Using Machine Learning to Predict Aggregate Excess Returns

Master Thesis in Finance at the Stockholm School of Economics

Authors
Erik Jonsson*
Sebastian Gierlowski Carling†

May 15, 2021

Abstract

In this paper we examine whether standard linear regression and machine learning tools can be used to predict the time series of total returns in excess of the risk-free rate on the S&P500 and FTSE100 indices. We have virtually no success in predicting monthly returns. However, we do have some success in predicting annual returns. Fully 78 out of our 132 attempts at annual return prediction have a positive out-of-sample R^2. Furthermore, we find that this translates to economic gains. We find that in 87 cases, long-only portfolios formed based on our models have a Sharpe ratio higher than a simple buy-and-hold portfolio. We achieve our best result when using US data from 1926 to 2019, to predict annual returns, using the random forest algorithm and a cumulative estimation window. We find that if we use this model to form a long-short portfolio, we have a Sharpe ratio of 0.996. This set up also has an out-of-sample R^2 of 0.564. We use a Diebold-Mariano test to conclude that this performance, relative to benchmark predictions equal to the historical mean, is significant on the 1% level.

Acknowledgments: We would like to express our gratitude towards our thesis supervisor, Assistant Professor Tobias Sichert. His excellent guidance and sincere enthusiasm for the topic inspired and helped us immensely. Moreover, we would like to thank our good friend Fredrik Omstedt, whose tips helped us navigate the world of deep learning.

Keywords: Equity premium, Machine learning, Non-linear models, Penalized linear models, Prediction

Tutor: Tobias Sichert, Assistant Professor with the Department of Finance
Examiner: Jungsuk Han, Associate Professor with the Department of Finance

*23621@student.hhs.se
†50503@student.hhs.se

Contents

1 Introduction

People within the field of finance have long been interested in the question of whether stock returns are predictable. Within academia this interest is largely driven by curiosity about the process underlying stock returns in practice and how that squares with theory. Within business the interest is driven by commercial considerations, if stock returns are predictable there may be opportunities to make money. Consequently, the last century has seen endless attempts at predicting stock returns, both by researchers and by asset managers. Lately, these attempts have to an increasing extent utilized the methods of machine learning.

In this paper, we examine whether we are able to predict stock returns using machine learning algorithms. We use monthly data from the US and the UK to predict aggregate stock returns on both a monthly and an annual basis. For the US we try to predict the return on the S&P500 index and for the UK we use the FTSE100 index. The returns we try to predict are the logarithm of returns in excess of the risk-free rate.

We use combinations of six different datasets, two return horizons, two training methods and eleven statistical algorithms, to make a total of 54 384 predictions. Our results are based on these predictions.

We find that it is hard, but not impossible, to predict returns. In total we make 132 attempts to predict monthly returns and equally as many to predict annual returns.[1] Of these only 13 monthly forecasts are successful in terms of having a positive out-of-sample R^2. Having a positive out-of-sample R^2 means the model makes more accurate predictions than a benchmark consisting of the historical mean return.

We are more successful when predicting annual returns. Fully 78 attempts are successful in terms of out-of-sample R^2. This discrepancy between annual and monthly prediction is in line with

[1] In this context, an *attempt* should not be confused with a *prediction*. The latter is the outcome of a single forecast for a single period, the former is what we call the aggregate results of a combination of data, model estimation method etcetera.

the literature we have studied. Returns should be easier to predict in the long run, see for instance Shiller (2013).

We find that our results translate to economic gains. If we average the performance of our models over different set ups we find that the best performer when it comes to predicting annual returns is the random forest model. In our results, this model is able to achieve good financial performance if implemented to form a trading strategy.

We form two different strategies based on the direction of our predictions. One strategy is a long-only portfolio and the other is a long-short portfolio. The random forest, on average over our set ups, achieves a long-only Sharpe ratio of 0.79. This compares to a market average over our set ups of 0.47.

There is a risk that these results are spurious. We fit many models and there is thus a risk that some of them will perform well due to happenstance. One way in which we address this is by comparing US results to UK results. We find that we are about as likely to succeed in predicting returns in the UK as in the US. However, we find the best performing models are not the same in both countries. This fact increases our concern about "data mining." However, we perform significance testing for some of our results and find that several of our successful forecasts on annual returns hold up using the Diebold-Mariano test.

Finally it is worth putting the results in the context of machine learning versus other methods. We find that machine learning in general does not succeed in outperforming historical benchmarks when it comes to predicting monthly returns. We do however find that this is the case when it comes to annual returns. Furthermore, we find that non-linear models are more successful than linear models when it comes to predicting annual returns.

We structure the rest of the paper in the following way. Section two gives a brief overview of relevant literature. Section three lays out what data we have used. Section four contains a presentation of the method we used to arrive at our results. Section five contains our main results. Section six concludes.

2 Literature review

The finance literature contains volumes of research into the question of whether stock returns are predictable, and if so, how to best predict them. Attempts to formalize the theory around stock price movements can be traced back at least a century to when C. H. Dow posited that dividend ratios could help make predictions and that short-term price fluctuations were likely to reverse in the medium term (Dow and Selden, 1920). Dividend ratios are still a common tool discussed in the literature on return prediction.

In this section we provide a brief overview of relevant research and we split it into four parts. First, what the theory says about aggregate stock market predictability. Second, the history of empirical research into the question of aggregate stock market predictability. Third, the entry of machine learning into the field of stock market prediction. Fourth, a summary of how we hope to contribute to this literature.

2.1 Theory

In 1970, Euguene Fama published his seminal paper *Efficient Capital Markets: A Review of Theory and Empirical Work*. In it, he puts forth three forms of efficiency in financial markets, a framework still used today. First, weak efficiency, prices only reflect historical information implying that returns cannot be predicted from for example historical prices. Second, semi-strong efficiency, prices reflect all currently available public information implying that returns cannot be predicted from any information openly available at the time of prediction. Third, strong efficiency, this is similar to the semi-strong form apart from that it also includes private information making it virtually impossible to predict returns at all (Fama, 1970).

The thinking goes that competition among profit seeking rational agents should drive the price of an asset to equal the present value of its future cash flows. In response to new information becoming available this process should happen near instantaneously. To quote the great Robert

Shiller "As the theory went, because they are rationally determined, they are changed from day to day primarily by genuine news, which is by its very nature essentially unforecastable" (Shiller, 2013).

Since the initial days of efficient market theory, finance has evolved as a field. In 1981, Shiller published an influential paper. In it he noted that if stock prices are to reflect the present values of their cash flows, essentially their dividends, stock prices should probably vary less than realized dividends. This is however not the case. In a graph that has become rather famous, at least so far as graphs from economics papers can be famous, he illustrated that the price of the US stock market has fluctuated more than what he called the rational ex-post price based on its dividends (Shiller, 1981).

Since then, several attempts have been made at explaining why expected returns vary over time. One example is the habit model suggested by Campbell and Cochrane (1999). In brief, they posit that the incorporation of fluctuations in consumption level may explain variations in stock prices. However, Shiller (2013) points out that this model has also failed in providing a conclusive explanation of aggregate movements in stock prices.

2.2 Empirical evidence

For the purpose of our paper, the theory behind whether, how and why stock returns may or may not be predictable is of limited interest. These are important subjects and in a perfect world we would delve into them as well. However, this is primarily an empirical paper and we are concerned with two main questions, are aggregate stock returns predictable, and if so, are they predictable enough to allow for superior returns to be achieved.

We are of course not the first to concern ourselves with this question. The fourth issue of the 21st volume of the Review of Financial Studies can be considered a special edition on return predictability. Going over all the papers in detail would be unnecessary, but to illustrate the disagreement still existent in the field, two papers should be mentioned.

The first one is *The Dog that did not Bark: A Defense of Return Predictability* by Cochrane. He reiterates that if returns are not predictable then dividend growth must be and vice versa. He goes on to argue that it is returns that are predictable. He does this partly by regressing long-term returns on dividend-to-price ratios with the result being statistically and economically significant coefficients (Cochrane, 2008).

The other one is *A Comprehensive Look at the Empirical Performance of Equity Premium Prediction* by Welch and Goyal. They take a broader approach examining the out-of-sample performance of a vast number of models suggested during the prior decades. They come to the conclusion that most of these models perform poorly out-of-sample and that they are volatile (Welch and Goyal, 2008).

In light of these disagreements it is worth pointing out that there is some agreement on that, if returns are predictable, returns are probably more predictable in the long-term (Shiller, 2013).

2.3 Machine learning

Machine learning applies algorithms to data to detect patterns and make predictions. A common feature is that the model is continuously updated as new data is gathered. In recent years there has been a surge in interest in these techniques as computing power and the amount of data available has increased (James et al., n.d.).

In recent years, these methods have been garnering more interest in the field of finance. In particular, they have been hypothesized to be able to make gains when it comes to predicting stock returns, both with regard to the cross section and in the aggregate. There are several reasons for this. One is that even simple machine learning methods make use of tuning of parameters which allows for more sophisticated learning processes compared to just simple iterative linear regression. Furthermore, many models allow for weaker assumptions when it comes to the form of the data generating process of the target variable. This is probably more in line with the complex process underlying the equity risk premium (Gu, Kelly, and Xiu, 2020).

The verdict on machine learning methods in predicting asset returns has so far not been conclusive. Several papers do however find that machine learning can significantly improve upon more traditional models. For instance, Freyberger, Neuhierl, and Weber (2020) are able to produce an out-of-sample Sharpe ratio of 2.75 when predicting the cross section of returns compared to 1.06 for a simple linear model. Furthermore Gu, Kelly, and Xiu (2020) are able to produce a Sharpe ratio of 0.77 when predicting aggregate returns. Their best result is achieved with a neural network model and they hypothesize that this is because these models are good at picking up subtle non-linear relationships. This is supported by the findings of Feng, He, and Polson (2018) who achieve their best results with deep learning models.

2.4 Our contributions

To summarize, the theory says that stock returns should be hard to predict. Nevertheless, in the literature there is an abundance of papers that manage to predict stock returns. However, these results have been disputed as spurious by subsequent papers. Recently, researchers have tried to improve upon models for predicting stock returns by implementing machine learning algorithms. The results have so far been somewhat promising, especially for deep learning and neural networks.

In light of this, we would like to contribute to the field in the following three ways. First, follow in the footsteps of researchers before us by applying machine learning algorithms to more recent US data. Second, try to replicate these results with UK data, thus trying to determine whether the results are spurious. Third, we use iterative tuning of hyperparameters hoping to achieve better results.

3 Data

Throughout the paper we use six different datasets described briefly below.

First, the dataset we would simply like to call "US data, full sample." This is our largest dataset; it spans from December 1926 until November 2019 and contains sixteen variables of US data described in more detail later on.

Second, a dataset we refer to as "US data, VRP subsample." This dataset contains the same variables as US data with the addition of the predictor VRP. It spans from January 1990 to November 2019.

Third, a dataset we refer to as "US data, VRP subsample excluding the VRP." This spans the same exact time period as the second dataset but only contains the same variables as the first dataset.

Fourth, "UK data, full sample." It spans from February 1998 to January 2021 and contains some 12 variables of UK data described in more detail later on.

Fifth, a dataset we refer to as "UK data, VRP subsample." This dataset contains the same variables as UK data with the addition of the predictor VRP. It spans from January 2000 to December 2019.

Sixth, a dataset we refer to as "UK data, VRP subsample excluding the VRP." This spans the same exact time period as the fifth dataset but only contains the same variables as the fourth dataset.

The variables of the aforementioned datasets are now laid out in more detail by country.

3.1 US data

Here we describe how we calculated the variables we used to arrive at our results for the US data. Unless otherwise specified the source of the data is the updated version of the dataset used by Welch and Goyal in their 2008 paper. The dataset contains monthly observations of the S&P500 index, its return and accompanying variables (Goyal, n.d.).

When calculating our predictor variables we have also largely followed the approach suggested by Welch and Goyal (2008).

3.1.1 Variable "period"

This variable contains exactly the information that can be expected from its name. It is simply the year and the month to which the predictors and the target variables are attributable.

3.1.2 Variable "dp"

This variable contains the dividend to price ratio prevailing at the time of the observation. We calculate it as follows:

$$dp_t = \log(D_t) - \log(Index_t) \tag{1}$$

In this equation D is the twelve-month moving sum of dividends and $Index$ is the level of the S&P500 index.

3.1.3 Variable "dy"

This variable describes the dividend yield an investor would have received if they invested twelve months prior. We calculate it in the following way:

$$dy_t = \log(D_t) - \log(Index_{t-12}) \tag{2}$$

In this equation D is the twelve-month moving sum of dividends and $Index$ is the level of the S&P500 index.

3.1.4 Variable "ep"

The variable ep describes the ratio between earnings per share and the price per share. When calculating it we take the following approach:

$$ep_t = \log(E_t) - \log(Index_t) \tag{3}$$

In this case E is the twelve-month moving sum of earnings and $Index$ is the level of the S&P500 index.

3.1.5 Variable "de"

This variable describes the dividend payout ratio, the degree to which earnings are returned to shareholders in the form of dividends. We calculate it as follows:

$$de_t = \log(D_t) - \log(E_t) \tag{4}$$

In this case E is the twelve-month moving sum of earnings and D is the twelve-month moving sum of dividends.

3.1.6 Variable "svar"

The variable *svar* represents the volatility of the S&P500 index and is the sum of squared daily returns. We do no calculations of our own here, the variable is readily available in the dataset.

3.1.7 Variable "bm"

This variable describes the book-to-market ratio of the Dow Jones Industrial Average. We do no calculations of our own here, the variable is readily available in the dataset.

3.1.8 Variable "ntis"

The variable *ntis* is a measure of the issuing activity of corporations in the US. A higher value means more equity issuing activity relative to the market capitalization of firms listed on the New York Stock Exchange and vice versa. We do no calculation of our own here, the variable is readily available in the original dataset.

3.1.9 Variable "tbl"

This variable describes the rate at which the government can conduct its short-term borrowing activity. It is based on the historical rates for US Treasury Bills with different short maturities. Here we do no calculations of our own.

3.1.10 Variable "ltr"

This variable describes the long-term rate of total return on long-term US government bonds. Here we do no calculations of our own. Welch and Goyal retrieve the numbers from Stocks, Bonds, Bills and Inflation (Ibbotson and Harrington, 2020).

3.1.11 Variable "tms"

The variable *tms* is a measure of the *term spread* on US government bonds, that is the difference between the yield on long-term and short-term government bonds. We calculate it in the following way:

$$tms_t = lty_t - tbl_t \tag{5}$$

Here *lty* is the yield on long-term US government securities, based on different maturities over time and we calculate *tbl* as above.

3.1.12 Variable "dfy"

This variable describes the *default yield spread*, the difference between the yield on highly rated corporate bonds and lower rated corporate bonds. We use the following formula to calculate it:

$$dfy_t = BAA_t - AAA_t \tag{6}$$

Here BAA is the yield on corporate bonds rated BAA and similar for AAA.

3.1.13 Variable "dfr"

The variable dfr is a measure of the *default return spread*. It is the difference between the long-term total return on corporate bonds and government bonds. When calculating it, we take the following approach:

$$dfr_t = corpr_t - ltr_t \tag{7}$$

Here we calculate *ltr* as described earlier and *corpr* is a similar measure for corporate bonds. Both are readily available in the dataset.

3.1.14 Variable "infl"

This variable describes the latest published, not seasonally adjusted, inflation rate according to the urban consumer price index. Here we do no calculations of our own, the variable is readily available in the original dataset.

3.1.15 Variable "vrp"

This variable describes the variance risk premium. The variance risk premium is the difference between the volatility implied by option prices and the actual realized volatility. It is defined as follows:

$$vrp_t = IV_t - RV_t \tag{8}$$

In this equation IV is the end of month value of the VIX index squared and then divided by twelve. RV is the sum of squared five-minute log return for the S&P500. We get this data readily prepared from Zhou (n.d.).

3.1.16 Variable "log.mo.ex.ret"

This is our target variable when we predict monthly returns. We calculate it as follows:

$$log.mo.ex.ret = \log\left(1 + CRSP.SPvw_{t+1} - Rfree_{t+1}\right) \tag{9}$$

Here $CRSP.SPvw$ is the value weighted total return for the S&P500. $Rfree$ is the risk-free rate which is just the variable *tbl* divided by 12.

3.1.17 Variable "log.ye.ex.ret"

This is our target variable when we predict annual returns. We take the following approach to calculating it:

$$log.ye.ex.ret = \log\left(\frac{tot.idx_{t+12}}{tot.idx_t} - tbl_t\right) \tag{10}$$

Here *tbl* is the variable we have already mentioned and *tot.idx* is a total return index we calculate based on the $CRSP.SPvw$ variable.

3.2 UK data

In this section we describe how we arrive at the variables in the UK data. Our methods are similar to those for the US data but different in that we use fewer variables and that we have collected the raw data ourselves.

3.2.1 Variable "period"

This variable contains exactly the information that can be expected from its name. It is simply the year and the month to which the predictors and the target variables are attributable.

3.2.2 Variable "dp"

This variable contains the dividend to price ratio prevailing at the time of the observation. We calculate it in the following way:

$$dp_t = \log(D_t) - \log(p.idx_t) \tag{11}$$

In this equation D is the dividend level at the time and $p.idx$ is the level of the FTSE100 price index. We retrieve the price index directly from Datastream. We calculate the dividend level based

on the dividend yield, retrieved from Datastream, in the following way.

$$D_t = d_t * p.idx \tag{12}$$

Where d is the current dividend to price ratio of the FTSE100 and $p.idx$ is the level of the FTSE100 index. We retrieve both of these from Datastream.

3.2.3 Variable "dy"

The variable dy is the dividend yield an investor would have received if they invested twelve months prior. We use the following approach when calculating it:

$$dy_t = \log(D_t) - \log(p.idx_{t-12}) \tag{13}$$

In this equation D is the dividend level at the time and $p.idx$ is the level of the FTSE100 price index.

3.2.4 Variable "de"

This variable describes the dividend payout ratio, the degree to which earnings are returned to shareholders in the form of dividends. We calculate it in the following way:

$$de_t = \log(D_t) - \log(E_t) \tag{14}$$

In this equation D is the dividend level at the time and E is the earnings level of the FTSE100 index. We calculate the earnings level based on the price-to-earnings ratio, retrieved from Datastream, in the following way.

$$E_t = \frac{1}{pe} * p.idx \tag{15}$$

Where pe is the current price to earnings ratio of the FTSE100 and $p.idx$ is the level of the FTSE100 index. We retrieve both of these from Datastream.

3.2.5 Variable "ep"

The variable ep describes the earnings to price ratio and we calculate it in the following way.

$$ep_t = \log(E_t) - \log(p.idx_t) \tag{16}$$

Where the constituent parts are calculated and retrieved as described above.

3.2.6 Variable "svar"

This variable describes the volatility of the FTSE 100 and is the sum of squared daily returns. The daily returns are calculated by dividing the total return index of one day with that of the previous day. We retrieve the total return index from Datastream.

3.2.7 Variable "tbl"

This variable describes the yield on short-term government bonds. For this we use the one-year rate if that is available, otherwise we use the six-month rate and if even that is not available we use the 18-month rate. We retrieve yields on government bonds from the Bank of England (n.d.).

3.2.8 Variable "tms"

This variable describes the term spread, the difference between long-term and short-term yields on UK government bonds. We calculate it in the following way:

$$tms_t = lty_t - tbl_t \tag{17}$$

For lty we use the 20-year rate if that is available, otherwise we use the 19-year rate and if even that is not available we use the 15-year rate. We retrieve yields on government bonds from the Bank of England (n.d.).

3.2.9 Variable "dfy"

This variable describes the default yield spread, the difference between the yield on triple A rated corporate bonds and bonds rated BAA. We take the following approach when calculating it.

$$dfy_t = BAA_t - AAA_t \tag{18}$$

For BAA we use the yield on the "S&P UK BBB IG CORP BOND INDEX" and for AAA we use the yield on the "S&P UK AAA IG CORP BOND INDEX."[2] We retrieve both from Datastream.

3.2.10 Variable "infl"

This variable describes the latest published, not seasonally adjusted, inflation rate. We calculate this as the one month percentage increase in the retail price index (Office of National Statistics, n.d.).

[2] BBB corresponds to BAA depending on ratings system.

3.2.11 Variable "vrp"

This variable describes the variance risk premium. The variance risk premium is the difference between the volatility implied by option prices and the actual realized volatility. We use the following formula when calculating it.

$$vrp_t = IV_t - RV_t \tag{19}$$

In this equation IV is the end of month value of the FTSE100 Implied Volatility Index squared and then divided by twelve (FTSE-Russell, 2021). Furthermore, the RV is calculated as the monthly sum of squared log five-minute returns added to the sum of squared log close-to-open returns, all this multiplied by 10 000. The input numbers were provided by our excellent tutor Tobias Sichert, Assistant Professor at the Department of Finance, Stockholm School of Economics (Sichert, 2021).

3.2.12 Variable "log.mo.ex.ret"

This is our target variable when we predict monthly returns. We calculate it as follows:

$$log.mo.ex.ret = \log\left(\frac{tot.idx_{t+1}}{tot.idx_t} - Rfree_t\right) \tag{20}$$

Here $tot.idx$ is the total return index for the FTSE100. $Rfree$ is the risk-free rate which is just the variable *tbl* divided by 12.

3.2.13 Variable "log.ye.ex.ret"

This is our target variable when we predict annual returns. We use the following approach when calculating it.

$$log.ye.ex.ret = \log\left(\frac{tot.idx_{t+12}}{tot.idx_t} - tbl_t\right) \tag{21}$$

4 Method

In this section we describe the method used to arrive at our results. We split it into three parts. First, a description of our approach in general. Second, going deeper into the individual algorithms used for prediction. Third, a description of the performance measures we use to evaluate our results.

4.1 General approach

We perform time series prediction of aggregate stock returns in the United States and in the United Kingdom. In making these predictions we use the datasets described in the "data" section of the paper. The returns we predict are the total returns in excess of the risk-free rate for the S&P500 and the FTSE100 indices respectively.

In all, we make 54 384 predictions. Each prediction consists of a choice of dataset, model, return horizon, estimation method and estimation window. As our six different datasets have already been discussed in length, we will omit further description of them from this section. The other choices will be described briefly.

First, we make use of eleven different models to predict returns. These will be discussed further later on in the methods section. The models are (our short names for them within parenthesis); ordinary least squares (OLS), partial least squares (PLS), least absolute shrinkage and selection operator (Lasso), ridge regression (Ridge), elastic net (ElasticNet), principal component regression (PCR), linear support vector machines (SVM), k-nearest neighbors (KNN), random forest (RF), single-layer neural network (NN), three-layer deep neural network (DL1) and historical average (Hist).

Second, we predict returns over two different horizons, monthly returns and annual returns. In both cases, predictions are made monthly, that is, annual returns are also predicted every month.

Third, estimation method and estimation window. We use two different kinds of estimation

methods, rolling window and cumulative window. In the first case, we use a fixed number of months to fit our model. When we have done our prediction and move one month forward, both the starting month and end month of the estimation window increase by one. The length of this window depends on the dataset being used. The lengths in months are; US data (600), US data VRP subsample (180), US data VRP subsample excluding the VRP (180), UK data (120), UK data VRP subsample (120) and UK data VRP subsample excluding the VRP (120).

The second kind of estimation method, cumulative window, is similar to that of the rolling window. The first model fitting is done with a window of equal length to that done with a rolling window. The difference is then that for every month we step forward, the starting month is kept constant whereas the end month is increased by one. Thus, the cumulative window grows as we move forward in time.

We will now move on to describing the fitting process in more detail.

4.2 Models fitted

In this section, we will describe the theoretical foundations of our models as well as our practical implementation of them. Before diving into the specific models though, we want to make some general points.

First, our models can roughly be divided into three groups; benchmark models, penalized linear models and non-linear models. The benchmark models are *Hist* and *OLS*. These are basic models with no machine learning and no tuning. They are included for comparison.

The penalized linear models are; *PLS*, *PCR*, *Lasso*, *Ridge*, *ElasticNet* and *SVM*. Penalized linear models work by adding what is called a regularization or penalty term to the original loss function of the OLS model. This addition is made to counteract overfitting. Gu, Kelly, and Xiu (2020) describe that the problem of overfitting increases with the number of independent variables used, because it starts to fit more noise to the model. The idea behind the penalization term is that it

will decrease the degree to which the regression model is fitted to noise, making it an important addition to evaluate.

The remaining models are the non-linear ones; *RF*, *KNN*, *NN* and *DL1*. The purpose of testing these kinds of models are to see if the ability to pick up more subtle and complex relationships in the data can improve predictions.

Second, for all models except *DL1* and *Hist* we use a process of iterative tuning of our models implemented via the *Caret* package available for R (Kuhn, 2020). This means that for each prediction we make, we refit our model from scratch and tune its hyperparameters. This is done using time series cross validation in which the model is fitted to a continuous window corresponding to 80% of the training dataset and then deployed on the following twelve observations.[3] This is then repeated with a fixed size window as many times as the training data allows. The hyperparameters that minimize root mean square error are then used.

Third, for all models except *Hist*, *OLS*, *PLS* and *RF* we preprocess the data by centering and scaling it.

Fourth, when we deploy our models we of course always do it out-of-sample. The predictions are of course made on the observation following the end of our estimation window.

We will now go over the models we use and describe their theoretical foundations and our practical implementation.

4.2.1 OLS

The most basic model we use to predict index returns is OLS, ordinary least squares. It is a linear regression on the form:

$$\hat{y} = \widehat{\beta_0} + \widehat{\beta_1} * x_1 + ... + \widehat{\beta_k} * x_k \tag{22}$$

[3]Note, this is only in the tuning process. When we make our predictions they are always one month at a turn if we predict monthly returns. They are always a twelve-month return if we predict annual returns.

Where $\widehat{\beta_k}$ is the estimate of β_k and the estimates are obtained by minimizing the sum of squared residuals. That is, minimize the following expression:

$$\sum_{i=1}^{n}(\hat{y_i} - y_i)^2 \tag{23}$$

This is done for n observations of y (Woolridge, 2016, pp. 60–105).

In implementing OLS we use the base R package (R Core Team, 2021). Only one tuning parameter exists for OLS, whether to use an intercept or not. We always use an intercept and thus no tuning takes place.

4.2.2 PLS and PCR

Partial least squares, or PLS, is a dimension reduction method. It uses the independent variables to create a new set of variables that are linear combinations of the original ones. It then fits a linear model via least squares using these variables. PLS is a supervised learning method, which means it takes advantage of knowing the target variable during the dimension reduction (James et al., n.d., pp. 203–259).

Principal component regression, or PCR, reduces overfitting by constructing M principal components, and then using least squares to fit a linear regression model with these components as predictors. The components are derived using principal component analysis, or PCA, which is a dimension reduction technique that seeks the components which makes the observations vary the most. PCR is similar to PLS, but unlike PLS, PCR is an unsupervised method which means it does not take the target variable into consideration when performing dimensionality reduction (James et al., n.d., pp. 203–259, 373–413).

Both PLS and PCR have the same single tuning parameter, *ncomp*, deciding how many predictors to include when making predictions. We tune over all possible values for *ncomp*, from including only one predictor to including all predictors. For both models we use the package "pls"

in R (Mevik, Wehrens, and Liland, 2020).

4.2.3 Lasso, Ridge and ElasticNet

Lasso, ridge and elastic net are three algorithms that closely resemble each other, and we thus discuss them together. Ridge regression penalizes the size of the regression coefficients. This shrinks the coefficients which reduces the degree to which the model is fit to the noise. This is especially true when the coefficients are highly correlated. Specifically, ridge regression models are penalized for the sum of squared residuals (Hastie, Tibshirani, and Friedman, 2017, pp. 61–73).

Lasso, or least absolute shrinkage and selection operator, is similar to ridge. However, it differs in that it penalizes the model for the sum of absolute residuals, rather than sum of squared residuals. It also has the feature that it not only shrinks coefficients, but also sets some coefficients to zero. Hence, it also works as a subset selector (Tibshirani, 1996).

Lasso has the drawback of being somewhat indifferent in the choice between two or more variables that are strongly correlated. Ridge, contrarily, has a tendency to shrink the coefficients of correlated variables such that they converge towards each other (Hastie, Tibshirani, and Friedman, 2017, pp. 661–666). Zou and Hastie (2005) propose a combination of these two methods, the elastic net. It tries to get the best out of both penalization methods without suffering from the same drawbacks. This is done via the α term of the model which can take on a value from zero to one. In the extremes, this would mean that elastic net is just ridge or lasso (Hastie, Tibshirani, and Friedman, 2017, pp. 661–666).

In training these three models we use the Glmnet package available for R (Friedman, 2010). This allows for two tuning parameters, *lambda* and *alpha*. As previously mentioned, *alpha* should be constant for Ridge and Lasso. For these two we only tune over different lambdas; 0, 0.0001, 0.001, 0.01 and 0.1. For ElasticNet we also tune over different alphas; 0.05, 0.1, 0.35, 0.5, 0.65, 0.9, 0.95. In this case, all the interactions between different *alphas* and *lambdas* are tried.

4.2.4 SVM

Support vector machines, or SVM, is a method that was originally used for classification problems. It works by fitting the optimal hyperplane in the case when different classes cannot be separated by a linear boundary. This method can be expanded upon to be applicable for regression. The model then fits a hyperplane like in ordinary SVM and finds an error term ϵ. The hyperplane $\pm\epsilon$ essentially creates a boundary line, within which those values with a sufficiently small error are used for a regression (Hastie, Tibshirani, and Friedman, 2017, pp. 423–438).

In implementing SVM we use the Kernlab package available for R (Karatzoglou et al., 2004). The model only has one tuning parameter, the argument C. This is the cost of constraints violation used in the Lagrange formulation. The default value is 1 and for feasibility reasons this is the only value we use. No tuning thus takes place.

4.2.5 RF

Random forest, or RF, is a method that builds upon the idea of decision trees. There are two types of decision trees, classification trees and regression trees. We use regression trees for our predictions. A regression tree splits the data into branches based on the value of a certain variable, and continues these splits until it reaches the end nodes, or leaves. The number given for each leaf is then the average of the values that end up in that particular leaf. A drawback with decision trees is that they suffer from high variance in their predictions. Bagging is a method of reducing this variance, by averaging a number of observations. However, the gains from bagging are reduced if the bagged trees are highly correlated. Random forest tries to mitigate this issue by "decorrelating" the bagged trees. This is done by restricting each split to only consider a subset of the predictors, which prevents the most important predictor from being highly overrepresented (James et al., n.d., pp. 303–332).

When implementing RF we use the package "randomForest" (Liaw and Wiener, 2002). RF only has one tuning parameter, *mtry* which describes the number of variables to try at each split of the

tree. We try three different values *mtry* and let Caret choose these for us. In addition, we have to manually limit the number of trees to fit each time due to limited computing power. We set the number of trees to 40 which should be enough considering the limited size of our dataset.

4.2.6 KNN

K-nearest-neighbors, or KNN, is based on the idea that the best prediction of y_i is determined by the values for y of the observations closest to i. In mathematical terms, closeness is determined by the least *Euclidean distance* based on the predictor variables. The number K determines how many of the closest values the algorithm should consider when making its prediction. It then averages the y-values for the K nearest neighbors and uses that as its estimate for y_i (Hastie, Tibshirani, and Friedman, 2017, pp. 14–15).

In implementing KNN we use the base R package (R Core Team, 2021). KNN only has one tuning parameter, K. This describes the number of closest neighbors to use for prediction. We try five different values and let Caret choose these for us.

4.2.7 NN

Our neural network model, or NN, is a generic one-layer neural network. Shallow one hidden layer networks can be seen as a special case of deep neural networks, but with just one hidden layer (Poggio et al., 2017). The more hidden layers in a neural network, the deeper the network is. In terms of applying deep learning for asset pricing, the model creates non-linear hidden factors in the hidden layers, and uses these factors, as well as the base variables, to minimize the loss function (Feng, He, and Polson, 2018).

In implementing NN we use the package "nnet" (Venables and Ripley, 2002). This allows for two tuning parameters, *size* and *decay*. *Size* determines the number of neurons in the single layer network and *decay* determines the rate at which the network decays in between. Once again we let

Caret choose three values for each of these and then tune over all their interactions.

We manually have to specify the number of maximum iterations and maximum number of weights. Once again, this is for computational reasons. We set them to 50 and 500 respectively.

4.2.8 DL1

Our deep learning model, or DL1, is the model in which we have applied the largest degree of manual tuning. Taking inspiration from Feng, He, and Polson (2018), we build a three hidden layer deep neural network model with 32 neurons in the first hidden layer, 16 neurons in the second hidden layer, and 8 neurons in the third hidden layer. Like Feng, He and Polson, we use the Stochastic Gradient Descent algorithm (SGD) to minimize the loss function of our variables and factors. However, unlike theirs, our deep learning model is not built as a Long-Short-Term-Memory (LSTM) model.[4]

SGD is the most efficient algorithm for training artificial neural networks, and uses gradients, or vectors of partial derivatives of the target function with respect to the input variables, in order to minimize the loss function. These gradients are calculated via a method called back-propagation, which is an automatic differentiation algorithm that calculates the gradient of a loss function (Brownlee, n.d.). The simplest way to visualize this is to imagine that back-propagation calculates the multidimensional direction to take a step, and SGD decides the length of the step and takes it.

In terms of hyperparameters for the DL1 model, we manually tune the learning rate, epochs, momentum and decay. The learning rate hyperparameter is rather intuitive, it is a measure of how fast a model learns from the data, or in technical terms, how fast the model is adapted to the problem and converges towards a solution. An epoch or training epoch is one run of the model through the entire training dataset. More epochs mean that the model run through and train on the data more times. A higher epoch count typically tends to be paired with a lower learning rate,

[4]LSTM provides a way for a neural network to remove irrelevant information from past time steps and add relevant information from the current time step (Feng, He, and Polson, 2018).

and vice versa (Hastie, Tibshirani, and Friedman, 2017, pp. 389–415).

A common issue with using gradient descent to minimize the error function for a given learning objective is that the minimum can end up in a narrow valley. In those cases, following the gradient direction can cause large oscillations in the search process. One way to counteract this is to add a momentum term. Instead of simply following the current calculated gradient, the use of momentum adds a portion of the previous gradient direction and calculates a weighted average of these two terms. The learning rate is commonly tuned in conjunction with the momentum, and the tuning of these hyperparameters is typically done by trial and error or a random search. The optimal hyperparameters are dependent on the task at hand, and therefore there is no one general approach to handle the tuning of the hyperparameters (Rojas, 1996, pp. 186–187).

Decay, or weight decay, is a method similar to ridge but for neural networks. It aims to reduce overfitting by adding a penalization term. A larger decay value corresponds to a larger penalization (Hastie, Tibshirani, and Friedman, 2017, pp. 389–415).

Our DL model is not fitted using our usual Caret approach. Instead, we use Keras (Allaire and Chollet, 2021) and Tensorflow (Allaire and Tang, 2021). This means that we have to manually choose our hyperparameters. The parameters are set to a *learning rate* of 0.125, a *momentum* of 0, *decay* of 0.1 and *number of epochs* of 125. Even though the hyperparameters are not iteratively tuned, we still retrain the model for each prediction.

4.2.9 Hist

This model simply predicts that the return in a given period will be equal to the historical arithmetic average so far. It is used as a benchmark.

4.3 Performance measures

When measuring the performance of our models we do it in primarily three ways. The root mean square error, out-of-sample R^2 and Sharpe ratio. Below we explain each of these in turn.

4.3.1 Root mean square error

Root mean square error, or RMSE, is a performance metric that describes the out-of-sample prediction errors of the model compared to the observed value. We use the following approach when calculating it (Hyndman and Koehler, 2006):

$$RMSE = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(\widehat{log.ex.ret}_t - log.ex.ret_t)^2} \tag{24}$$

In this equation $log.ex.ret$ is the target variable and T is the number of observations.

The root mean square error, or *RMSE*, can be thought of as the average deviation of predictions from the actual value.

4.3.2 Out-of-sample R^2

This performance metric describes the accuracy of our models compared to a benchmark of historical returns. A value below zero means the model performs worse than the historical average and a value above zero means it is better than the historical average. It can never be larger than one.

We use the following approach when calculating it (Feng, He, and Polson, 2018):

$$OOSR^2 = 1 - \frac{\sum_{t=1}^{T}(log.ex.ret_t - \widehat{log.ex.ret}_t)^2}{\sum_{t=1}^{T}(log.ex.ret_t - \overline{log.ex.ret_t})^2} \tag{25}$$

In this equation $log.ex.ret$ is the target variable and T is the number of observations.

4.3.3 Returns

We use our predictions to form two kinds of portfolios in each setting. One portfolio is a long-only strategy that invests when we predict positive excess returns and otherwise stays out of the market. The other is a long-short strategy that also invests when we predict positive excess returns. However, in the case of this portfolio we go short the excess return whenever we predict it to be negative.

For monthly returns this approach is straight forward. Every month we make predictions and the realized return for that month is dependent on the strategy. For annual returns it is less straightforward however. In this case we still make predictions every month but from the perspective of an investment strategy it would not make sense to predict annual returns but still re-weight the portfolio every month. In this case we re-balance the portfolio once per year with our first investment occasion being the first for which we have a prediction. We then hold the position the strategy recommended for twelve months before rebalancing.

For the purpose of calculating Sharpe ratios, we use log returns. When computing cumulative returns we use non-log returns.

4.3.4 Sharpe ratio

Sharpe ratio is a measure of risk-adjusted excess returns. The purpose is to offer a way of penalizing portfolios that achieve high returns but that are risky. We use the following simple approach when calculating it (Lo, 2002):

$$SR = \frac{\overline{re_p}}{\sigma_p} \tag{26}$$

In this equation $\overline{re_p}$ is the arithmetic average of the log excess return of the portfolio and σ_p is the volatility of the excess return of the portfolio.

This equation requires no adjustment when looking at annual returns but for monthly returns

the numbers have to be annualized for a comparison to be possible. It is convention to quote Sharpe ratios in annual terms rather than monthly. We use a simple way of annualizing returns per the below:

$$SR = \sqrt{12} * \frac{\overline{re_p}}{\sigma_p} \tag{27}$$

This approach to annualization has been criticized by for instance Lo (2002). However, it is still a widely used standard and even Lo points out that if the returns are not too high and roughly independent and identically distributed, the standard approach works reasonably well. Lo further argues that when the arithmetic average is used for calculating the average return, then log returns should be used.

4.3.5 Diebold-Mariano test statistic

The Diebold-Mariano test statistic is a method used to evaluate whether the difference between the prediction errors of two forecasts are statistically significant. The definition of this test involves some tedious algebra outside the scope of this paper. We however want to briefly state that we follow the definition laid out in the article in which the test was originally proposed (Diebold and Mariano, 1995) and that our implementation uses the "forecast" package in R (Hyndman and Khandakar, 2008).

5 Main results

In this thesis we implement time series prediction for the S&P500 and FTSE100 indices and in this section we present our findings. It is divided into four subsections. First, we present the overall results. Second, we present some differences between results for the US and the UK and their implications. Third, we present our results with respect to the addition of the variance risk premium as a predictor. Fourth, we test the statistical significance of some of our predictions.

It is worth noting that the goal of this paper is prediction, not inference. We will thus not go into depth concerning which variables give our models predictive power. Rather, we will focus on the overall results as to whether prediction is possible at all.

For a more detailed and comprehensive overview of our results, please refer to the appendix.

5.1 Overall results

In this paper we use monthly data from the US and the UK to predict aggregate stock returns on both a monthly and an annual basis. For the US we try to predict the return on the S&P500 index and for the UK we use the FTSE100 index. Using combinations of six different datasets, two return horizons, two training methods and ten statistical algorithms, we make a total of 54 384 predictions. It is these predictions that our results are based on.

Our results confirm that aggregate stock returns are hard to predict. Out of a total of 132 attempts to predict monthly returns only 13 have a positive out-of-sample R^2, meaning that the predictions perform better than a benchmark consisting of the historical average.[5] Furthermore, only 64 of the long-only portfolios formed have a Sharpe ratio higher than a benchmark of a simple buy-and-hold portfolio. The corresponding number when it comes to long-short portfolios is 36.

[5] In this context, an *attempt* refers to a combination of model, dataset, training method and return horizon. It does not refer to individual predictions.

Moreover, our results are in line with conventional wisdom in the sense that stock returns are more predictable over longer horizons. Out of 132 attempts at predicting annual returns, fully 78 have an out-of-sample R^2 above zero. In addition, 87 of the long-only portfolios formed have a Sharpe ratio higher than a benchmark of a simple buy-and-hold portfolio. The corresponding number when it comes to long-short portfolios is 81.

For comparison we provide the following table in which buying and holding the market portfolio is provided as a benchmark. More complete results can be found in the appendix.

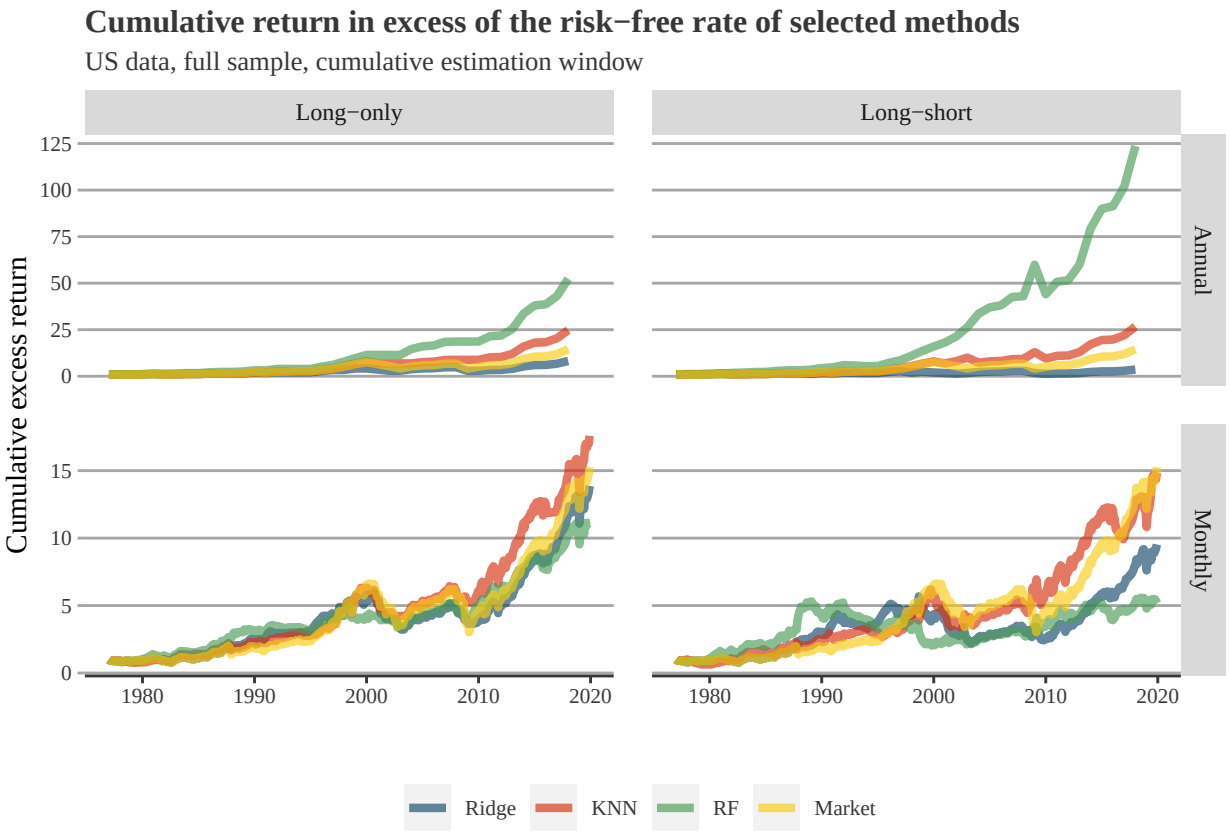
Table 1: Sharpe ratio of models, US full sample, rolling estimation

Measure	Long-only portfolio	Long-short portfolio
1. Annual		
Median	0.401	0.418
P25	0.346	0.214
P75	0.602	0.563
Market	0.379	0.379
2. Monthly		
Median	0.411	0.335
P25	0.401	0.187
P75	0.441	0.361
Market	0.430	0.430

[a] Table is based on the full US sample with rolling estimation. The results are based on eleven models and 'Median', 'P25' and 'P75' indicates the performance of the median model, the model in the 25 percentile and the model in the 75th percentile respectively.
[b] The long-only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long-short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re-evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk-free rate.
[1] Row panel '1' is based on predicting annual returns.
[2] Row panel '2' is based on predicting monthly returns.

That stock returns are hard to predict does of course not mean that it is impossible to find examples of models that perform well. For instance, using US data, annual return prediction and

cumulative estimation, our random forest performs well.



The models with the best average ranking over root mean square error, out−of−sample R−squared, long−only Sharpe ratio and long−short Sharpe ratio are displayed. The base value for all strategies is set to one and develops with the excess return that results from the strategy. The long−only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long−short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re−evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk−free rate.

Figure 1

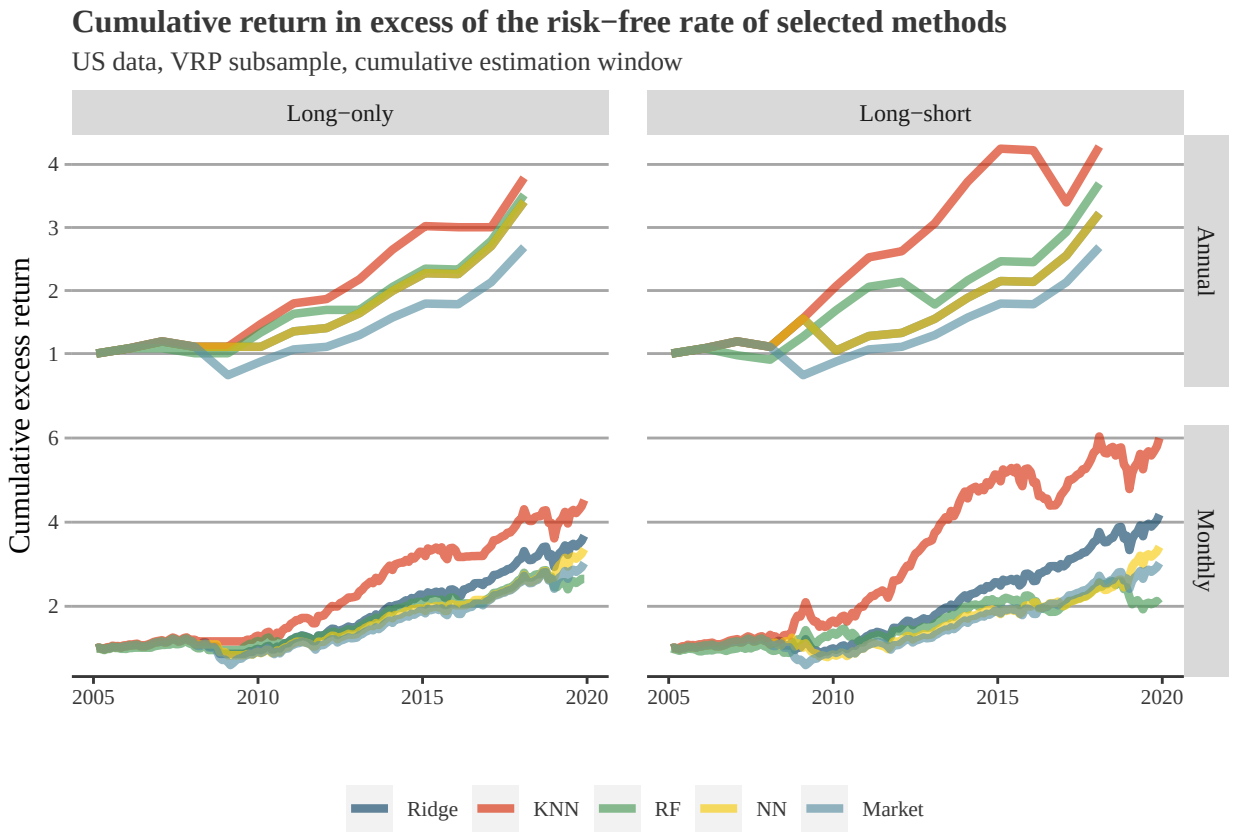
However, there is always the risk of these outperformers being a consequence of "data mining." In the course of writing this thesis we have noticed that some models are sensitive to minor adjustments. One example is our neural network. It is the top performer when it comes to predicting monthly returns with our dataset "US data, VRP subsample" if we use a rolling estimation window. However, it barely beats the market if we use a cumulative estimation window. This is also contrary to intuition, if anything it should be the other way around since more data is usually seen as warranting better predictions.

Cumulative return in excess of the risk−free rate of selected methods

US data, VRP subsample, rolling estimation window

The models with the best average ranking over root mean square error, out−of−sample R−squared, long−only Sharpe ratio and long−short Sharpe ratio are displayed. The base value for all strategies is set to one and develops with the excess return that results from the strategy. The long−only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long−short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re−evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk−free rate.

Figure 2



The models with the best average ranking over root mean square error, out−of−sample R−squared, long−only Sharpe ratio and long−short Sharpe ratio are displayed. The base value for all strategies is set to one and develops with the excess return that results from the strategy. The long−only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long−short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re−evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk−free rate.

Figure 3

There are several possible explanations for this result. In the particular case of Neural Networks, a contributing factor is that these kinds of models feature a lot of randomness. Small adjustments in the initial assumptions of the model, coupled with the inherit randomness of its components, give rise to volatile results. The performance of this model should thus be viewed with caution.

Furthermore, even disregarding the randomness of some models, there is always the risk of spurious results. In this paper we make use of many models over many different settings and it would be surprising if we did not find some that outperformed more naive and simple methods.

However, even considering this, some generalizations seem to be possible. Overall, Ridge seems to be our best model with regard to predicting monthly returns.[6] Over 132 different set ups it has an average root mean square error of 0.04. Furthermore it has an average out-of-sample R^2 of -0.042. With a long-only investment strategy it has an average Sharpe ratio of 0.59. For a long-short investment strategy, the corresponding number is 0.55. For comparison, the median numbers over all set ups and models are 0.043, -0.099, 0.53 and 0.42.

When looking at the results for the annual return predictions, the top performers change. There, the best model is RF. Over 132 different set ups it has an average root mean square error of 0.094. Furthermore it has an average out-of-sample R^2 of 0.59. With a long-only investment strategy it has an average Sharpe ratio of 0.79. For a long-short investment strategy, the corresponding number is also 0.79. For comparison, the median numbers over all set ups and models are 0.14, 0.073, 0.78 and 0.59.

[6]There is no obvious way to determine which model is the "best." In this case we have chosen the algorithm that has the best average ranking across the metrics root mean square error, out-of-sample R^2, long-only Sharpe ratio and long-short Sharpe ratio, over all different set ups.

Table 2: Average performance of top three models and benchmarks

RANK	MODEL	RMSE	OOSR2	SR_LO	SR_LS
1. Monthly					
1	Ridge	0.040	-0.042	0.588	0.554
2	ElasticNet	0.040	-0.037	0.489	0.390
3	Lasso	0.039	-0.019	0.457	0.291
NA	Median model	0.043	-0.099	0.526	0.423
NA	Hist	0.039	0.000	0.506	0.367
NA	Market	NA	NA	0.520	0.520
2. Annual					
1	RF	0.094	0.593	0.788	0.794
2	KNN	0.108	0.467	0.782	0.803
3	NN	0.113	0.390	0.775	0.735
NA	Median model	0.136	0.073	0.780	0.592
NA	Hist	0.147	0.000	0.427	0.270
NA	Market	NA	NA	0.469	0.469

[1] Row panel '1' is based on predicting monthly returns.
[2] Row panel '2' is based on predicting annual returns.
[*] The long-only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long-short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re-evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk-free rate.
[†] RMSE = root mean square error ; OOSR2 = out-of-sample R-squared ; SR_LO = Sharpe ratio of long only portfolio ; SR_LS Sharpe ratio of long-short portfolio

The results in the table above provides another interesting insight. It appears to be the case that linear models perform better than non-linear models when predicting monthly returns, and vice versa for annual returns. It is not clear why this is the case, but further investigation may be warranted.

5.2 US data versus UK data

When considering whether any results are spurious, it is interesting to examine whether results are replicable in another country. If that is the case, there is some support for the notion that returns are actually predictable. If they are not, it would point towards the results being a consequence of happenstance.

For instance, in our case, an interesting result is that about as many attempts are successful in terms of having positive out-of-sample R^2 on an annual basis for both the UK and the US, 40 out of 66 in the US and 38 out of 66 in the UK.

However, this does not mean that all is well. It is worrying that the performance of different models varies quite a lot between the two countries. However, this may be a consequence of different sample periods and some differences in available variables. Nonetheless, further investigation is warranted.

Table 3: Average performance of different models by country

MODEL	RMSE_UK	RMSE_US	OOSR2_UK	OOSR2_US	SR_LO_UK	SR_LO_US	SR_LS_UK	SR_LS_US
1. Annual								
OLS	0.139	0.162	-0.211	0.066	0.735	0.579	0.775	0.291
PLS	0.126	0.169	0.012	-0.034	0.602	0.694	0.612	0.494
Lasso	0.135	0.167	-0.140	-0.006	0.525	0.473	0.214	0.354
Ridge	0.115	0.157	0.174	0.116	0.723	0.726	0.751	0.548
ElasticNet	0.124	0.162	0.045	0.054	0.700	0.511	0.610	0.415
PCR	0.120	0.155	0.094	0.135	0.725	0.739	0.755	0.582
SVM	0.130	0.175	-0.060	-0.095	0.735	0.667	0.775	0.483
KNN	0.087	0.129	0.518	0.416	0.719	0.845	0.829	0.776
RF	0.082	0.106	0.579	0.607	0.671	0.905	0.751	0.837
NN	0.100	0.126	0.367	0.413	0.703	0.848	0.732	0.738
DL1	0.190	0.262	-1.387	-1.439	0.612	0.404	0.351	0.132
Hist	0.126	0.169	0.000	0.000	0.500	0.353	0.187	0.353
Market	0.125	0.181	0.018	-0.145	0.586	0.353	0.586	0.353
2. Monthly								
OLS	0.041	0.044	-0.226	-0.131	0.574	0.625	0.560	0.530
PLS	0.039	0.043	-0.103	-0.099	0.435	0.565	0.248	0.408
Lasso	0.038	0.041	-0.046	0.007	0.408	0.506	0.105	0.478
Ridge	0.038	0.041	-0.084	-0.001	0.535	0.642	0.494	0.615
ElasticNet	0.038	0.041	-0.076	0.002	0.424	0.554	0.247	0.533
PCR	0.039	0.043	-0.113	-0.089	0.482	0.567	0.392	0.437
SVM	0.041	0.046	-0.255	-0.259	0.618	0.653	0.640	0.576
KNN	0.038	0.042	-0.081	-0.042	0.392	0.666	0.182	0.525
RF	0.039	0.043	-0.098	-0.114	0.505	0.535	0.358	0.321
NN	0.039	0.045	-0.119	-0.169	0.544	0.637	0.505	0.546
DL1	0.214	0.155	-35.220	-14.932	0.429	0.289	-0.049	-0.135
Hist	0.037	0.041	0.000	0.000	0.505	0.508	0.226	0.508
Market	0.037	0.042	-0.001	-0.016	0.532	0.508	0.532	0.508

[1] Row panel '1' is based on predicting annual returns.
[2] Row panel '2' is based on predicting monthly returns.
[*] The long-only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long-short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re-evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk-free rate.
[†] RMSE = root mean square error ; OOSR2 = out-of-sample R-squared ; SR_LO = Sharpe ratio of long-only portfolio ; SR_LS Sharpe ratio of long-short portfolio

5.3 Addition of the variance risk premium

Even though inference is not the primary purpose of our paper, one variable in particular is of some interest. That is the variance risk premium. Remember, the variance risk premium describes the difference between the volatility implied by option prices and the actual realized volatility.

Our thinking goes that this should improve predictability, especially in the short run. This because patterns of clustered volatility have historically been observed in conjunction with market crashes. The notion that stock market ups tend to be long and flat whereas market downs tend to be short and steep.

We find scant evidence that the variance risk premium increases overall performance. Out of 88 attempts, the out-of-sample R^2 increased in only 47 cases. However, out of these cases, fully 27 was when we predicted monthly returns.

5.4 Robustness of results

Some of our models achieve a superior performance compared to that of benchmark methods such as the historical average. In order to get a sense of whether these results are spurious we perform a robustness check using the Diebold-Mariano test statistic. We only perform this test for predictions that use the full US sample.

Table 4: Performance and robustness with US data, full sample

	A. Monthly			*B. Annual*		
MODEL	A.RMSE	A.OOSR2	A.DM	B.RMSE	B.OOSR2	B.DM
1. Rolling						
OLS	0.044	-0.055	-1.42	0.168	-0.083	-1.45
PLS	0.044	-0.065	-2.11**	0.174	-0.156	-2.69***
Lasso	0.043	0.001	0.12	0.160	0.019	1.64
Ridge	0.043	-0.015	-0.81	0.157	0.052	1.3
ElasticNet	0.043	-0.003	-0.34	0.157	0.052	3.17***
PCR	0.044	-0.026	-1.08	0.161	0.010	0.25
SVM	0.045	-0.079	-2.4**	0.186	-0.330	-4.24***
KNN	0.044	-0.053	-1.96*	0.124	0.409	5.97***
RF	0.045	-0.107	-3.11***	0.104	0.587	8.24***
NN	0.045	-0.084	-2.12**	0.134	0.316	3.98***
DL1	0.088	-3.227	-6.35***	0.237	-1.147	-9.97***
2. Cumulative						
OLS	0.045	-0.078	-1.78*	0.179	-0.230	-3.98***
PLS	0.044	-0.058	-1.49	0.187	-0.343	-5.53***
Lasso	0.043	-0.011	-1.66*	0.166	-0.056	-3.37***
Ridge	0.043	0.003	0.24	0.169	-0.089	-2.36**
ElasticNet	0.043	-0.005	-0.38	0.165	-0.047	-2.53**
PCR	0.043	-0.020	-0.74	0.179	-0.222	-3.93***
SVM	0.044	-0.032	-0.88	0.171	-0.125	-2.64***
KNN	0.044	-0.042	-1.35	0.133	0.326	4.69***
RF	0.045	-0.116	-2.24**	0.107	0.564	7.96***
NN	0.052	-0.480	-1.5	0.171	-0.125	-1.43
DL1	0.131	-8.293	-13.98***	0.293	-2.278	-9.6***

[a] Column panel 'A' is based on one month returns predicted every month.
[b] Column panel 'B' is based on twelve-month returns predicted every month.
[1] Row panel '1' is based on models refitted with tuning every month using a rolling estimation window of 600 months.
[2] Row panel '2' is based on models refitted with tuning every month using a cumulative estimation window of minimum 600 months.
[*] Robustness is checked via the Diebold-Mariano test statistic (DM). It tests whether the prediction errors of a model are sginficantly different from those generated when using the historical average. Significane levels are; * (10%), ** (5%) and *** (1%).
[†] RMSE = root mean square error ; OOSR2 = out-of-sample R-squared

It becomes clear that what positive results we have on a monthly basis are probably a consequence of pure chance. No model with a performance better than that of the historical average shows significance.

When it comes to annual predictions, most models show significance. Crucially this is true for most of the models that outperform the historical average. This indicates that our positive results with regard to annual prediction are robust.

6 Conclusion

In this paper we examine whether total returns in excess of the risk-free rate, on the S&P500 and FTSE100 indices, are predictable using time series prediction. We have virtually no success in predicting monthly returns. However, we do have some success in predicting annual returns.

Fully 78 out of our 132 attempts at annual return prediction have a positive out-of-sample R^2. Furthermore, we find that these this translates to economic gains. We find that in 87 cases, long-only portfolios formed based on our models have a Sharpe ratio higher than a simple buy-and-hold portfolio.

We achieve our best result when using US data from 1926 to 2019, to predict annual returns using the random forest algorithm and a cumulative estimation window. We find that if we use this model to form a long-short portfolio we have a Sharpe ratio of 0.996. This set up also has an out-of-sample R^2 of 0.564. We use a Diebold-Mariano test to conclude that this performance, relative to benchmark predictions equal to the historical mean, is significant on the 1% level.

Overall, our results are somewhat in line with previous research. We find that stock returns are hard to predict, easier to predict over longer horizons than short ones and that machine learning methods can improve upon simpler methods for prediction.

We think it would be beneficial to pursue further research into the methods of deep learning. Our review of previous research shows that these models often are the best performing ones. Yet, that is not the case in our study. This discrepancy is most likely a result of insufficient tuning of hyperparameters.

We would find it interesting to see how a successful deep learning algorithm, along the lines of for instance Feng, He, and Polson (2018), would perform if used to implement our trading strategies. It would be interesting to see if it could outperform the market in terms of Sharpe ratio when it comes to monthly predictions, and whether it could improve upon already good Sharpe ratios of other machine learning methods when it comes to annual prediction.

7 References

Allaire, J., and Chollet, F. (2021). *Keras: R interface to 'keras'*. https://CRAN.R-project.org/package=keras

Allaire, J., and Tang, Y. (2021). *Tensorflow: R interface to 'TensorFlow'*. https://CRAN.R-project.org/package=tensorflow

Bank of England. (n.d.). *Yield curves*. https://www.bankofengland.co.uk/statistics/yield-curves

Brownlee, J. (n.d.). *Difference between backpropagation and stochastic gradient descent*. https://machinelearningmastery.com/difference-between-backpropagation-and-stochastic-gradient-descent/

Campbell, J. Y., and Cochrane, J. H. (1999). By force of habit: A consumption-based explanation of aggregate stock market behavior. *Journal of Political Economy*, *107*(2), 205–251. https://www-jstor-org.ez.hhs.se/stable/10.1086/250059

Cochrane, J. (2008). The dog that did not bark: A defense of return predictability. *The Review of Financial Studies*, *21*(4), 1533–1575. https://www-jstor-org.ez.hhs.se/stable/40056861

Diebold, F. X., and Mariano, R. S. (1995). Comparing predictive accuracy. *Journal of Business & Economics Statistics*, *13*(3), 253–263. https://www.jstor.org/stable/1392185

Dow, C. H., and Selden, G. C. (1920). *Scientific stock speculation*. The Magazine of Wall Street. https://archive.org/details/scientificstocks00dowc/

Fama, E. (1970). Efficient capital markets: A review of theory and empirical work. *The Journal of Finance*, *25*(2), 383–417. https://www.jstor.org/stable/2325486

Feng, G., He, J., and Polson, N. (2018). *Deep learning for predicting asset returns*. https://arxiv.org/pdf/1804.09314.pdf

Freyberger, J., Neuhierl, A., and Weber, M. (2020). Dissecting characteristics nonparametrically. *The Review of Financial Studies*, *33*(5), 2326–2377. https://doi-org.ez.hhs.se/10.1093/rfs/hhz123

Friedman, T. T., Jerome; Hastie. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, *33*(1), 1–22. https://www.jstatsoft.org/v33/i01/

FTSE-Russell. (2021). *Mail correspondence*.

Goyal, A. (n.d.). *Amit goyal - university of lausanne*. http://www.hec.unil.ch/agoyal/docs/PredictorData2019.xlsx

Gu, S., Kelly, B., and Xiu, D. (2020). Empirical asset pricing via machine learning. *The Review of Financial Studies*, *33*(5), 2223–2273. https://doi-org.ez.hhs.se/10.1093/rfs/hhaa009

Hastie, T., Tibshirani, R., and Friedman, J. (2017). *The elements of statistical learning: Data mining, inference and prediction*. Springer. https://web.stanford.edu/~hastie/ElemStatLearn/printings/ESLII_print12_toc.pdf

Hyndman, R. J., and Khandakar, Y. (2008). Automatic time series forecasting: The forecast package for R. *Journal of Statistical Software*, *26*(3), 1–22. https://www.jstatsoft.org/article/view/v027i03

Hyndman, R. J., and Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, *22*(4), 679–688. https://www-sciencedirect-com.ez.hhs.se/science/article/pii/S0169207006000239

Ibbotson, R. G., and Harrington, J. P. (2020). *Stocks, bonds, bills and inflation (SBBI)*. Duff & Phelps. https://www.cfainstitute.org/-/media/documents/book/rf-publication/2020/rf-sbbi-summary-edition.ashx

James, G., Witten, D., Tibshirani, R., and Hastie, T. (n.d.). *An introduction to statistical learning: With applications in r*. Springer. https://static1.squarespace.com/static/5ff2adbe3fe4fe33db902812/t/6062a083acbfe82c7195b27d/1617076404560/ISLR%2BSeventh%2BPrinting.pdf

Karatzoglou, A., Smola, A., Hornik, K., and Zeileis, A. (2004). Kernlab – an S4 package for kernel methods in R. *Journal of Statistical Software*, *11*(9), 1–20. http://www.jstatsoft.org/v11/i09/

Kuhn, M. (2020). *Caret: Classification and regression training*. https://CRAN.R-project.org/package=caret

Liaw, A., and Wiener, M. (2002). Classification and regression by randomForest. *R News*, *2*(3), 18–22. https://CRAN.R-project.org/doc/Rnews/

Lo, A. W. (2002). The statistics of sharpe ratios. *Financial Analysts Journal*, *58*(4), 36–52. https://www-jstor-org.ez.hhs.se/stable/4480405

Mevik, B.-H., Wehrens, R., and Liland, K. H. (2020). *Pls: Partial least squares and principal component regression*. https://CRAN.R-project.org/package=pls

Office of National Statistics. (n.d.). *Retail price index: Long run series*. https://www.ons.gov.uk/economy/inflationandpriceindices/timeseries/cdko/mm23

Poggio, T., Mhaskar, H., Rosasco, L., Miranda, B., and Liao, Q. (2017). Why and when can deep-but not shallow-networks avoid the curse of dimensionality: A review. *International Journal of Automation and Computing*, *14*, 503–519. https://link-springer-com.ez.hhs.se/article/10.1007/

s11633-017-1054-2

R Core Team. (2021). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. https://www.R-project.org/

Rojas, R. (1996). *Neural networks: A systematic introduction*. Springer. https://page.mi.fu-berlin.de/rojas/neural/neuron.pdf

Shiller, R. (1981). Do stock prices move too much to be justified by subsequent changes in dividends? *The American Economic Review*, *71*(3), 421–436. https://www-jstor-org.ez.hhs.se/stable/1802789

Shiller, R. (2013). Speculative asset prices. *Nobel Prize Lectures*. https://www.nobelprize.org/uploads/2018/06/shiller-lecture.pdf

Sichert, T. (2021). *Mail correspondence*.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B*, *58*(1), 267–288. https://www.jstor.org/stable/2346178

Venables, W. N., and Ripley, B. D. (2002). *Modern applied statistics with s* (Fourth). Springer. https://www.stats.ox.ac.uk/pub/MASS4/

Welch, I., and Goyal, A. (2008). A comprehensive look at the empirical performance of equity premium prediction. *The Review of Financial Studies*, *21*(4), 1455–1508. https://doi-org.ez.hhs.se/10.1093/rfs/hhm014

Woolridge, J. M. (2016). *Introductory econometrics: A modern approach*. Cengage Learning.

Zhou, H. (n.d.). *VRPtable.txt*. https://drive.google.com/open?id=0Bwai7ZTn1nJSbThUcHp2OTFoUWs

Zou, H., and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society. Series B*, *67*(2), 301–320. http://ez.hhs.se/login?url=http://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=buh&AN=16342068&site=ehost-live

8 Appendix

This appendix contains a more detailed and comprehensive version of our results, it has three subsections. First, the accuracy of our predictions. Second, the accuracy of the *direction* of our predictions. Third, plots illustrating the cumulative excess returns of different models.

8.1 Appendix A

This section contains tables describing the accuracy of our predictions.

Table 5: Performance with UK data, full sample

	A. Monthly				*B. Annual*			
MODEL	A.RMSE	A.OOSR2	A.SR_LO	A.SR_LS	B.RMSE	B.OOSR2	B.SR_LO	B.SR_LS
1. Rolling								
OLS	0.045	-0.217	0.537	0.581	0.162	-0.160	0.608	0.694
PLS	0.045	-0.189	0.476	0.464	0.150	0.009	0.195	0.191
Lasso	0.043	-0.098	0.328	0.106	0.165	-0.208	0.540	0.270
Ridge	0.044	-0.145	0.443	0.405	0.140	0.134	0.608	0.694
ElasticNet	0.044	-0.138	0.355	0.191	0.157	-0.093	0.540	0.270
PCR	0.044	-0.162	0.424	0.364	0.151	-0.005	0.614	0.705
SVM	0.048	-0.356	0.516	0.592	0.161	-0.145	0.608	0.694
KNN	0.043	-0.080	0.328	0.229	0.104	0.519	0.243	0.285
RF	0.043	-0.069	0.354	0.217	0.093	0.616	0.195	0.191
NN	0.046	-0.251	0.327	0.179	0.122	0.339	0.191	0.184
DL1	0.361	-75.896	0.310	-0.188	0.172	-0.300	0.530	0.489
Hist	0.041	0.000	0.499	0.304	0.151	0.000	0.426	0.147
Market	NA	NA	0.375	0.375	NA	NA	0.198	0.198
2. Cumulative								
OLS	0.045	-0.181	0.628	0.757	0.180	-0.439	0.608	0.694
PLS	0.044	-0.123	0.388	0.312	0.173	-0.327	0.111	0.015
Lasso	0.043	-0.081	0.269	-0.042	0.179	-0.429	0.313	0.045
Ridge	0.043	-0.089	0.501	0.515	0.153	-0.047	0.608	0.694
ElasticNet	0.043	-0.072	0.165	-0.137	0.171	-0.304	0.540	0.270
PCR	0.043	-0.109	0.252	0.107	0.161	-0.151	0.614	0.705
SVM	0.049	-0.401	0.312	0.185	0.172	-0.315	0.608	0.694
KNN	0.043	-0.073	0.255	0.106	0.106	0.502	0.198	0.198
RF	0.044	-0.115	0.422	0.249	0.107	0.489	0.240	0.278
NN	0.044	-0.128	0.527	0.594	0.130	0.250	0.527	0.536
DL1	0.201	-22.814	0.052	-0.317	0.353	-4.525	0.596	0.247
Hist	0.041	0.000	0.460	0.134	0.150	0.000	0.313	0.045
Market	NA	NA	0.375	0.375	NA	NA	0.198	0.198

[a] Column panel 'A' is based on one month returns predicted every month.

[b] Column panel 'B' is based on twelve-month returns predicted every month.

[1] Row panel '1' is based on models refitted with tuning every month using a rolling estimation window of 120 months.

[2] Row panel '2' is based on models refitted with tuning every month using a cumulative estimation window of minimum 120 months.

[*] The long-only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long-short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re-evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk-free rate.

[†] RMSE = root mean square error ; OOSR2 = out-of-sample R-squared ; SR_LO = Sharpe ratio of long-only portfolio ; SR_LS Sharpe ratio of long-short portfolio

Table 6: Performance with UK data, VRP subsample

	A. Monthly				*B. Annual*			
MODEL	A.RMSE	A.OOSR2	A.SR_LO	A.SR_LS	B.RMSE	B.OOSR2	B.SR_LO	B.SR_LS
1. Rolling								
OLS	0.039	-0.273	0.542	0.436	0.111	0.011	0.798	0.816
PLS	0.035	-0.064	0.452	0.161	0.096	0.262	0.780	0.780
Lasso	0.036	-0.072	0.442	0.190	0.113	-0.022	0.644	0.381
Ridge	0.036	-0.071	0.549	0.475	0.095	0.289	0.780	0.780
ElasticNet	0.036	-0.092	0.533	0.430	0.101	0.187	0.780	0.780
PCR	0.037	-0.175	0.502	0.356	0.092	0.331	0.780	0.780
SVM	0.037	-0.159	0.870	0.984	0.100	0.198	0.798	0.816
KNN	0.036	-0.075	0.460	0.237	0.079	0.500	0.969	1.123
RF	0.037	-0.150	0.453	0.213	0.072	0.591	0.798	0.816
NN	0.036	-0.112	0.616	0.593	0.074	0.564	0.932	1.073
DL1	0.141	-15.893	0.888	0.179	0.142	-0.615	0.373	-0.092
Hist	0.034	0.000	0.545	0.360	0.112	0.000	0.628	0.361
Market	NA	NA	0.610	0.610	NA	NA	0.780	0.780
2. Cumulative								
OLS	0.039	-0.237	0.585	0.555	0.133	-0.324	0.798	0.816
PLS	0.037	-0.124	0.230	-0.203	0.122	-0.118	0.969	1.123
Lasso	0.035	-0.002	0.450	0.040	0.120	-0.078	0.503	0.104
Ridge	0.036	-0.066	0.588	0.558	0.104	0.195	0.780	0.780
ElasticNet	0.035	-0.033	0.482	0.294	0.105	0.165	0.780	0.780
PCR	0.036	-0.072	0.589	0.565	0.113	0.050	0.780	0.780
SVM	0.038	-0.197	0.613	0.611	0.123	-0.138	0.798	0.816
KNN	0.035	-0.043	0.457	0.201	0.080	0.523	0.969	1.123
RF	0.035	-0.038	0.647	0.576	0.071	0.620	0.932	1.073
NN	0.036	-0.082	0.529	0.423	0.109	0.103	0.994	1.185
DL1	0.262	-56.103	0.317	-0.286	0.135	-0.359	0.958	0.861
Hist	0.035	0.000	0.491	0.101	0.115	0.000	0.503	0.104
Market	NA	NA	0.610	0.610	NA	NA	0.780	0.780

[a] Column panel 'A' is based on one month returns predicted every month.

[b] Column panel 'B' is based on twelve-month returns predicted every month.

[1] Row panel '1' is based on models refitted with tuning every month using a rolling estimation window of 120 months.

[2] Row panel '2' is based on models refitted with tuning every month using a cumulative estimation window of minimum 120 months.

[*] The long-only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long-short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re-evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk-free rate.

[†] RMSE = root mean square error ; OOSR2 = out-of-sample R-squared ; SR_LO = Sharpe ratio of long-only portfolio ; SR_LS Sharpe ratio of long-short portfolio

Table 7: Performance with UK data, VRP subsample excluding the VRP

	A. Monthly				*B. Annual*			
MODEL	A.RMSE	A.OOSR2	A.SR_LO	A.SR_LS	B.RMSE	B.OOSR2	B.SR_LO	B.SR_LS
1. Rolling								
OLS	0.038	-0.238	0.564	0.480	0.111	0.015	0.798	0.816
PLS	0.036	-0.079	0.545	0.385	0.094	0.302	0.780	0.780
Lasso	0.035	-0.031	0.512	0.298	0.113	-0.022	0.644	0.381
Ridge	0.036	-0.073	0.564	0.504	0.095	0.287	0.780	0.780
ElasticNet	0.036	-0.100	0.533	0.430	0.103	0.156	0.780	0.780
PCR	0.036	-0.115	0.541	0.409	0.092	0.322	0.780	0.780
SVM	0.038	-0.215	0.768	0.815	0.101	0.186	0.798	0.816
KNN	0.037	-0.131	0.441	0.200	0.077	0.523	0.969	1.123
RF	0.036	-0.117	0.548	0.411	0.074	0.568	0.932	1.073
NN	0.036	-0.113	0.613	0.574	0.082	0.466	0.932	1.073
DL1	0.149	-17.713	0.563	0.493	0.176	-1.465	0.710	0.496
Hist	0.034	0.000	0.545	0.360	0.112	0.000	0.628	0.361
Market	NA	NA	0.610	0.610	NA	NA	0.780	0.780
2. Cumulative								
OLS	0.038	-0.212	0.585	0.555	0.135	-0.370	0.798	0.816
PLS	0.035	-0.042	0.517	0.371	0.119	-0.055	0.780	0.780
Lasso	0.035	0.006	0.450	0.040	0.120	-0.084	0.503	0.104
Ridge	0.036	-0.060	0.563	0.504	0.104	0.184	0.780	0.780
ElasticNet	0.035	-0.021	0.474	0.274	0.106	0.159	0.780	0.780
PCR	0.035	-0.043	0.586	0.554	0.115	0.016	0.780	0.780
SVM	0.038	-0.203	0.631	0.649	0.124	-0.146	0.798	0.816
KNN	0.036	-0.085	0.410	0.117	0.078	0.543	0.969	1.123
RF	0.036	-0.101	0.604	0.485	0.074	0.586	0.932	1.073
NN	0.035	-0.025	0.648	0.670	0.083	0.478	0.644	0.338
DL1	0.170	-22.903	0.442	-0.176	0.166	-1.057	0.503	0.104
Hist	0.035	0.000	0.491	0.101	0.115	0.000	0.503	0.104
Market	NA	NA	0.610	0.610	NA	NA	0.780	0.780

[a] Column panel 'A' is based on one month returns predicted every month.

[b] Column panel 'B' is based on twelve-month returns predicted every month.

[1] Row panel '1' is based on models refitted with tuning every month using a rolling estimation window of 120 months.

[2] Row panel '2' is based on models refitted with tuning every month using a cumulative estimation window of minimum 120 months.

[*] The long-only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long-short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re-evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk-free rate.

[†] RMSE = root mean square error ; OOSR2 = out-of-sample R-squared ; SR_LO = Sharpe ratio of long-only portfolio ; SR_LS Sharpe ratio of long-short portfolio

Table 8: Performance with US data, full sample

	A. Monthly				*B. Annual*			
MODEL	A.RMSE	A.OOSR2	A.SR_LO	A.SR_LS	B.RMSE	B.OOSR2	B.SR_LO	B.SR_LS
1. Rolling								
OLS	0.044	-0.055	0.411	0.176	0.168	-0.083	0.329	0.211
PLS	0.044	-0.065	0.368	0.107	0.174	-0.156	0.354	0.217
Lasso	0.043	0.001	0.426	0.374	0.160	0.019	0.408	0.432
Ridge	0.043	-0.015	0.410	0.335	0.157	0.052	0.398	0.359
ElasticNet	0.043	-0.003	0.456	0.429	0.157	0.052	0.401	0.418
PCR	0.044	-0.026	0.413	0.342	0.161	0.010	0.550	0.627
SVM	0.045	-0.079	0.465	0.348	0.186	-0.330	0.298	0.152
KNN	0.044	-0.053	0.515	0.423	0.124	0.409	0.677	0.587
RF	0.045	-0.107	0.402	0.198	0.104	0.587	0.865	0.874
NN	0.045	-0.084	0.400	0.218	0.134	0.316	0.653	0.539
DL1	0.088	-3.227	0.114	-0.266	0.237	-1.147	0.338	0.049
Hist	0.043	0.000	0.430	0.430	0.162	0.000	0.379	0.379
Market	NA	NA	0.430	0.430	NA	NA	0.379	0.379
2. Cumulative								
OLS	0.045	-0.078	0.458	0.216	0.179	-0.230	0.304	0.165
PLS	0.044	-0.058	0.448	0.151	0.187	-0.343	0.293	0.144
Lasso	0.043	-0.011	0.370	0.271	0.166	-0.056	0.351	0.318
Ridge	0.043	0.003	0.500	0.400	0.169	-0.089	0.323	0.205
ElasticNet	0.043	-0.005	0.404	0.306	0.165	-0.047	0.351	0.318
PCR	0.043	-0.020	0.406	0.167	0.179	-0.222	0.368	0.261
SVM	0.044	-0.032	0.533	0.373	0.171	-0.125	0.316	0.228
KNN	0.044	-0.042	0.555	0.475	0.133	0.326	0.676	0.578
RF	0.045	-0.116	0.525	0.321	0.107	0.564	0.962	0.996
NN	0.052	-0.480	0.484	0.323	0.171	-0.125	0.523	0.291
DL1	0.131	-8.293	0.474	0.241	0.293	-2.278	0.496	0.199
Hist	0.043	0.000	0.430	0.430	0.162	0.000	0.379	0.379
Market	NA	NA	0.430	0.430	NA	NA	0.379	0.379

[a] Column panel 'A' is based on one month returns predicted every month.

[b] Column panel 'B' is based on twelve-month returns predicted every month.

[1] Row panel '1' is based on models refitted with tuning every month using a rolling estimation window of 600 months.

[2] Row panel '2' is based on models refitted with tuning every month using a cumulative estimation window of minimum 600 months.

[*] The long-only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long-short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re-evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk-free rate.

[†] RMSE = root mean square error ; OOSR2 = out-of-sample R-squared ; SR_LO = Sharpe ratio of long-only portfolio ; SR_LS Sharpe ratio of long-short portfolio

Table 9: Performance with US data, VRP subsample

	A. Monthly				*B. Annual*			
MODEL	A.RMSE	A.OOSR2	A.SR_LO	A.SR_LS	B.RMSE	B.OOSR2	B.SR_LO	B.SR_LS
1. Rolling								
OLS	0.043	-0.139	0.752	0.740	0.150	0.273	0.725	0.454
PLS	0.043	-0.156	0.718	0.763	0.158	0.189	0.961	0.940
Lasso	0.040	0.034	0.567	0.575	0.145	0.316	0.841	0.702
Ridge	0.040	0.011	0.718	0.696	0.143	0.341	1.019	1.015
ElasticNet	0.040	0.002	0.524	0.471	0.138	0.383	0.961	0.940
PCR	0.044	-0.197	0.716	0.660	0.141	0.354	0.961	0.940
SVM	0.044	-0.193	0.831	0.802	0.157	0.198	0.841	0.702
KNN	0.040	0.001	0.777	0.563	0.125	0.491	0.961	0.940
RF	0.042	-0.071	0.522	0.343	0.107	0.631	1.019	1.015
NN	0.041	-0.021	0.831	0.844	0.105	0.643	1.019	1.015
DL1	0.141	-11.221	0.559	0.147	0.238	-0.838	0.808	0.492
Hist	0.040	0.000	0.546	0.546	0.176	0.000	0.340	0.340
Market	NA	NA	0.546	0.546	NA	NA	0.340	0.340
2. Cumulative								
OLS	0.043	-0.131	0.659	0.588	0.168	0.016	0.740	0.345
PLS	0.041	-0.054	0.879	0.855	0.201	-0.398	0.852	0.555
Lasso	0.040	0.005	0.517	0.473	0.199	-0.376	0.250	0.130
Ridge	0.040	0.010	0.736	0.741	0.174	-0.051	0.852	0.555
ElasticNet	0.040	0.000	0.659	0.645	0.188	-0.232	0.250	0.130
PCR	0.042	-0.104	0.656	0.512	0.171	-0.012	0.852	0.555
SVM	0.043	-0.141	0.832	0.793	0.199	-0.377	0.852	0.555
KNN	0.040	-0.011	1.012	0.957	0.136	0.356	0.836	0.673
RF	0.043	-0.126	0.644	0.416	0.108	0.596	0.805	0.632
NN	0.042	-0.105	0.724	0.651	0.116	0.531	0.852	0.555
DL1	0.252	-38.402	-0.048	-0.621	0.228	-0.814	0.425	0.070
Hist	0.040	0.000	0.546	0.546	0.170	0.000	0.340	0.340
Market	NA	NA	0.546	0.546	NA	NA	0.340	0.340

[a] Column panel 'A' is based on one month returns predicted every month.

[b] Column panel 'B' is based on twelve-month returns predicted every month.

[1] Row panel '1' is based on models refitted with tuning every month using a rolling estimation window of 180 months.

[2] Row panel '2' is based on models refitted with tuning every month using a cumulative estimation window of minimum 180 months.

[*] The long-only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long-short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re-evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk-free rate.

[†] RMSE = root mean square error ; OOSR2 = out-of-sample R-squared ; SR_LO = Sharpe ratio of long-only portfolio ; SR_LS Sharpe ratio of long-short portfolio

Table 10: Performance with US data, VRP subsample excluding the VRP

	A. Monthly				*B. Annual*			
MODEL	A.RMSE	A.OOSR2	A.SR_LO	A.SR_LS	B.RMSE	B.OOSR2	B.SR_LO	B.SR_LS
1. Rolling								
OLS	0.043	-0.159	0.866	0.948	0.144	0.334	0.633	0.225
PLS	0.043	-0.133	0.592	0.459	0.131	0.441	0.852	0.555
Lasso	0.040	0.021	0.649	0.726	0.141	0.359	0.739	0.408
Ridge	0.040	0.012	0.829	0.900	0.134	0.422	0.909	0.597
ElasticNet	0.040	0.032	0.653	0.734	0.136	0.402	0.852	0.555
PCR	0.042	-0.076	0.596	0.442	0.125	0.498	0.852	0.555
SVM	0.050	-0.509	0.587	0.504	0.141	0.358	0.841	0.702
KNN	0.042	-0.060	0.627	0.434	0.124	0.502	0.961	0.940
RF	0.043	-0.118	0.486	0.175	0.104	0.652	0.891	0.754
NN	0.043	-0.130	0.745	0.716	0.117	0.558	1.019	1.015
DL1	0.157	-14.114	0.422	-0.068	0.307	-2.052	0.103	-0.150
Hist	0.040	0.000	0.546	0.546	0.176	0.000	0.340	0.340
Market	NA	NA	0.546	0.546	NA	NA	0.340	0.340
2. Cumulative								
OLS	0.044	-0.223	0.602	0.514	0.162	0.086	0.740	0.345
PLS	0.043	-0.131	0.385	0.114	0.164	0.061	0.852	0.555
Lasso	0.040	-0.006	0.503	0.447	0.193	-0.301	0.250	0.130
Ridge	0.041	-0.025	0.659	0.618	0.168	0.020	0.852	0.555
ElasticNet	0.040	-0.013	0.627	0.615	0.188	-0.232	0.250	0.130
PCR	0.042	-0.108	0.613	0.501	0.153	0.182	0.852	0.555
SVM	0.051	-0.597	0.669	0.634	0.193	-0.297	0.852	0.555
KNN	0.042	-0.085	0.508	0.298	0.130	0.412	0.961	0.940
RF	0.043	-0.144	0.633	0.471	0.106	0.613	0.891	0.754
NN	0.044	-0.195	0.639	0.525	0.113	0.554	1.019	1.015
DL1	0.157	-14.334	0.213	-0.241	0.269	-1.507	0.253	0.134
Hist	0.040	0.000	0.546	0.546	0.170	0.000	0.340	0.340
Market	NA	NA	0.546	0.546	NA	NA	0.340	0.340

[a] Column panel 'A' is based on one month returns predicted every month.

[b] Column panel 'B' is based on twelve-month returns predicted every month.

[1] Row panel '1' is based on models refitted with tuning every month using a rolling estimation window of 180 months.

[2] Row panel '2' is based on models refitted with tuning every month using a cumulative estimation window of minimum 180 months.

[*] The long-only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long-short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re-evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk-free rate.

[†] RMSE = root mean square error ; OOSR2 = out-of-sample R-squared ; SR_LO = Sharpe ratio of long-only portfolio ; SR_LS Sharpe ratio of long-short portfolio

8.2 Appendix B

This section contains tables describing the accuracy of the *direction* our predictions.

Table 11: Prediction direction performance with UK data, full sample

	A. Monthly					*B. Annual*				
MODEL	A.ACC	A.TPR	A.TNR	A.FNR	A.FPR	B.ACC	B.TPR	B.TNR	B.FNR	B.FPR
1. Rolling										
OLS	0.60	0.87	0.22	0.13	0.78	0.77	0.94	0.38	0.06	0.62
PLS	0.59	0.92	0.14	0.08	0.86	0.76	0.95	0.31	0.05	0.69
Lasso	0.58	0.77	0.32	0.23	0.68	0.68	0.86	0.24	0.14	0.76
Ridge	0.58	0.89	0.14	0.11	0.86	0.75	0.95	0.26	0.05	0.74
ElasticNet	0.57	0.84	0.19	0.16	0.81	0.72	0.93	0.21	0.07	0.79
PCR	0.59	0.89	0.17	0.11	0.83	0.73	0.93	0.26	0.07	0.74
SVM	0.63	0.89	0.25	0.11	0.75	0.77	0.93	0.38	0.07	0.62
KNN	0.56	0.83	0.17	0.17	0.83	0.80	0.94	0.45	0.06	0.55
RF	0.56	0.78	0.24	0.22	0.76	0.85	0.95	0.60	0.05	0.40
NN	0.59	0.87	0.20	0.13	0.80	0.79	0.93	0.45	0.07	0.55
DL1	0.45	0.12	0.92	0.88	0.08	0.52	0.40	0.81	0.60	0.19
Hist	0.60	0.78	0.34	0.22	0.66	0.58	0.74	0.21	0.26	0.79
Market	0.58	1.00	0.00	0.00	1.00	0.70	1.00	0.00	0.00	1.00
2. Cumulative										
OLS	0.62	0.94	0.17	0.06	0.83	0.74	0.92	0.31	0.08	0.69
PLS	0.58	0.95	0.07	0.05	0.93	0.67	0.89	0.14	0.11	0.86
Lasso	0.55	0.61	0.46	0.39	0.54	0.54	0.62	0.33	0.38	0.67
Ridge	0.60	0.93	0.14	0.07	0.86	0.73	0.95	0.21	0.05	0.79
ElasticNet	0.56	0.73	0.31	0.27	0.69	0.68	0.89	0.17	0.11	0.83
PCR	0.56	0.93	0.03	0.07	0.97	0.73	0.96	0.17	0.04	0.83
SVM	0.58	0.92	0.12	0.08	0.88	0.73	0.89	0.36	0.11	0.64
KNN	0.51	0.77	0.14	0.23	0.86	0.80	0.90	0.55	0.10	0.45
RF	0.58	0.69	0.42	0.31	0.58	0.82	0.93	0.55	0.07	0.45
NN	0.60	0.93	0.14	0.07	0.86	0.78	0.88	0.55	0.12	0.45
DL1	0.44	0.34	0.58	0.66	0.42	0.57	0.68	0.31	0.32	0.69
Hist	0.55	0.61	0.46	0.39	0.54	0.51	0.59	0.33	0.41	0.67
Market	0.58	1.00	0.00	0.00	1.00	0.70	1.00	0.00	0.00	1.00

[a] Column panel 'A' is based on one month returns predicted every month.

[b] Column panel 'B' is based on twelve-month returns predicted every month.

[1] Row panel '1' is based on models refitted with tuning every month using a rolling estimation window of 120 months.

[2] Row panel '2' is based on models refitted with tuning every month using a cumulative estimation window of minimum 120 months.

[*] ACC = accuracy ; TPR = true positive rate ; TNR = true negative rate ; FPR = false positive rate ; FNR = false negative rate

Table 12: Prediction direction performance with UK data, VRP subsample

	A. Monthly					*B. Annual*				
MODEL	A.ACC	A.TPR	A.TNR	A.FNR	A.FPR	B.ACC	B.TPR	B.TNR	B.FNR	B.FPR
1. Rolling										
OLS	0.58	0.88	0.13	0.12	0.87	0.77	0.98	0.29	0.02	0.71
PLS	0.58	0.88	0.13	0.12	0.87	0.76	1.00	0.17	0.00	0.83
Lasso	0.59	0.88	0.15	0.12	0.85	0.70	0.95	0.09	0.05	0.91
Ridge	0.59	0.94	0.04	0.06	0.96	0.73	0.98	0.14	0.02	0.86
ElasticNet	0.59	0.96	0.02	0.04	0.98	0.72	1.00	0.06	0.00	0.94
PCR	0.60	0.90	0.13	0.10	0.87	0.72	0.99	0.09	0.01	0.91
SVM	0.66	0.92	0.26	0.08	0.74	0.76	0.96	0.26	0.04	0.74
KNN	0.54	0.79	0.15	0.21	0.85	0.76	0.94	0.34	0.06	0.66
RF	0.55	0.72	0.28	0.28	0.72	0.80	0.93	0.49	0.07	0.51
NN	0.62	0.93	0.15	0.07	0.85	0.79	0.90	0.51	0.10	0.49
DL1	0.52	0.36	0.77	0.64	0.23	0.56	0.77	0.06	0.23	0.94
Hist	0.61	0.90	0.17	0.10	0.83	0.64	0.88	0.06	0.12	0.94
Market	0.61	1.00	0.00	0.00	1.00	0.71	1.00	0.00	0.00	1.00
2. Cumulative										
OLS	0.60	0.99	0.00	0.01	1.00	0.73	0.96	0.17	0.04	0.83
PLS	0.54	0.83	0.09	0.17	0.91	0.66	0.87	0.17	0.13	0.83
Lasso	0.54	0.67	0.34	0.33	0.66	0.61	0.79	0.17	0.21	0.83
Ridge	0.61	0.97	0.04	0.03	0.96	0.72	1.00	0.06	0.00	0.94
ElasticNet	0.60	0.94	0.06	0.06	0.94	0.71	1.00	0.00	0.00	1.00
PCR	0.60	0.97	0.02	0.03	0.98	0.71	1.00	0.03	0.00	0.97
SVM	0.61	0.99	0.02	0.01	0.98	0.73	0.95	0.20	0.05	0.80
KNN	0.54	0.72	0.26	0.28	0.74	0.75	0.90	0.37	0.10	0.63
RF	0.62	0.83	0.30	0.17	0.70	0.83	0.98	0.49	0.02	0.51
NN	0.60	0.94	0.06	0.06	0.94	0.76	0.90	0.43	0.10	0.57
DL1	0.48	0.31	0.74	0.69	0.26	0.58	0.71	0.26	0.29	0.74
Hist	0.55	0.69	0.32	0.31	0.68	0.52	0.65	0.20	0.35	0.80
Market	0.61	1.00	0.00	0.00	1.00	0.71	1.00	0.00	0.00	1.00

[a] Column panel 'A' is based on one month returns predicted every month.

[b] Column panel 'B' is based on twelve-month returns predicted every month.

[1] Row panel '1' is based on models refitted with tuning every month using a rolling estimation window of 120 months.

[2] Row panel '2' is based on models refitted with tuning every month using a cumulative estimation window of minimum 120 months.

[*] ACC = accuracy ; TPR = true positive rate ; TNR = true negative rate ; FPR = false positive rate ; FNR = false negative rate

Table 13: Prediction direction performance with UK data, VRP subsample excluding the VRP

	A. Monthly					*B. Annual*				
MODEL	A.ACC	A.TPR	A.TNR	A.FNR	A.FPR	B.ACC	B.TPR	B.TNR	B.FNR	B.FPR
1. Rolling										
OLS	0.59	0.89	0.13	0.11	0.87	0.76	0.98	0.26	0.02	0.74
PLS	0.60	0.93	0.09	0.07	0.91	0.76	0.99	0.20	0.01	0.80
Lasso	0.60	0.88	0.17	0.12	0.83	0.70	0.95	0.09	0.05	0.91
Ridge	0.60	0.94	0.06	0.06	0.94	0.72	0.98	0.11	0.02	0.89
ElasticNet	0.59	0.96	0.02	0.04	0.98	0.72	1.00	0.06	0.00	0.94
PCR	0.60	0.89	0.15	0.11	0.85	0.71	0.96	0.11	0.04	0.89
SVM	0.64	0.90	0.23	0.10	0.77	0.76	0.96	0.26	0.04	0.74
KNN	0.55	0.83	0.13	0.17	0.87	0.77	0.94	0.37	0.06	0.63
RF	0.57	0.82	0.19	0.18	0.81	0.82	0.95	0.49	0.05	0.51
NN	0.62	0.92	0.17	0.08	0.83	0.81	0.94	0.49	0.06	0.51
DL1	0.61	0.93	0.11	0.07	0.89	0.69	0.74	0.57	0.26	0.43
Hist	0.61	0.90	0.17	0.10	0.83	0.64	0.88	0.06	0.12	0.94
Market	0.61	1.00	0.00	0.00	1.00	0.71	1.00	0.00	0.00	1.00
2. Cumulative										
OLS	0.60	0.99	0.00	0.01	1.00	0.73	0.96	0.17	0.04	0.83
PLS	0.59	0.96	0.02	0.04	0.98	0.64	0.88	0.06	0.12	0.94
Lasso	0.54	0.67	0.34	0.33	0.66	0.61	0.79	0.17	0.21	0.83
Ridge	0.60	0.96	0.04	0.04	0.96	0.70	0.99	0.00	0.01	1.00
ElasticNet	0.60	0.93	0.09	0.07	0.91	0.71	1.00	0.00	0.00	1.00
PCR	0.60	0.97	0.02	0.03	0.98	0.71	0.99	0.03	0.01	0.97
SVM	0.61	1.00	0.02	0.00	0.98	0.73	0.95	0.20	0.05	0.80
KNN	0.52	0.78	0.13	0.22	0.87	0.76	0.90	0.43	0.10	0.57
RF	0.59	0.82	0.23	0.18	0.77	0.84	0.98	0.51	0.02	0.49
NN	0.63	0.97	0.11	0.03	0.89	0.81	0.93	0.51	0.07	0.49
DL1	0.43	0.25	0.70	0.75	0.30	0.65	0.83	0.20	0.17	0.80
Hist	0.55	0.69	0.32	0.31	0.68	0.52	0.65	0.20	0.35	0.80
Market	0.61	1.00	0.00	0.00	1.00	0.71	1.00	0.00	0.00	1.00

[a] Column panel 'A' is based on one month returns predicted every month.

[b] Column panel 'B' is based on twelve-month returns predicted every month.

[1] Row panel '1' is based on models refitted with tuning every month using a rolling estimation window of 120 months.

[2] Row panel '2' is based on models refitted with tuning every month using a cumulative estimation window of minimum 120 months.

[*] ACC = accuracy ; TPR = true positive rate ; TNR = true negative rate ; FPR = false positive rate ; FNR = false negative rate

Table 14: Prediction direction performance with US data, full sample

	A. Monthly					*B. Annual*				
MODEL	A.ACC	A.TPR	A.TNR	A.FNR	A.FPR	B.ACC	B.TPR	B.TNR	B.FNR	B.FPR
1. Rolling										
OLS	0.52	0.62	0.36	0.38	0.64	0.69	0.77	0.42	0.23	0.58
PLS	0.51	0.62	0.34	0.38	0.66	0.72	0.79	0.49	0.21	0.51
Lasso	0.57	0.89	0.07	0.11	0.93	0.77	0.95	0.17	0.05	0.83
Ridge	0.59	0.92	0.06	0.08	0.94	0.77	0.91	0.27	0.09	0.73
ElasticNet	0.58	0.90	0.07	0.10	0.93	0.79	1.00	0.08	0.00	0.92
PCR	0.59	0.86	0.14	0.14	0.86	0.78	0.90	0.34	0.10	0.66
SVM	0.57	0.75	0.27	0.25	0.73	0.69	0.79	0.33	0.21	0.67
KNN	0.60	0.79	0.29	0.21	0.71	0.80	0.89	0.50	0.11	0.50
RF	0.54	0.67	0.33	0.33	0.67	0.86	0.92	0.66	0.08	0.34
NN	0.53	0.69	0.29	0.31	0.71	0.79	0.83	0.65	0.17	0.35
DL1	0.46	0.41	0.54	0.59	0.46	0.37	0.32	0.56	0.68	0.44
Hist	0.61	1.00	0.00	0.00	1.00	0.78	1.00	0.00	0.00	1.00
Market	0.61	1.00	0.00	0.00	1.00	0.78	1.00	0.00	0.00	1.00
2. Cumulative										
OLS	0.54	0.61	0.45	0.39	0.55	0.67	0.74	0.44	0.26	0.56
PLS	0.50	0.50	0.49	0.50	0.51	0.63	0.73	0.28	0.27	0.72
Lasso	0.58	0.88	0.12	0.12	0.88	0.71	0.88	0.12	0.12	0.88
Ridge	0.57	0.82	0.18	0.18	0.82	0.66	0.77	0.29	0.23	0.71
ElasticNet	0.57	0.85	0.13	0.15	0.87	0.70	0.87	0.12	0.13	0.88
PCR	0.53	0.62	0.39	0.38	0.61	0.62	0.74	0.23	0.26	0.77
SVM	0.58	0.74	0.32	0.26	0.68	0.73	0.86	0.30	0.14	0.70
KNN	0.60	0.79	0.31	0.21	0.69	0.79	0.89	0.45	0.11	0.55
RF	0.53	0.64	0.36	0.36	0.64	0.87	0.94	0.65	0.06	0.35
NN	0.55	0.68	0.36	0.32	0.64	0.66	0.69	0.58	0.31	0.42
DL1	0.51	0.54	0.46	0.46	0.54	0.52	0.50	0.59	0.50	0.41
Hist	0.61	1.00	0.00	0.00	1.00	0.78	1.00	0.00	0.00	1.00
Market	0.61	1.00	0.00	0.00	1.00	0.78	1.00	0.00	0.00	1.00

[a] Column panel 'A' is based on one month returns predicted every month.

[b] Column panel 'B' is based on twelve-month returns predicted every month.

[1] Row panel '1' is based on models refitted with tuning every month using a rolling estimation window of 600 months.

[2] Row panel '2' is based on models refitted with tuning every month using a cumulative estimation window of minimum 600 months.

[*] ACC = accuracy ; TPR = true positive rate ; TNR = true negative rate ; FPR = false positive rate ; FNR = false negative rate

Table 15: Prediction direction performance with US data, VRP subsample

	A. Monthly					*B. Annual*				
MODEL	A.ACC	A.TPR	A.TNR	A.FNR	A.FPR	B.ACC	B.TPR	B.TNR	B.FNR	B.FPR
1. Rolling										
OLS	0.66	0.87	0.25	0.13	0.75	0.77	0.84	0.44	0.16	0.56
PLS	0.68	0.87	0.30	0.13	0.70	0.89	0.97	0.44	0.03	0.56
Lasso	0.65	0.93	0.08	0.07	0.92	0.85	0.94	0.41	0.06	0.59
Ridge	0.65	0.88	0.20	0.12	0.80	0.83	0.91	0.44	0.09	0.56
ElasticNet	0.65	0.94	0.08	0.06	0.92	0.89	0.99	0.41	0.01	0.59
PCR	0.65	0.86	0.25	0.14	0.75	0.87	0.96	0.44	0.04	0.56
SVM	0.68	0.92	0.20	0.08	0.80	0.82	0.89	0.44	0.11	0.56
KNN	0.65	0.84	0.27	0.16	0.73	0.84	0.92	0.44	0.08	0.56
RF	0.59	0.76	0.27	0.24	0.73	0.87	0.91	0.63	0.09	0.37
NN	0.65	0.84	0.28	0.16	0.72	0.88	0.91	0.74	0.09	0.26
DL1	0.53	0.58	0.42	0.42	0.58	0.65	0.70	0.41	0.30	0.59
Hist	0.66	1.00	0.00	0.00	1.00	0.84	1.00	0.00	0.00	1.00
Market	0.66	1.00	0.00	0.00	1.00	0.84	1.00	0.00	0.00	1.00
2. Cumulative										
OLS	0.64	0.87	0.18	0.13	0.82	0.80	0.87	0.44	0.13	0.56
PLS	0.63	0.77	0.35	0.23	0.65	0.81	0.90	0.37	0.10	0.63
Lasso	0.66	0.98	0.03	0.02	0.97	0.80	0.94	0.07	0.06	0.93
Ridge	0.66	0.92	0.17	0.08	0.83	0.82	0.89	0.44	0.11	0.56
ElasticNet	0.65	0.93	0.10	0.07	0.90	0.82	0.94	0.19	0.06	0.81
PCR	0.62	0.87	0.13	0.13	0.87	0.85	0.94	0.37	0.06	0.63
SVM	0.67	0.91	0.20	0.09	0.80	0.84	0.92	0.44	0.08	0.56
KNN	0.69	0.86	0.35	0.14	0.65	0.83	0.91	0.44	0.09	0.56
RF	0.60	0.74	0.32	0.26	0.68	0.87	0.93	0.59	0.07	0.41
NN	0.64	0.84	0.23	0.16	0.77	0.86	0.89	0.70	0.11	0.30
DL1	0.47	0.55	0.33	0.45	0.67	0.46	0.44	0.52	0.56	0.48
Hist	0.66	1.00	0.00	0.00	1.00	0.84	1.00	0.00	0.00	1.00
Market	0.66	1.00	0.00	0.00	1.00	0.84	1.00	0.00	0.00	1.00

[a] Column panel 'A' is based on one month returns predicted every month.

[b] Column panel 'B' is based on twelve-month returns predicted every month.

[1] Row panel '1' is based on models refitted with tuning every month using a rolling estimation window of 180 months.

[2] Row panel '2' is based on models refitted with tuning every month using a cumulative estimation window of minimum 180 months.

[*] ACC = accuracy ; TPR = true positive rate ; TNR = true negative rate ; FPR = false positive rate ; FNR = false negative rate

Table 16: Prediction direction performance with US data, VRP subsample excluding the VRP

	A. Monthly					*B. Annual*				
MODEL	A.ACC	A.TPR	A.TNR	A.FNR	A.FPR	B.ACC	B.TPR	B.TNR	B.FNR	B.FPR
1. Rolling										
OLS	0.69	0.89	0.28	0.11	0.72	0.79	0.86	0.44	0.14	0.56
PLS	0.63	0.86	0.18	0.14	0.82	0.89	0.97	0.44	0.03	0.56
Lasso	0.69	0.97	0.12	0.03	0.88	0.86	0.94	0.44	0.06	0.56
Ridge	0.68	0.90	0.23	0.10	0.77	0.83	0.90	0.48	0.10	0.52
ElasticNet	0.69	0.98	0.10	0.02	0.90	0.89	0.97	0.44	0.03	0.56
PCR	0.63	0.84	0.20	0.16	0.80	0.89	0.98	0.44	0.02	0.56
SVM	0.63	0.88	0.13	0.12	0.87	0.85	0.93	0.44	0.07	0.56
KNN	0.63	0.81	0.28	0.19	0.72	0.87	0.94	0.48	0.06	0.52
RF	0.57	0.73	0.25	0.27	0.75	0.87	0.91	0.63	0.09	0.37
NN	0.66	0.86	0.28	0.14	0.72	0.84	0.89	0.63	0.11	0.37
DL1	0.46	0.41	0.55	0.59	0.45	0.60	0.62	0.48	0.38	0.52
Hist	0.66	1.00	0.00	0.00	1.00	0.84	1.00	0.00	0.00	1.00
Market	0.66	1.00	0.00	0.00	1.00	0.84	1.00	0.00	0.00	1.00
2. Cumulative										
OLS	0.65	0.90	0.15	0.10	0.85	0.81	0.89	0.44	0.11	0.56
PLS	0.61	0.90	0.05	0.10	0.95	0.85	0.94	0.37	0.06	0.63
Lasso	0.65	0.98	0.00	0.02	1.00	0.79	0.94	0.00	0.06	1.00
Ridge	0.64	0.92	0.10	0.08	0.90	0.84	0.92	0.41	0.08	0.59
ElasticNet	0.66	0.97	0.05	0.03	0.95	0.81	0.94	0.15	0.06	0.85
PCR	0.64	0.91	0.12	0.09	0.88	0.87	0.96	0.41	0.04	0.59
SVM	0.67	0.92	0.17	0.08	0.83	0.85	0.93	0.44	0.07	0.56
KNN	0.60	0.79	0.22	0.21	0.78	0.84	0.92	0.44	0.08	0.56
RF	0.63	0.82	0.25	0.18	0.75	0.89	0.94	0.59	0.06	0.41
NN	0.64	0.87	0.20	0.13	0.80	0.86	0.89	0.70	0.11	0.30
DL1	0.53	0.56	0.47	0.44	0.53	0.70	0.81	0.11	0.19	0.89
Hist	0.66	1.00	0.00	0.00	1.00	0.84	1.00	0.00	0.00	1.00
Market	0.66	1.00	0.00	0.00	1.00	0.84	1.00	0.00	0.00	1.00

[a] Column panel 'A' is based on one month returns predicted every month.

[b] Column panel 'B' is based on twelve-month returns predicted every month.

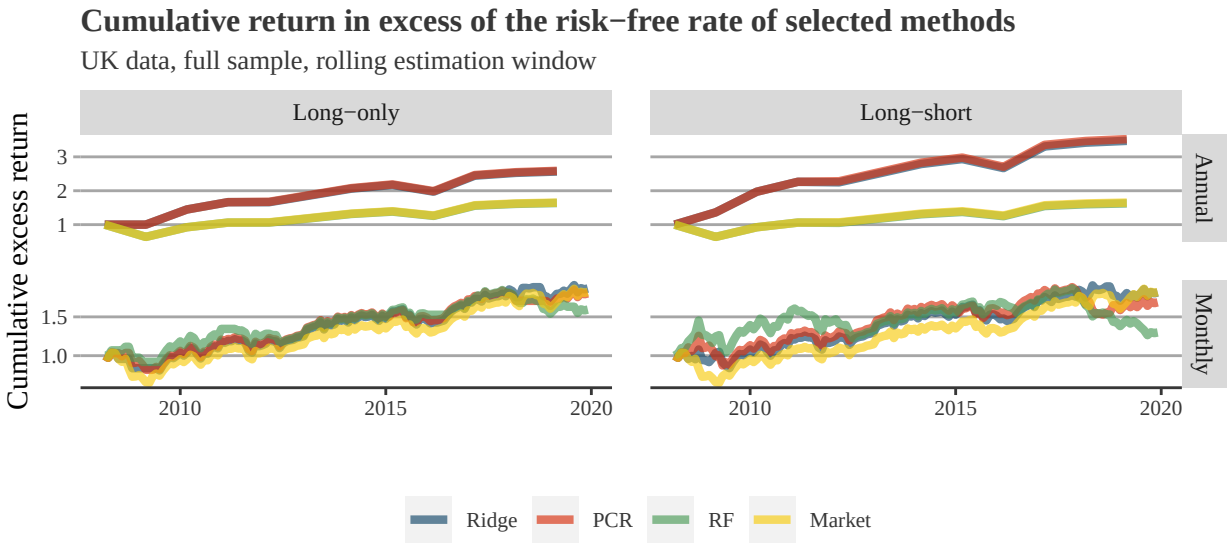
[1] Row panel '1' is based on models refitted with tuning every month using a rolling estimation window of 180 months.

[2] Row panel '2' is based on models refitted with tuning every month using a cumulative estimation window of minimum 180 months.

[*] ACC = accuracy ; TPR = true positive rate ; TNR = true negative rate ; FPR = false positive rate ; FNR = false negative rate

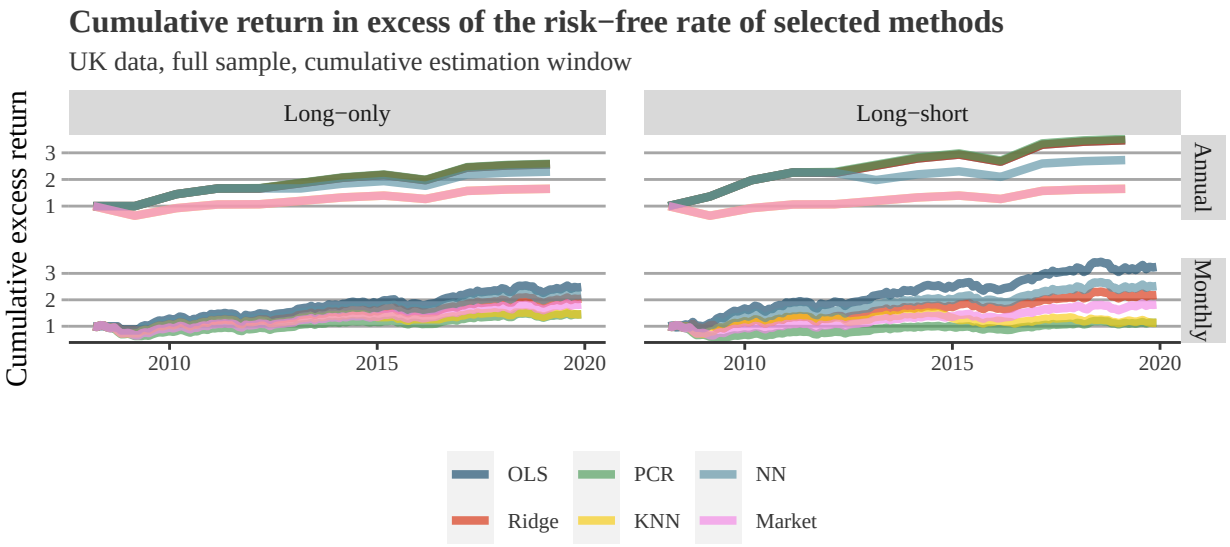
8.3 Appendix C

This section contains plots illustrating the cumulative excess returns of different models.



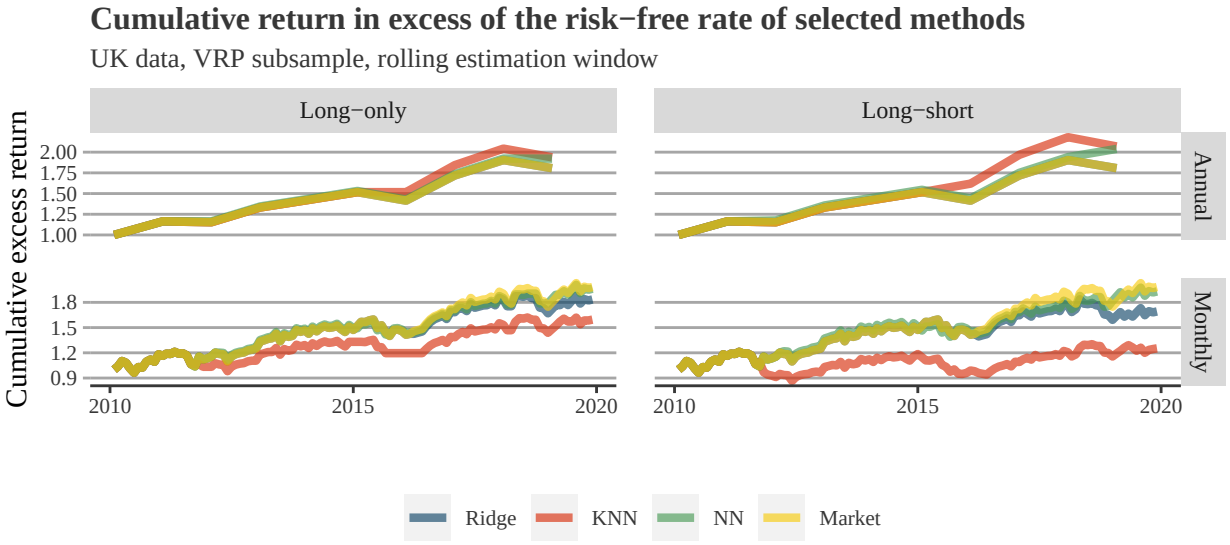
The models with the best average ranking over root mean square error, out−of−sample R−squared, long−only Sharpe ratio and long−short Sharpe ratio are displayed. The base value for all strategies is set to one and develops with the excess return that results from the strategy. The long−only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long−short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re−evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk−free rate.

Figure 4



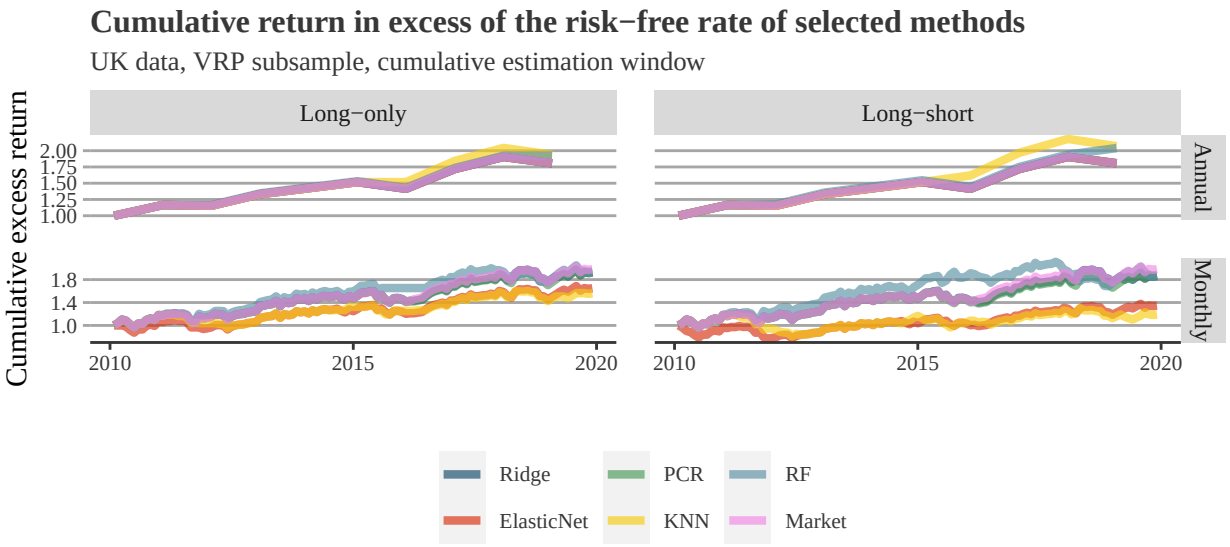
The models with the best average ranking over root mean square error, out−of−sample R−squared, long−only Sharpe ratio and long−short Sharpe ratio are displayed. The base value for all strategies is set to one and develops with the excess return that results from the strategy. The long−only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long−short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re−evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk−free rate.

Figure 5



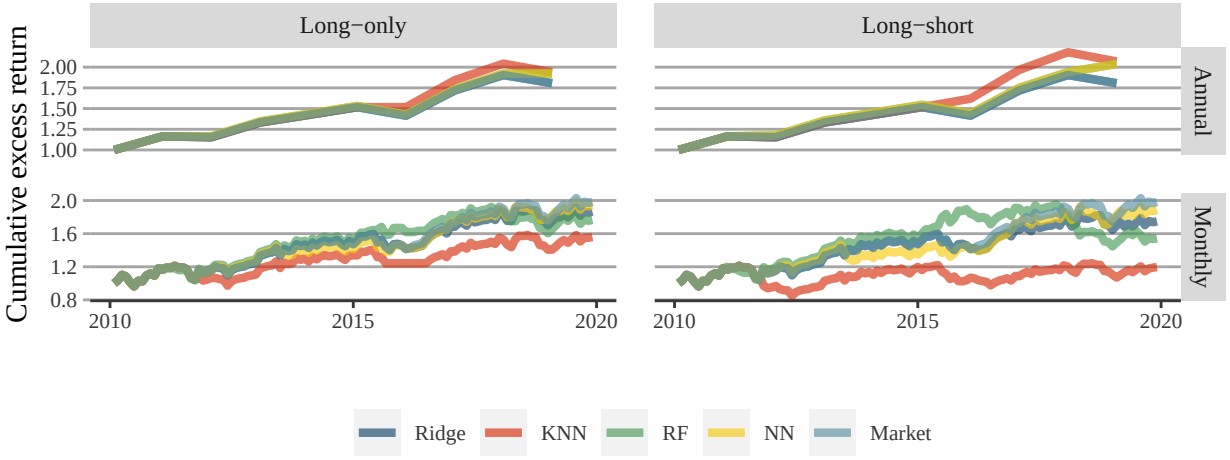
The models with the best average ranking over root mean square error, out−of−sample R−squared, long−only Sharpe ratio and long−short Sharpe ratio are displayed. The base value for all strategies is set to one and develops with the excess return that results from the strategy. The long−only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long−short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re−evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk−free rate.

Figure 6



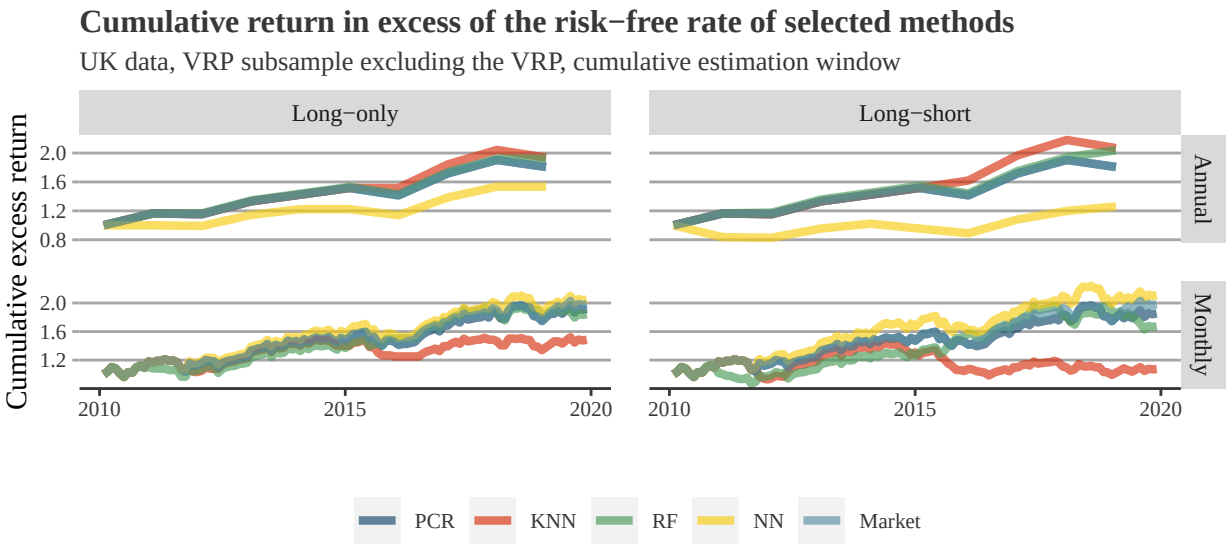
The models with the best average ranking over root mean square error, out−of−sample R−squared, long−only Sharpe ratio and long−short Sharpe ratio are displayed. The base value for all strategies is set to one and develops with the excess return that results from the strategy. The long−only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long−short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re−evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk−free rate.

Figure 7



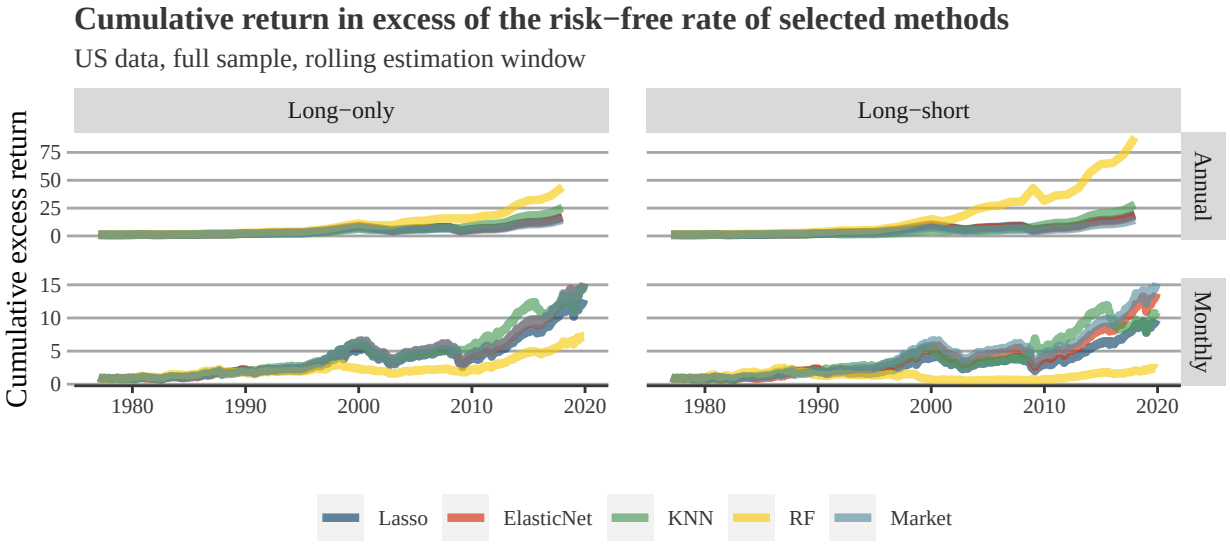
The models with the best average ranking over root mean square error, out−of−sample R−squared, long−only Sharpe ratio and long−short Sharpe ratio are displayed. The base value for all strategies is set to one and develops with the excess return that results from the strategy. The long−only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long−short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re−evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk−free rate.

Figure 8



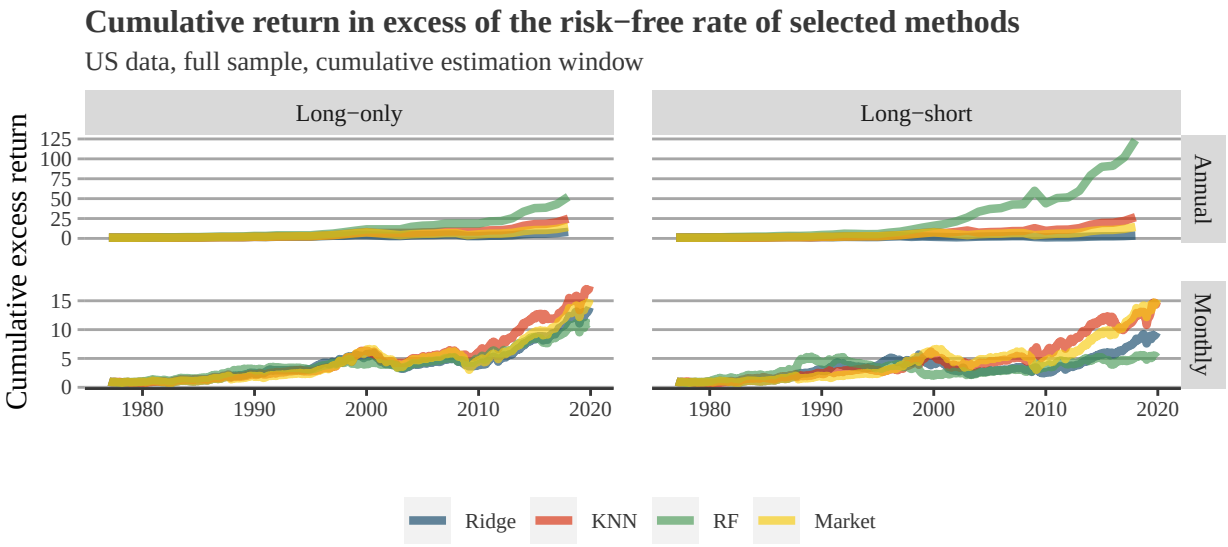
The models with the best average ranking over root mean square error, out−of−sample R−squared, long−only Sharpe ratio and long−short Sharpe ratio are displayed. The base value for all strategies is set to one and develops with the excess return that results from the strategy. The long−only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long−short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re−evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk−free rate.

Figure 9



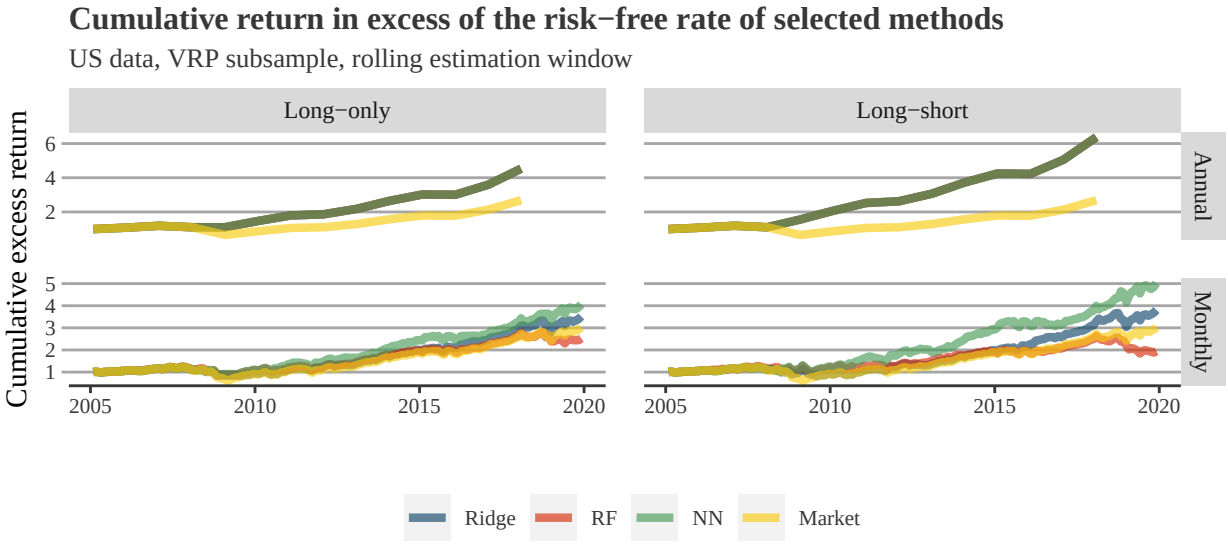
The models with the best average ranking over root mean square error, out−of−sample R−squared, long−only Sharpe ratio and long−short Sharpe ratio are displayed. The base value for all strategies is set to one and develops with the excess return that results from the strategy. The long−only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long−short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re−evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk−free rate.

Figure 10



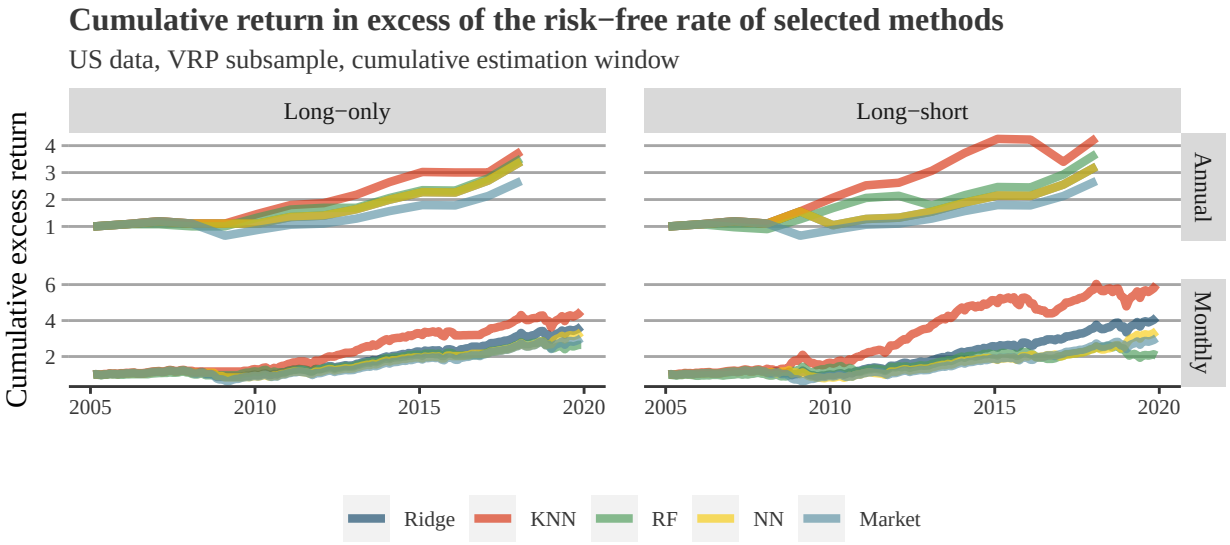
The models with the best average ranking over root mean square error, out−of−sample R−squared, long−only Sharpe ratio and long−short Sharpe ratio are displayed. The base value for all strategies is set to one and develops with the excess return that results from the strategy. The long−only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long−short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re−evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk−free rate.

Figure 11



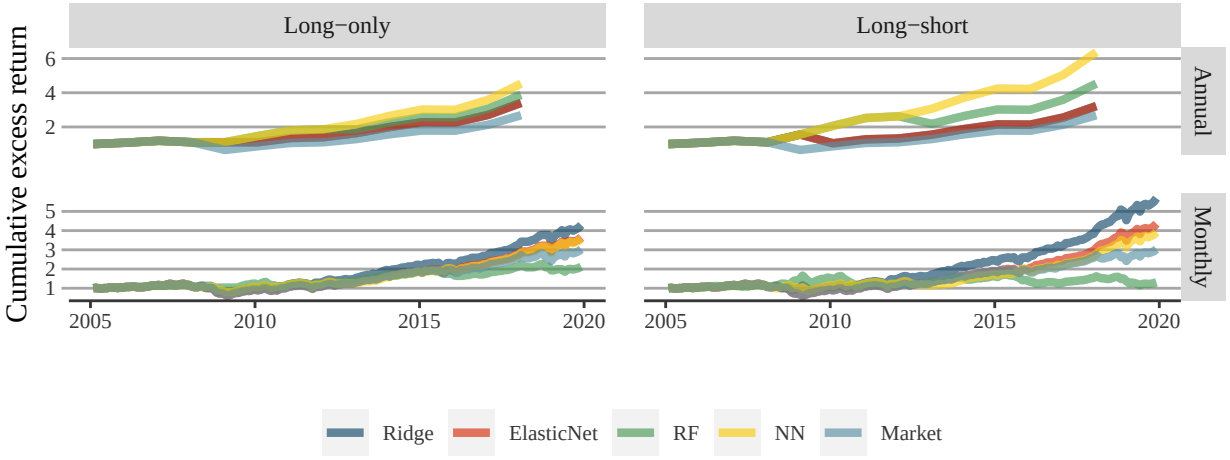
The models with the best average ranking over root mean square error, out−of−sample R−squared, long−only Sharpe ratio and long−short Sharpe ratio are displayed. The base value for all strategies is set to one and develops with the excess return that results from the strategy. The long−only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long−short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re−evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk−free rate.

Figure 12



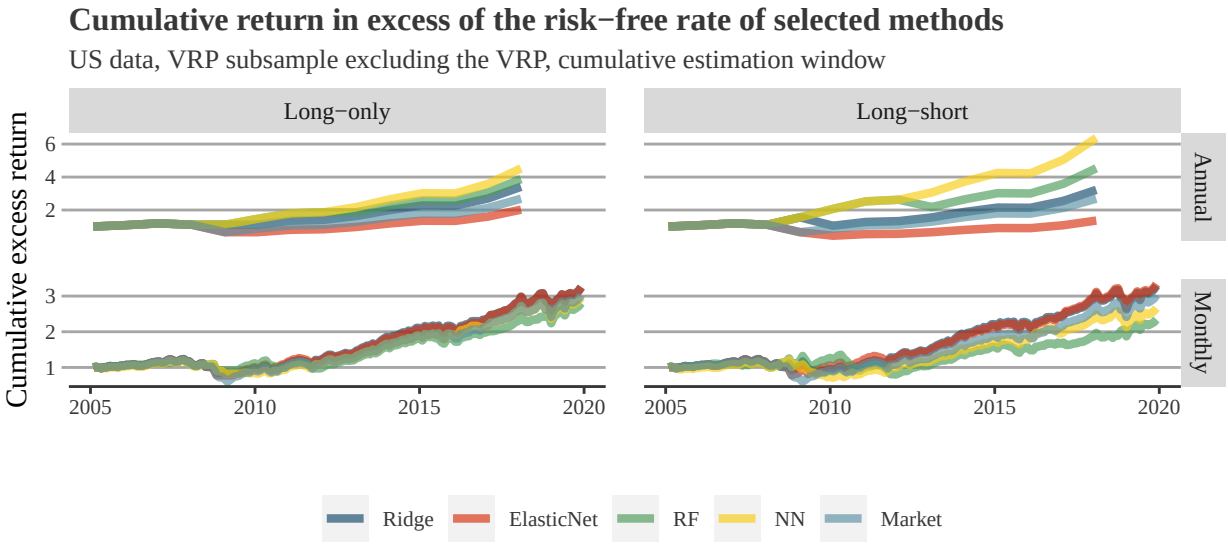
The models with the best average ranking over root mean square error, out−of−sample R−squared, long−only Sharpe ratio and long−short Sharpe ratio are displayed. The base value for all strategies is set to one and develops with the excess return that results from the strategy. The long−only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long−short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re−evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk−free rate.

Figure 13



The models with the best average ranking over root mean square error, out−of−sample R−squared, long−only Sharpe ratio and long−short Sharpe ratio are displayed. The base value for all strategies is set to one and develops with the excess return that results from the strategy. The long−only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long−short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re−evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk−free rate.

Figure 14



The models with the best average ranking over root mean square error, out−of−sample R−squared, long−only Sharpe ratio and long−short Sharpe ratio are displayed. The base value for all strategies is set to one and develops with the excess return that results from the strategy. The long−only strategy buys the market portfolio when positive or zero return is predicted and is otherwise not invested. The long−short strategy buys when positive or zero return is predicted and otherwise it shorts the market, i.e., generates the negative value of the realized excess return. Strategies based on annual prediction re−evaluates once per year, those based on monthly returns do it every month. Excess returns are returns in excess of the risk−free rate.

Figure 15