0forecast* is omitted. We also have the model confidence set with 5% significance level and the estimation of the MCS does not require the choice of benchmark model. Table (3) shows the statistics for all h values.

We observe that as the forecast horizon h lengthens, R^2_{OOS} increases for most of the models. This agrees with the general observation in the literature that accuracy of the return prediction increases with h. Apart from predicting monthly excess returns with $h = 1$, our results show that the random forest model provides a better out-of-sample fit relative to the others and is the only model included in the MCS. This suggests that when forecasting multi-horizon returns for firms altogether, the random forest model is the preferred choice among the models we tested in our paper.

Table 3: All firms: R^2_{oos}, 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.

We observe that the ridge model exhibits a poor out-of-sample fit across all horizons. We conclude that this is due to the sheer number of predictors used; shrinkage towards zero alone was not enough for a more precise prediction. This is further supported by noting that the out-of-sample fit between the elastic net model and the LASSO model is very similar across all horizons. This observation suggests that the elastic net primarily benefited from variable selection, which reduced the number of parameters estimated in the model.

Table (4) displays test results for SPA between model pairs with 5% significance. The test for SPA is reported alongside with MCS from Table (3) as this test will have better power for pairwise comparisons. 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.

Table 4: All firms: SPA results(pairwise)

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

Note: For the model j, in rows, that has SPA over the model j', in columns, its name is written. For example, at $h = 3$, between 0forecast and elastic net model, the latter has SPA.

We can see that the random forest model demonstrates superiority by having SPA over most other individual models. This observation is consistent with the findings presented in Table (3). There are also noteworthy results when considering dimension reduction techniques. For instance, methods like partial least squares have shown SPA advantages. Similarly, when we examine model selection methods, both the elastic net and LASSO models exhibit SPA over the *0forecast* and certain other models. Yet, when combining all these findings and translating them into practical advice, the random forest model would be the preferred choice for this return prediction exercise.

Next, we report test results for the models' conditional superior predictive ability (CSPA) over other models. We test the null hypothesis that model j has CSPA over *all* competing models uniformly given a conditioning state variable X_t.

$$\mathbb{E}(\Delta\mathbb{L}_{j,j',t+h-1}|X_t = x_t) \leq 0, \ \ \forall j' \in \mathbb{J}, X_t \in \mathbb{X} \tag{23}$$

This test is not a pairwise comparison of models. Instead, it is a test where the model is compared

against all other models simultaneously, conditional on the chosen state variable. We conduct the test for CSPA individually for each of the eleven state variables in our paper.

The results are displayed in Table (5). A ($\star$) inside the table for a specific row and column pair indicates the rejection of the CSPA of the model j with 5% significance level. The names of each model j being tested are written on the left column, and the names of each state variable used per the CSPA test are written on the top row. For example, in $h = 12$, the random forest model demonstrates CSPA over all other forecast models uniformly for all our state variables. The uncertainty indices from Jurado, Ludvigson, and Ng (2015) for semi-annual returns are unavailable and are thus not reported here.

Table 5: All firms: CSPA results

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

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

We note that, for almost all values of h and the state variables, the random forest model consistently demonstrates CSPA over all other models. This is indicated by the absence of ($\star$) marks in the rows for *RF* in Table (5), with the exception of the state variables *ANFCI* and *vxo* for $h = 1$. The model *0forecast* achieved CSPA for all state variables when $h = 1$ predicting monthly excess returns. This observation

aligns with the estimated MCS presented in Table (3). This complements the results from before and provides more evidence that the random forest model is a good choice of model for return prediction.

Examining other models, the partial least squares model shows CSPA except for $h = 1$, and for $h = 12$ when tested with the financial uncertainty state variable *JLF_f*. Both the elastic net model and LASSO display similar CSPA across all h values. In contrast, the ridge model is almost always rejected for all horizons and state variables, underscoring the importance of the variable selection process in the data for effective prediction.

Next, we evaluate the forecast performance of ML models based on their ability to predict multi-horizon excess returns, on average, for firms within the same industry sector. Let the loss $\mathbb{L}_{sic_k,j,t+h-1}$, as defined in equation (20), be the time series of the cross-sectional averages of the squared loss from the predicting excess returns of firms belong to the industry sic_k. This loss serves as the primary criterion for evaluating the models' predictive performance. Industry sectors are defined by the first digit of a firm's Standard Industrial Classification (SIC) code, yielding a total of ten sectors. For the sake of simplicity, we omit the results from the pairwise SPA test here and instead rely on the MCS for testing SPA over all other models unconditionally.

Table (6) displays the out-of-sample fit and MCS for each industry sector and horizon. Industries are labeled in the far-left column as sic_k with $k = 0, \cdots, 9$. As an example, the rows of Table (6) labeled as sic0 correspond to the out-of-sample fit and MCS for firms with a first digit of their SIC code being 0, belonging to the *Agriculture, Forestry, and Fishing* sector in the US economy.

Compared to Table (3), where we observed that the MCS included only *0forecast* for $h = 1$ and the random forest for all other horizons, we now see some other models are included in the MCS. First, we observe that for industries sic3 to sic 6, the random forest model has a good out-of-sample fit and stands as the exclusive member of MCS for all horizons. Conversely, for industries sic7 and sic9, its fit is inferior compared to other models and it is never included in the MCS. Notably, groups with a first-digit SIC code of 3, 4, 5, or 6 constitute the majority of firms in Figure (4). This change in the model fit would have remained undetected had we not categorized our firms into distinct groups.

Table 6: By industry: R^2_{oos}, MCS results

		h=1		h=3		h=6		h=12	
		R^2_{oos}	MCS	R^2_{oos}	MCS	R^2_{oos}	MCS	R^2_{oos}	MCS
	0forecast	-	✓	-		-		-	
	Enet	-0.0095		0.0024		0.0161		0.0267	
	LASSO	-0.0090	✓	0.0023		0.0149		0.0257	
sic0	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	
	0forecast	-	✓	-		-		-	
	Enet	-0.0211		0.0191	✓	0.0142	✓	0.0069	
	LASSO	-0.0212		0.0191	✓	0.0139		0.0084	
sic1	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	
	0forecast	-		-		-		-	
	Enet	0.0090	✓	0.0086		0.0069		0.0074	
	LASSO	0.0089		0.0086		0.0065		0.0099	
sic2	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	✓
	0forecast	-		-		-		-	
	Enet	0		0.0175		0.0201		0.0269	
	LASSO	0		0.0175		0.0199		0.0281	
sic3	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	✓
	0forecast	-		-		-		-	
	Enet	0.0095		0.0115		0		-0.0012	
	LASSO	0.0094		0.0115		0		0.0035	
sic4	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	✓
	0forecast	-		-		-		-	
	Enet	0.0154		0.0112		-0.0033		-0.0023	
	LASSO	0.0153		0.0111		-0.0038		0.0032	
sic5	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	✓
	0forecast	-		-		-		-	
	Enet	0.0115		0.0138		0.0138		0.0202	
	LASSO	0.0114		0.0137		0.0132		0.0197	
sic6	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	✓
	0forecast	-		-		-		-	
	Enet	0.0151		0.0101		0.0054		0.0075	
	LASSO	0.0150		0.0101		0.0052		0.0108	✓
sic7	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	
	0forecast	-		-		-		-	
	Enet	0.0137		0.0225		0.0344		0.0498	
	LASSO	0.0137		0.0226		0.0347		0.0501	✓
sic8	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	✓
	0forecast	-		-		-		-	
	Enet	0.0472		0.0100		-0.0043		0	
	LASSO	0.0472		0.0100	✓	-0.0046		-0.0012	
sic9	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 on a model's CSPA over all other models given a state variable forecasting multi-horizon returns for firms belong to the same industry. For each model, we tally the number of times it demonstrates CSPA over all competing models for a given conditioning state variable.Table (7) presents the results. The model with the highest count within a specific industry sector for h is highlighted with its count written in bold. For instance, in the *Agriculture, Forestry, and Fishing* sector, identified by the SIC code 0, the partial least squares model displays CSPA over all other models across all 11 state variables for $h = 3$. Detailed CSPA test results for individual state variables per model are provided in the appendix.

Table 7: By industry: CSPA results

		sic0	sic1	sic2	sic3	sic4	sic5	sic6	sic7	sic8	sic9
	0forecast	**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
h=1	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
	0forecast	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
h=3	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
	0forecast	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
h=6	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
	0forecast	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
h=12	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) presents the CSPA test results of models for forecasting multi-horizon returns in different industry sectors. For the *Transportation, Communications, Electric, Gas, and Sanitary service*, *Wholesale Trade & Retail Trade*, and *Finance, Insurance, and Real Estate* sectors, identified by SIC codes 4, 5, and 6 respectively, the random forest model demonstrates CSPA over other models across all 11 state variables. Given the significant proportion of firms in these industries, it is evident that the random forest model still excels in prediction overall.

However, the predictive prowess of the random forest is not consistent. This aligns with the CSPA test results when testing for all firms in the market in Table (5). As an example, for the *Services* sector with SIC code 7, the random forest model rejects the null hypothesis of CSPA across all 11 state variables. More notably, for industries sic7 and sic9 code, the principal component regression model has CPSA over others for certain state variables. This also agrees with the SPA test result in Table (6). Testing solely for the models' CSPA for firms altogether in the market would not have shown this result.

Aside from the *Agriculture, Forestry, and Fishing* industry during monthly returns prediction, the ridge model rejects the CSPA null across all sectors and horizons. This is similar result to the previous forecast evaluation results in Table (6) for MCS estimation. 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.

In sum, no single model, or fixed model group, consistently excels in CSPA across all sectors and horizons. This suggests inherent variability in predictive performance of models across industry sectors. Forecasters should be aware to this variability in forecasting results of ML models. The discussion of why we observe such heterogeneity in the test result will involve an extensive research. This topic will be deferred for future research.

7 Conclusion

We investigated the predictive ability of supervised machine learning models in forecasting multi-horizon returns of individual stock market firms. For this purpose, we utilized an extensive set of predictors derived from empirical asset pricing literature and made predictions for monthly, quarterly, semi-annual, and annual continuously compounded returns minus the risk-free rate.

Our main contribution to the research field is the use of forecast evaluation tests and adding statistical significance to the out-of-sample forecast exercises that typically rely solely on simple sample statistics. We make return predictions for (1) individual firms' excess returns in the market and (2) returns for

groups of firms classified by their industry sectors. We assessed models based on their *superior predictive ability* and *conditional superior predictive ability* as defined by Hansen (2005) and Li, Liao, and Quaedvlieg (2022). For testing a model's conditional superior predictive ability, we used conditioning state variables listed in Table (1) to test if its predictive ability is robust to the varying states of the economy.

First, we evaluated the forecast performance of ML models by their ability to predict multi-horizon excess returns of individual firms in the stock market on average. We use time series of the cross-sectional averages of the squared loss from the predicting individual firm's returns, denoted as $\mathbb{L}_{j,t+h-1}$ in equation (14). We found that partial least squares model and the random forest model generally performed better than others. Notably, our evaluation revealed the significance of variable selection over shrinkage, as evident in the lack of predictive ability in the ridge model compared to LASSO or the elastic net model.

Second, we evaluate the forecast performance of ML models based on their ability to predict multi-horizon excess returns, on average, for firms within the same industry sector. We find that no single model or a set of models has SPA or CSPA uniformly over all other models. This variability can be attributed to the intrinsic heterogeneity among firms from different sectors and how they react to changes in the state of the economy. The researchers who wish to find guidance on what ML model should be used for return prediction can reference the test results in Table (6) and Table (7).

References

Atanasov, V., S. V. Møller, and R. Priestley (2020): "Consumption fluctuations and expected returns," *The Journal of Finance*, 75(3), 1677–1713.

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.

Campbell, J. Y., and S. B. Thompson (2008): "Predicting excess stock returns out of sample: Can anything beat the historical average?," *The Review of Financial Studies*, 21(4), 1509–1531.

Chinco, A., A. D. Clark-Joseph, and M. Ye (2019): "Sparse signals in the cross-section of returns," *The Journal of Finance*, 74(1), 449–492.

Clark, T., and M. McCracken (2013): "Advances in forecast evaluation," in *Handbook of economic forecasting*, vol. 2, pp. 1107–1201. Elsevier.

Clark, T. E., and M. W. McCracken (2001): "Tests of equal forecast accuracy and encompassing for nested models," *Journal of econometrics*, 105(1), 85–110.

——— (2012): "Reality checks and comparisons of nested predictive models," *Journal of Business & Economic Statistics*, 30(1), 53–66.

Clark, T. E., and K. D. West (2006): "Using out-of-sample mean squared prediction errors to test the martingale difference hypothesis," *Journal of econometrics*, 135(1-2), 155–186.

——— (2007): "Approximately normal tests for equal predictive accuracy in nested models," *Journal of econometrics*, 138(1), 291–311.

De Jong, S. (1993): "SIMPLS: an alternative approach to partial least squares regression," *Chemometrics and intelligent laboratory systems*, 18(3), 251–263.

Diebold, F., and R. Mariano (1995): "Comparing Predictive Accuracy," *Journal of Business & Economic Statistics*, 13(3), 253–63.

Efron, B., and T. Hastie (2021): *Computer Age Statistical Inference, Student Edition: Algorithms, Evidence, and Data Science*, vol. 6. Cambridge University Press.

Fama, E. F., and K. R. French (1993): "Common risk factors in the returns on stocks and bonds," *Journal of financial economics*, 33(1), 3–56.

——— (2015): "A five-factor asset pricing model," *Journal of financial economics*, 116(1), 1–22.

Freyberger, J., A. Neuhierl, and M. Weber (2020): "Dissecting characteristics nonparametrically," *The Review of Financial Studies*, 33(5), 2326–2377.

Giacomini, R., and H. White (2006): "Tests of conditional predictive ability," *Econometrica*, 74(6), 1545–1578.

Goulet Coulombe, P. (2020): "The macroeconomy as a random forest," *Available at SSRN 3633110*.

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.

Hansen, P. R., and A. Timmermann (2015): "Equivalence between out-of-sample forecast comparisons and Wald statistics," *Econometrica*, 83(6), 2485–2505.

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.

Jondeau, E., Q. Zhang, AND X. Zhu (2019): "Average skewness matters," *Journal of Financial Economics*, 134(1), 29–47.

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.

McCracken, M. W. (2007): "Asymptotics for out of sample tests of Granger causality," *Journal of econometrics*, 140(2), 719–752.

Mullainathan, S., AND J. Spiess (2017): "Machine learning: an applied econometric approach," *Journal of Economic Perspectives*, 31(2), 87–106.

Newey, W. K., AND K. D. West (1987): "A simple, positive semi-definite, heteroskedasticity and autocorrelationconsistent covariance matrix," Discussion Paper 3.

Patton, A. J., AND A. Timmermann (2012): "Forecast rationality tests based on multi-horizon bounds," *Journal of Business & Economic Statistics*, 30(1), 1–17.

Politis, D. N., AND J. P. Romano (1994): "The stationary bootstrap," *Journal of the American Statistical association*, 89(428), 1303–1313.

Rapach, D. E., M. C. Ringgenberg, AND G. Zhou (2016): "Short interest and aggregate stock returns," *Journal of Financial Economics*, 121(1), 46–65.

Rapach, D. E., J. K. Strauss, J. Tu, AND G. Zhou (2019): "Industry return predictability: A machine learning approach," *The Journal of Financial Data Science*, 1(3), 9–28.

Rapach, D. E., J. K. Strauss, AND G. Zhou (2013): "International stock return predictability: What is the role of the United States?," *The Journal of Finance*, 68(4), 1633–1662.

Romano, J. P., A. M. Shaikh, AND M. Wolf (2008): "Formalized data snooping based on generalized error rates," *Econometric Theory*, 24(2), 404–447.

Romano, J. P., AND M. Wolf (2005): "Stepwise multiple testing as formalized data snooping," *Econometrica*, 73(4), 1237–1282.

Sharpe, W. F. (1964): "Capital asset prices: A theory of market equilibrium under conditions of risk," *The journal of finance*, 19(3), 425–442.

Tibshirani, R. (1996): "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1), 267–288.

——— (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.

West, K. D. (1996): "Asymptotic inference about predictive ability," *Econometrica: Journal of the Econometric Society*, pp. 1067–1084.

White, H. (2000): "A reality check for data snooping," *Econometrica*, 68(5), 1097–1126.

Zou, H. (2006): "The adaptive lasso and its oracle properties," *Journal of the American statistical association*, 101(476), 1418–1429.

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.

8 Appendix

8.1 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. This complements the Table (7) in section (6) by showing which state variables gave the test rejection for each model.

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

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

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
	0forecast				⋆							⋆
	Enet	⋆	⋆	⋆		⋆				⋆	⋆	⋆
	LASSO	⋆	⋆	⋆		⋆				⋆	⋆	⋆
sic0	PCR	⋆	⋆	⋆		⋆		⋆	⋆	⋆	⋆	⋆
	PLS											
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF		⋆									
	0forecast									⋆		
	Enet									⋆		
	LASSO									⋆		
sic1	PCR	⋆	⋆	⋆		⋆	⋆	⋆	⋆	⋆	⋆	⋆
	PLS	⋆			⋆		⋆			⋆		
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF	⋆	⋆	⋆	⋆		⋆					
	0forecast			⋆		⋆	⋆			⋆		⋆
	Enet			⋆		⋆	⋆			⋆		⋆
	LASSO			⋆		⋆	⋆			⋆		⋆
sic2	PCR	⋆	⋆	⋆	⋆	⋆	⋆	⋆		⋆		⋆
	PLS			⋆		⋆	⋆			⋆		⋆
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF	⋆	⋆	⋆	⋆					⋆		
	0forecast		⋆	⋆		⋆		⋆	⋆	⋆	⋆	⋆
	Enet					⋆				⋆		
	LASSO					⋆				⋆		⋆
sic3	PCR	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	PLS					⋆				⋆		⋆
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF	⋆			⋆					⋆	⋆	⋆
	0forecast	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	Enet	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	LASSO	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
sic4	PCR	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	PLS	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF											
	0forecast	⋆	⋆	⋆		⋆	⋆			⋆		⋆
	Enet	⋆	⋆	⋆		⋆	⋆			⋆		⋆
	LASSO	⋆	⋆	⋆		⋆	⋆			⋆		⋆
sic5	PCR	⋆	⋆	⋆		⋆	⋆			⋆		⋆
	PLS	⋆	⋆	⋆		⋆	⋆			⋆		⋆
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF											
	0forecast	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	Enet	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	LASSO	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
sic6	PCR	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	PLS	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF											
	0forecast	⋆				⋆			⋆	⋆		
	Enet					⋆				⋆		
	LASSO					⋆			⋆	⋆		
sic7	PCR	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆		⋆	
	PLS											
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	0forecast	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	Enet	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	LASSO		⋆			⋆				⋆	⋆	⋆
sic8	PCR	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	PLS					⋆				⋆		⋆
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF	⋆	⋆	⋆	⋆	⋆	⋆			⋆		
	0forecast	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	Enet					⋆		⋆	⋆	⋆		
	LASSO					⋆				⋆		
sic9	PCR	⋆		⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	PLS					⋆						
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF	⋆	⋆	⋆	⋆	⋆	⋆			⋆	⋆	⋆

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
	0forecast	⋆							
	Enet		⋆	⋆					
	LASSO		⋆	⋆	⋆				
sic0	PCR	⋆	⋆	⋆		⋆			
	PLS		⋆	⋆	⋆				
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF								
	0forecast								
	Enet								
	LASSO					⋆			
sic1	PCR								
	PLS								
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF								
	0forecast			⋆		⋆			
	Enet			⋆		⋆	⋆		
	LASSO			⋆		⋆	⋆		
sic2	PCR			⋆		⋆	⋆		
	PLS			⋆		⋆			
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF								
	0forecast		⋆			⋆			⋆
	Enet					⋆			
	LASSO					⋆			
sic3	PCR		⋆	⋆	⋆	⋆		⋆	
	PLS					⋆			
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF								
	0forecast	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	Enet	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	LASSO	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
sic4	PCR	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	PLS	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF								
	0forecast			⋆		⋆	⋆		
	Enet	⋆	⋆	⋆		⋆	⋆		
	LASSO	⋆	⋆	⋆		⋆	⋆		
sic5	PCR		⋆	⋆		⋆	⋆		
	PLS	⋆	⋆	⋆		⋆	⋆		
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF								
	0forecast	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	Enet	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	LASSO	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
sic6	PCR	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	PLS	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF								
	0forecast					⋆		⋆	
	Enet					⋆			
	LASSO					⋆			
sic7	PCR					⋆		⋆	
	PLS								
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	0forecast	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	Enet	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	LASSO	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
sic8	PCR	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	PLS					⋆			
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF								
	0forecast			⋆	⋆	⋆			
	Enet	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	LASSO	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
sic9	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
	0forecast	⋆			⋆		⋆		⋆			⋆
	Enet	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	LASSO	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
sic0	PCR	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	PLS											
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF											
	0forecast											
	Enet		⋆	⋆	⋆						⋆	⋆
	LASSO				⋆							
sic1	PCR	⋆	⋆	⋆						⋆		⋆
	PLS											
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF						⋆					
	0forecast	⋆	⋆	⋆		⋆				⋆	⋆	⋆
	Enet		⋆	⋆		⋆				⋆	⋆	⋆
	LASSO		⋆	⋆		⋆				⋆	⋆	⋆
sic2	PCR		⋆	⋆		⋆	⋆			⋆	⋆	⋆
	PLS		⋆	⋆		⋆				⋆	⋆	⋆
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF											
	0forecast		⋆	⋆		⋆	⋆	⋆		⋆		⋆
	Enet			⋆		⋆				⋆		
	LASSO			⋆		⋆				⋆		
sic3	PCR		⋆	⋆	⋆	⋆	⋆	⋆		⋆	⋆	⋆
	PLS			⋆		⋆				⋆		
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF									⋆		
	0forecast	⋆	⋆	⋆	⋆	⋆	⋆	⋆		⋆	⋆	⋆
	Enet	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	LASSO	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
sic4	PCR	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	PLS	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF											
	0forecast	⋆		⋆		⋆		⋆		⋆	⋆	⋆
	Enet	⋆	⋆	⋆		⋆	⋆	⋆				⋆
	LASSO	⋆	⋆	⋆		⋆	⋆	⋆				⋆
sic5	PCR	⋆	⋆	⋆		⋆				⋆		⋆
	PLS	⋆	⋆	⋆		⋆	⋆	⋆			⋆	⋆
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF											
	0forecast	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	Enet	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	LASSO	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
sic6	PCR	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	PLS	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF											
	0forecast					⋆						
	Enet					⋆				⋆		
	LASSO					⋆				⋆		
sic7	PCR					⋆			⋆			⋆
	PLS											
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	0forecast	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	Enet		⋆	⋆		⋆			⋆			
	LASSO		⋆	⋆		⋆			⋆			
sic8	PCR	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	PLS					⋆						
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF											
	0forecast	⋆							⋆		⋆	
	Enet		⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	LASSO	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
sic9	PCR											
	PLS											
	Ridge	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆
	RF					⋆				⋆		

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