Erasmus University Rotterdam
Erasmus School of Economics
Bachelor Thesis Econometrics and Operations Research

Comparing different targeting techniques in targeted random forest

Rens van Steenveldt (531303ls)

Supervisor:	O'Neill, EP
Second assessor:	dr. Pick, A
Date final version:	July 2, 2023

Abstract

Forecasting out-of-sample stock market excess returns turn out to be a very difficult task (Campbell & Thompson, 2008). Therefore, ways of improving the forecast accuracy of methods with respect to this variable are of great relevance. In that context, random forests turn out to be a great machine learning technique in forecasting macroeconomic variables. Selecting the most important variables in a step prior to the estimation appears to improve the forecasting accuracy (Medeiros et al., 2021; Borup et al., 2023). In this paper, that same conclusion is investigated for predicting the stock market excess return, using data from Amit Goyal's website and the FREDMD database, for four different targeting techniques: LASSO regression, permutation feature importance, feature importance based on mean decrease in mean squared error, which we call variance of the responses and feature importance based on the VSURF R package from Genuer et al. (2015), which computes variable importance based on out-of-bag errors (Breiman, 1996). The four models combined with the subsequent random forest are compared with the ordinary random forest using a Diebold-Mariano test and the techniques are compared with each other using a model confidence set. All so called targeted random forests (TRF) turn out to perform better than the ordinary RF and the TRF based on the VSURF package appears to be the most accurate one.

Keywords: Random forest, targeted random forest, variable importance.

1 Introduction

Machine learning techniques are becoming more and more popular in recent years for the purpose of forecasting macroeconomic variables in high dimensional settings. Common known machine learning algorithms are linear regressions, decision trees, support vector machines, neural networks etc. From the class of decision tree based methods, one particular machine learning technique stands out, random forest (RF). This technique is introduced by Breiman (2001) and it combines the output of multiple decision trees to form a single result.

Decision trees are prediction and classification mechanisms that have evolved into highly transversal mechanisms in the world of machine learning, among others (De Ville, 2013). Each node in a decision tree has a question with two possible answers; a decision has to be made. If there are multiple nodes in the tree, those questions act as a means of splitting the data. However, regular decision trees can be prone to problems such as bias and overfitting. Therefore, ensemble methods showed up. Ensemble methods aggregate their predictions to come up with the most popular result. An example of an ensemble method is bagging (bootstrapping the data plus using the aggregate to make a prediction). It is a commonly used approach to deal with noisy data sets (Dietterich, 2000). An extension of the bagging method is a random forest. It uses both bagging and feature randomness to create an almost uncorrelated forest (the trees are decorrelated, but they are not fully uncorrelated) of decision trees. By generating a random subset of features, feature randomness guarantees low correlation among decision trees. This distinction is crucial in comparing decision trees with random forests, as decision trees evaluate all potential feature splits while random forests only choose a subset of features. Because of the widespread usage, versatility, and adaptability, random forest regression is highly favoured, especially for accommodating nonlinearity in data and effectively handling high-dimensional data sets. According to Fortin-Gagnon et al. (2022), RF regression is the best performing machine learning model in predicting industrial production and others conclude the same in other areas (Medeiros et al., 2021). RF regression can also be employed to carry out time series forecasting (Tyralis & Papacharalampous, 2017).

A recent trend in the prediction of macroeconomic variables is a so called targeting step before the estimation of an algorithm. Targeting refers to the fact that at first, the most important variables from a large data set are selected and after that, those fewer variables are fed into an algorithm. In that way, so called 'noise variables' are removed in order to only predict with variables that are relevant.

The primary focus of this paper pertains to maximizing the accuracy of predictions by RF for the excess return of the stock market. To investigate this issue, different targeting techniques before the estimation of a random forest are compared, in order to give more accurate predictions for the excess return on the stock market, where the forecasts are based on a macroeconomic application. Making accurate predictions is highly relevant for, for example, investors. The better their predictions for the excess returns are, the more money they make. Moreover, it holds scientific significance as it is advantageous to generate approaches aimed at enhancing the predictive capabilities of models, such as random forests.

Four targeting techniques are examined: LASSO regression, permutation importance, reduction in mean squared error (MSE), which we call variance of the responses and a technique

based on the VSURF package (Genuer et al., 2015). The VSURF package calculates variable importance based on out-of-bag errors, see Section 3.2.4. The forecast performance of the TRF models are compared with the ordinary RF using a Diebold-Mariano (DM) test (Diebold & Mariano, 2002) and a model confidence set (MCS) (Hansen et al., 2011) is calculated to observe which model is the best performing model. A combined monthly dataset of FRED-MD and Amit Goyal's website over a period of January 1970 - December 2018 is used.

In line with the existing literature, we find that TRF outperforms ordinary RF in all settings. More specifically, the TRF method based on the VSURF package is the best performing TRF method. This is confirmed by the MCS approach. A thing that stands out is that the absolute performance of all models decreases when the amount of selected predictors increases, indicating that all models perform better when they make use of less predictors. "Selected predictors" refers to the most important predictors rather than a number of randomly selected predictors.

Given the recency of this topic, not much has been written on the subject. In the past, machine learning algorithms were not taken entirely seriously because of the lack of interpretability, but nowadays more and more techniques are being developed, such as permutation feature importance, allowing these more accurate machine learning techniques to be used to a greater extent. This study contributes to the existing literature, because it proposes new combinations of existing methods in order to forecast macroeconomic variables. Macroeconomic forecasting is very important at the moment. For example, government officials and business managers employ economic forecasts to ascertain fiscal and monetary policies and plan future operating activities, respectively. Random forests have only recently been applied to macroeconomic forecasting, so it is highly relevant to investigate new ways in order to improve the forecast accuracy of this technique with respect to macroeconomic variables. In addition, in recent years more variables are available. Some people call it the big data era. Having access to much bigger data sets allows us to make better predictions and because machine learning methods such as random forest are known for being good at handling a lot of data, this could be a golden combination.

1.1 Literature review

Making predictions as accurate as possible with new methods has already been a main goal of many papers. For example, Tan et al. (2019) wrote an article about the selection of stocks with a random forest. They try to predict stock prices and based on that, select the best stocks in the Chinese stock market. They conclude that machine learning is probably helpful for quantitative traders in building their strategies. Rapach and Zhou (2020) also try new machine learning techniques to forecast stock returns more accurately.

The principle of adding a targeting step prior to an estimation step was introduced by Bai and Ng (2008), who predict with factor-based approaches. In their paper, they select the predictors in the targeting step based on hard and soft thresholding rules. They find improvements at all forecast horizons by estimating the factors using fewer but informative predictors. Also, when they allow for non-linearity, there are additional gains. Borup and Schütte (2022) forecast employment growth with Google Trends data. They also apply the idea of targeting predictors, because Google Trends have a few very strong predictors, surrounded by a very large amount of weak or irrelevant predictors. In that setting it can be especially advantageous to only use

a subset of such a big data set, in order not to disturb the estimation by noise variables. They also make use of a soft thresholding rule in the targeting step, more specifically, they use the elastic-net estimator (Zou & Hastie, 2005). Genuer et al. (2010) propose random forest as a variable selection method. They say that the general strategy is to make a ranking of the explanatory variables using the random forests score of importance and a stepwise ascending variable introduction strategy. They also published a paper about an R package for variable selection using RF (Genuer et al., 2015), which will be used a lot in this study.

In a RF framework, targeting predictors is extensively examined by Medeiros et al. (2021), Borup et al. (2023) and Charles and Darné (2022). Borup et al. (2023) wrote an extensive article, fully focused on the impact of targeting predictors via LASSO regression in a targeted random forest regression (TRF), providing a theoretical and empirical analysis. Based on their empirical analysis, they show that targeting predictors is particularly useful when a medium-sized set of predictors is selected, of about 5%-30% of the initial set of predictors. Sometimes, a gain of almost 13% in predictive accuracy of TRF relative to ordinary RF is achieved. An important thing they highlight is the trade-off between the correlation across individual trees in the forest and the tree strength of the individual trees. When there is much targeting (i.e. only a few variables are selected), the individual trees are very strong, because they are mainly based on relevant predictors, but in that case there is more correlation between individual trees, because there are less possibilities for differences. Due to this trade-off, a medium-sized amount of selected predictors is preferable. Medeiros et al. (2021) forecast inflation in the U.S. with machine learning methods and the availability of new data. They conclude that RF is the superior model in forecasting performance, but they also estimate a targeted RF model, where the regressors are selected using adaptive Lasso (adaLASSO). The results imply that for a forecast horizon of 1, the TRF outperforms the regular RF, but for longer forecast horizons, ordinary RF is still the best performing model. Lastly, Charles and Darné (2022) compare different targeting techniques for the estimation of a random forest. It is the only paper that compares different targeting techniques, rather than analyses whether an initial targeting is step is even advantageous. They compare soft thresholding methods, hard thresholding methods based on an univariate predictive regression and screening techniques based on the sure independence screening (SIS) procedure from Fan and Lv (2008). Their results show that the forecasting performance of the TRF models based on elastic-net and DC-SIS approaches outperform the ordinary RF models on almost all predicted variables.

The remainder of this paper is organized as follows. In Section 2, the data set used in this research is introduced. In Section 3 the random forest algorithm is explained, together with the several targeting techniques, the forecasting setting and the methods to compare the different models. In Section 4, we present our forecasting results and Section 5 concludes and discusses limitations and possible ideas for future research.

2 Data

The dependent variable in this research is the stock market excess return, Y_{t+h}, given by

$$Y_{t+h} = R_{t+h} - R^f_{t+h}, \tag{1}$$

where R_{t+h} is the S&P500 month-end cum dividend index returns from 1970 to 2018 and R^f_{t+h} is the continuously compounded risk-free rate of return in the period 1970-2018, where the monthly treasury bill rate is used for. The logarithmic return R_{t+h} is defined by the difference between the logarithm of the stock market index at the forecast horizon P_{t+h} and the logarithm of the stock market index at time t P_t:

$$R_{t+h} = logP_{t+h} - logP_t. \tag{2}$$

Both variables are obtained from Amit Goyal's website.[1] The data are from the paper by Welch and Goyal (2008), but we use the updated version, because we also need data after 2005.

The explanatory variables that are used in this research are gathered from the FRED-MD database (except for the 12 lags of the stock market excess return). The data set contains 128 macroeconomic, financial and sentiment variables. The sample period is restricted from 1970 to 2018 and the vintage of January 2019 is used, which contains monthly data up to and including December 2018.[2] This means that we have 588 observations. Because 12 lagged values of the dependent variable are also included in the set of predictors, after removing all variables with missing values we eventually have a set of 104 predictors. The variables from the FRED-MD database are transformed according to McCracken and Ng (2016) and the 12 lags of the stock market excess return are transformed as mentioned earlier in this section.

3 Methodology

In this section, a general explanation of the random forest (RF) method is given. After that, the different forms of the targeted random forest (TRF) are explained. Subsequently the forecasting setting is discussed and finally, two methods for comparing the models are explained in detail, namely the Diebold-Mariano (DM) test statistic (Diebold & Mariano, 2002) and the model confidence set (MCS) (Hansen et al., 2011).

In Appendix A, all used R packages are given, together with their specific settings.

3.1 Random Forest

The main machine learning technique in this paper is the method of random forest, introduced by Breiman (2001). Random forests combine the output of multiple decision trees in order to obtain a single result. In our case, the task is related to regression. Therefore, the mean or the average prediction of the individual trees is returned.

[1] https://sites.google.com/view/agoyal145.

[2] https://research.stlouisfed.org/econ/mccracken/fred-databases/. 2019-01.csv file.

Random forests work according to the following steps:

1. Create a bootstrapped data set from the original data set. To create a bootstrapped data set that is just the same size as the original, we randomly select samples from the original data set. It is allowed to pick the same sample more than once.

2. Create a decision tree using the bootstrapped data set, but only use a random subset of variables at each step. The number of randomly sampled potential splitting variables is often referred as mtry.

3. Repeat steps 1 and 2 a couple of 100 times.

4. If you now have all the measurements, you can run the first tree that you have made and observe the outcome. Do this for all the trees and count which outcome you have the most. This process of bootstrapping the data plus using the aggregate to make a decision is called bagging.

5. To know whether the random forest is good or not, we look at the samples in the original data set that we did not include in the bootstrapped data set. This is typically about 1/3 of the data. This is called the out-of-bag data set. Run this out-of-bag sample on all trees and see if it classifies correctly. The proportion of out-of-bag samples that were correctly classified measures the accuracy of the random forest. It is called the out-of-bag error.

3.2 Targeted Random Forest

As mentioned in the introduction, targeted random forests work the same as ordinary random forests, but they have an initial targeting step. This means that in step 2 of Section 3.1, the set of variables to choose from is smaller. In the initial targeting step, the most important variables are chosen. For this task, many methods can be used. For example, Borup et al. (2023) made use of LASSO regression. We will additionally make use of selecting variables with permutation feature importance, variance of the responses and a package called VSURF. In the following sections, those targeting techniques will be discussed.

3.2.1 LASSO regression

Least Absolute Shrinkage and Selection Operator (LASSO) regression is a technique introduced by Tibshirani (1996). This regression technique finds a balance between model simplicity and accuracy. With a penalty term λ and the l_1 norm of β, it can shrink unimportant coefficients exactly to zero, which is very useful in feature selection, because it automatically identifies and discards irrelevant variables. Therefore, in the paper by Borup et al. (2023), LASSO regression is used in the targeting step before the estimation with RF. The LASSO estimator $\hat{\beta}^{\lambda}$ of the linear regression coefficients is obtained as

$$(\hat{\alpha}^{\lambda}, \hat{\beta}^{\lambda}) = \arg\min_{\alpha,\beta} \sum_{i=1}^{n} (Y_i - \alpha - \beta' X_i)^2 + \lambda \|\beta\|_{l1}, \tag{3}$$

where $\|\beta\|_{l1}$ is the l_1 norm. The choice of λ will control the amount of times the entries of $\hat{\beta}^{\lambda}$ will be zero. The predictors i are only chosen if $\hat{\beta}_i^{\lambda} \neq 0$. Therefore, the choice of λ controls the degree of targeting, s'.

3.2.2 Permutation importance

Similar to random forests, the method of permutation importance was also introduced by Breiman (2001). In recent years, this method has gained popularity as a feature importance metric, because it is relatively easy to understand and it is computationally efficient. Therefore this method can also be used as a targeting technique, before applying the random forest in order to forecast the stock market excess return.

Permutation importance is based on the idea of the permutation of some values of a feature and the observation if the forecast error stays the same, or if it increases. If the forecast error after the permutation stayed the same, it is clear that the particular feature was not very important, but if the forecast error increased, the feature is very likely to be important. In short, permutation feature importance is calculated according to the following steps:

1. Calculate the forecast error of the original model.
2. Generate a permuted feature matrix by permuting one feature each time.
3. For each permuted feature matrix, calculate the forecast error based on the predictions of the new matrix.
4. After that, calculate the permutation feature importance (FI) as a quotient or a difference. For example,
$$FI_j = error_{perm}/error_{orig} \tag{4}$$
or
$$FI_j = error_{perm} - error_{orig}, \tag{5}$$
where FI_j is the permutation feature importance of feature j, $error_{perm}$ is the forecast error after permuting a particular feature and $error_{orig}$ is the forecast error of the original model.
5. Sort those features by descending permutation feature importance.

3.2.3 Variance of the responses

When setting 'importance=impurity' in the 'ranger' R package, the variable importance is computed based on impurity-based measures, in this case the decrease in mean squared error (MSE). These measures assess the extent to which each variable contributes to reducing the impurity or variability within the random forest model. Variables that lead to a significant reduction in impurity are considered more important.

The difference between variable importance based on permutation importance and variable importance based on variance of the responses, is that the technique based on variance of the responses focuses on impurity-based metrics to assess variable importance, while variable

importance based on permutation importance relies on the permutation of variable values to evaluate their impact on model performance.

By minimizing the impurity or variance of the responses, the random forest algorithm aims to create nodes that are as pure and homogeneous as possible, resulting in better predictions and more reliable variable importance assessments.

3.2.4 VSURF

After their paper in 2010 (Genuer et al., 2010), the same authors came with a sequel in 2015 (Genuer et al., 2015). They wrote the second paper in order to describe the R package called VSURF and they also illustrated it on real data sets.

Genuer et al. (2015) distinguish two steps. The first step is to rank the variables according to a variable importance measure. Based on this variable importance measure, the most unimportant variables are eliminated. After that, two different subsets of variables are obtained by the consideration of nested RF models either by the selection of the most accurate variables, or by the sequential introduction of the sorted variables. For the application of this paper, only step 1 is relevant, because we only need the 5, 10, 20, 30 or 50 most important variables to eventually make predictions with those variables. So the obtained rank of the variables in the first step is only of importance.

The definition of the RF variable importance in the paper is as follows. Every tree t in the random forest is constructed by a bootstrap sample. That means that every t also has an Out-Of-Bag (OOB) sample, data that are not included in the bootstrap sample. This data sample is denoted by OOB_t. The error of a tree t, denoted by $errOOB_t$, is the MSE on this sample. After the computation of $errOOB_t$, the values of the predictors j are randomly permuted, to get a sample $\widetilde{OOB}_t^j$. After that, $err\widetilde{OOB}_t^j$ is calculated, which is the error of predictor t on sample $\widetilde{OOB}_t^j$. The variable importance (VI) of predictor j is then calculated as follows:

$$VI^j = \frac{1}{ntree} \sum_t (err\widetilde{OOB}_t^j - errOOB_t), \tag{6}$$

where *ntree* is the number of trees in the RF. A compelling justification for employing this particular approach to variable selection, which combines a classical stepwise method with the utilization of a VI measure, stems from the notion that a variable absent from the true underlying model holds a theoretical importance of null significance.

When the ranking according to the preceding steps is obtained, the first few variables are selected and with them, the predictions with the RF can be made.

3.3 Forecasting setting

The general prediction model is

$$Y_{t+h} = g(Y_t, ..., Y_{t-11}, X_t) + \epsilon_{t+h}, \tag{7}$$

with Y_{t+h} the h-step-ahead forecast, Y_t the 588 x 1 vector of the dependent variable at time t and X_t the 588 x 104 matrix of independent variables at time t. We have namely 588 observations

and 104 predictors, see Section 2. We adopt an expanding window scheme as it allows for the utilization of progressively greater amounts of information in each iteration. This approach is preferred due to the potential dependence of forecasts on historical events from an extended time frame. The initial window spans 15 years, so the out-of-sample period begins in January 1985 and becomes smaller after every iteration.

3.4 Comparing the models

In this section, two ways of the comparison of different models are explained. The main goal of this paper is to investigate whether there is a significant better method than an ordinary RF and which method this is. Therefore it is very important to statistically compare the proposed methods with the already existing method, in order to get meaningful results. The Diebold-Mariano test and the model confidence set will be used and they will be discussed in the next sections.

An important thing to note is that both methods rely on a particular loss function. In this paper, for both methods the Mean Squared Error (MSE) is used, because this error metric gives more attention to outliers than other error metrics and because the MSE squares the differences of the predictions, it does not matter whether the forecast is higher or lower than the real value. The MSE is calculated as follows:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y_i})^2, \tag{8}$$

where $\hat{y}$ is the forecast of y and n is the number of observations. In our case, n = 408, because we have data from January 1970 and we make predictions from January 1986 to December 2018, which leaves 408 months to predict.

For the sake of completeness, in Section 4 the results based on mean absolute error (MAE) are also presented. The MAE is calculated as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y_i}|, \tag{9}$$

where $\hat{y}$ is the forecast of y and n is the number of observations.

3.4.1 Diebold-Mariano test

The Diebold-Mariano (DM) test statistic is proposed by Diebold and Mariano (2002) and it is used to determine whether two models perform significantly different from each other. The DM-statistic is mainly based on the difference in forecast errors of two models. Therefore, the loss differential d_t is calculated as

$$d_t = e_{it}^2 - e_{jt}^2, \tag{10}$$

where e_{it} is the forecast error of model i at time t and e_{jt} is the forecast error of model j at time t. As is clear from Equation 10, we have a squared loss function. For this application, the two compared models will be the ordinary RF and the TRF based on different targeting techniques.

The Diebold-Mariano statistic is given by:

$$DM = \frac{\bar{d}}{\sqrt{V(\hat{d}_{t+1})/P}}, \tag{11}$$

where $\bar{d}$ is the sample mean of d_{t+1} and $V(\hat{d}_{t+1})$ is an estimate of the variance of d_{t+1}. The variance of d_{t+1} can be computed as follows:

$$V(\hat{d}_{t+1}) = \frac{1}{P-1} \sum_{t=T}^{T+P-1} (d_{t+1} - \bar{d})^2, \tag{12}$$

where P is the amount of one-step-ahead forecasts.

The null hypothesis for this test is that both forecast models have equal forecasting performance; $\mathrm{E}[d_t] = 0$. In this paper, the alternative hypothesis is one-sided, namely that the TRF outperforms the ordinary RF.

3.4.2 Model confidence set

The model confidence set (MCS) is an approach to select the best model(s), given a confidence level, proposed by Hansen et al. (2011). It reduces the set of models to a smaller set, which includes the best forecasting model(s). This set is called the MCS. It is possible that there is more than one model present in the MCS. The "best" model is defined in terms of a specific criterion, which can be for example a squared or absolute loss function. In this paper, the main loss function is the squared loss, because that error metric gives the most attention to prominent errors.

After the calculation of the losses for every model and every amount of predictors selected (only for the ordinary RF, this amount is always equal to the total amount of variables), the p-values for the null hypothesis of 'no inferior model is present' can be calculated. If the p-value of a specific model is less than or equal to a specific significance level α, the null hypothesis is rejected and the model is not included in the MCS. Likewise, if the p-value of a specific model is greater than or equal to a specific significance level, the null hypothesis is not rejected and the model is included in the MCS. In Section 4, all p-values will be given, also the ones for the models who are eventually not included in the MCS.

4 Results

In Table 1, the ratios of the mean squared prediction error (MSE) of different versions of TRF to that of ordinary RF are given. Each column represents a TRF model with a given targeting technique. All ratios below unity indicate that the TRF performed better than the ordinary RF. *, ** and *** indicate statistical significance, based on a one-sided Diebold-Mariano test statistic, at levels 10%, 5% and 1% respectively. The lowest values per amount of selected predictors in the table, i.e. the biggest differences between the ordinary RF and the TRF, are in bold.

Table 1: Predictive ability of ordinary RF versus different versions of TRF using shallow trees for a forecast horizon of 1 and MSE loss function.

s'	**LASSO**	**Permutation importance**	**Variance of the responses**	**VSURF**
5	**0.699****	0.798**	0.819*	0.724**
10	0.816**	0.815**	0.812**	**0.774****
20	0.909***	0.866**	0.865**	**0.851****
30	0.962	0.905**	0.905**	**0.891****
50	0.982	0.939**	0.943*	**0.927*****

Each row in Table 1 represents a specific amount of predictors used in the TRF (s'). s' = 5 represents a setting where almost all variables are discarded, but only the few most important ones are kept. s' = 10, 20 or 30 represents a medium-dimensional setting where most of the variables are left out, but where a significant number is still retained. Finally, s' = 50 represents a setting where approximately half of the predictors are discarded and half of the predictors are kept. All models in different settings are compared with the ordinary RF, which uses the total amount of predictors, namely 104. For the TRF based on LASSO regression, the parameter λ is tuned to target 5, 10, 20, 30 or 50 predictors.

It is clear that for all TRF techniques in all settings, the MSE value is less than the MSE value of the ordinary RF, indicating that TRF outperforms the ordinary RF in all settings. For most of the models, the improvement of TRF against RF is statistically significant at a level of 5 % and for LASSO (s' = 20) and VSURF (s' = 50), this difference is even statistically significant at a level of 1 %. Only for TRF based on LASSO regression, the difference in forecasting performance is not statistically significant when 30 or 50 predictors are selected, which is not in line with the paper from Borup et al. (2023), who concluded for these very values that the difference was statistically significant.

An interesting thing to see is that the absolute performance of all models decreases when the amount of selected predictors increases (except for the step from 5 to 10 predictors for TRF based on variance of the responses). This suggests that the less predictors are selected, the better the model performance is in terms of MSE. The reason for this can be that for this application, only a few variables have a very big influence on the dependent variable, in our case the stock market excess return. In that sense, the more variables are included to forecast the dependent variable, the more 'noise' is added to the model and therefore the forecast performance decreases.

The last and most important observation is that the TRF based on the VSURF package performs the best in terms of absolute MSE in all settings, except for the sparse setting where only 5 predictors are included. In Table 2, this result is confirmed.

In this table, the p-values of the model confidence set (MCS) approach are given. The p-values assume that the null hypothesis of 'no inferior model present' in the confidence set is true. Low p-values reject this hypothesis and high p-values do not reject this hypothesis. Therefore, models with p-values below a certain confidence level are not included in the MCS, which indicates that these models are certainly not the best models, according to a squared loss function in our case. From Table 2 it becomes clear that the ordinary RF is never present in the MCS, because the p-value is 0.0000 in all settings. This supports the conclusion that targeting

predictors in a random forest regression increases the model performance (Borup et al., 2023).

To confirm the earlier statement that the TRF based on the VSURF package performs the best in terms of absolute MSE in all settings, except for the sparse setting where only 5 predictors are included, we can look at the reported p-values for this TRF technique. For s' = 10, 20, 30 and 50, this TRF technique has a p-value of 1.0000, indicating that this model will always be included in the MCS, for every value of α, which clearly indicates that this model is the best performing model in forecasting the stock market excess return. In Appendix B, MSE ratios of TRF based on a targeting technique other than LASSO versus TRF based on LASSO are given. Bold indicates that the TRF other than LASSO performed better than the TRF based on LASSO, indicating that it is a better idea to use a targeting technique other than LASSO.

Table 2: P-values of MCS approach.

s'	**RF**	**LASSO**	**Permutation importance**	**Variance of the responses**	**VSURF**
5	0.0000	1.0000	0.0000	0.0000	0.0358
10	0.0000	0.2806	0.2594	0.2806	1.0000
20	0.0000	0.0124	0.2566	0.2566	1.0000
30	0.0000	0.0070	0.1440	0.1440	1.0000
50	0.0000	0.0014	0.1554	0.1190	1.0000

For the sake of completeness, in Table 3 the ratios of the mean absolute prediction error (MAE) of different versions of TRF to that of ordinary RF are given. Again *, ** and *** indicate statistical significance, based on a one-sided Diebold-Mariano test statistic, at levels 10%, 5% and 1% respectively and the lowest values per amount of selected predictors in the table, i.e. the biggest differences between the ordinary RF and the TRF, are in bold. The results based on MSE values carry over to the case based on MAE values, because in any case in Table 3, TRF outperforms ordinary RF and just like in Table 1, TRF based on LASSO performs the best when estimating with the 5 most important predictors and TRF based on the VSURF package performs the best when estimating with the 10, 20, 30 and 50 most important predictors. A thing that stands out is that when an absolute loss function is used, almost all differences with an ordinary RF are statistical significant at a 1% level, giving even stronger support for using TRF over RF.

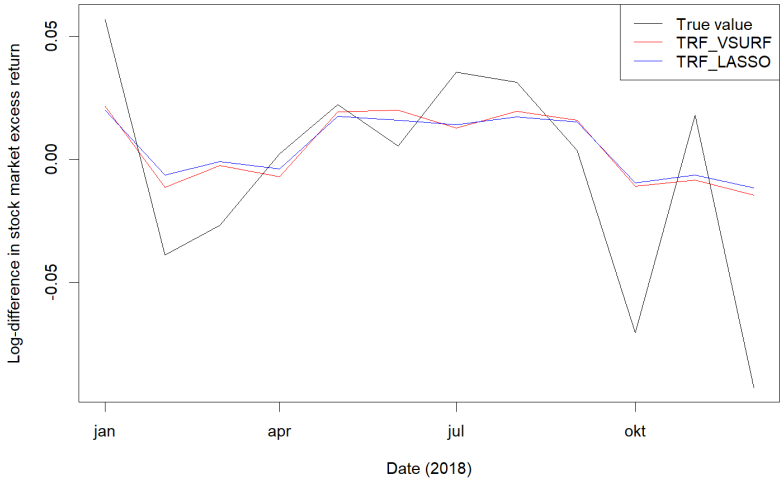
Table 3: Predictive ability of ordinary RF versus different versions of TRF using shallow trees for a forecast horizon of 1 and MAE loss function.

s'	**LASSO**	**Permutation importance**	**Variance of the responses**	**VSURF**
5	**0.845*** **	0.990***	0.906***	0.853***
10	0.898***	0.904***	0.902***	**0.883*** **
20	0.948***	0.927***	0.927***	**0.924*** **
30	0.976***	0.949***	**0.947*** **	**0.947*** **
50	0.990**	0.967***	0.967***	**0.961*** **

Finally, we compare the model that we conclude is the best model, TRF based on the VSURF package, with the model proposed by Borup et al. (2023), namely TRF based on LASSO regression, using a figure. Figure 1 shows the true value of the stock market excess return in the

year 2018 alongside the predicted values of the TRF model based on VSURF and the TRF model based on LASSO for the same data. Both TRF models select 50 predictors here. Appendix C shows the same figures, but for 5, 10, 20 and 30 predictors. The year 2018 was chosen, because it is the most recent year in our data set and it is therefore most important to predict well in this year, compared to earlier years. In Figure 1, we see that in the months of February, March and May, the line of the TRF model based on VSURF is the closest to the line of the true values. This is also clearly the case for the months of August, September and October. In the months when this is less obvious, LASSO is also not clearly better. Moreover, the peaks are better represented by the red line than the blue line, implying that TRF based on the VSURF package performs better in that area too. This confirms our conclusion, which will be clearly stated in the next section.

Figure 1: Forecasting performance of TRF based on VSURF package against forecasting performance of TRF based on LASSO, both with 50 selected predictors.



5 Conclusion

This paper is an examination whether targeting predictors in a step prior to the estimation by a random forest improves the forecasting performance and which of the four discussed targeting techniques make the most accurate predictions. The discussed targeting techniques are LASSO regression, permutation feature importance, feature importance based on variance of the responses and feature importance based on the VSURF R package from Genuer et al. (2015). All predictions are made for a forecast horizon of one. Based on MSE values, from the results it becomes clear that all targeted random forests make more accurate predictions than a random forest with all initial independent variables included. Moreover, the TRF based on the VSURF package turns out to be the best model in forecasting the out-of-sample stock market excess return. Another interesting result is that the higher the degree of targeting is, the better the forecasting performance for all models is, indicating that a RF with only a few variables selected is preferred.

In conclusion, adding a targeting step prior to the estimation of a random forest is regarded

as a favorable and sensible notion. Moreover, Borup et al. (2023) compared forecasts of the ordinary RF with forecasts made by using an autoregressive (AR) model with 12 lags, in order to assess the absolute level of predictability exhibited by the original RF. They find that the RF outperforms the AR(12) model across all horizons and all settings, which indicates that all improvements outlined in this paper can be regarded as advancements over an already superior forecasting model. This implies that in the future, when out-of-sample forecasts of stock market excess returns (and maybe also other economic variables) need to be made, targeted random forests with the VSURF package as a targeting technique can be properly applied.

In future research, different targeting techniques can be investigated, such as feature importance based on Shapley values (Lundberg & Lee, 2017). Also other macroeconomic variables can be investigated. Something else that might be interesting to explore is the possibility of getting better model performance if irrelevant or noise variables are added correctly, stated by Mentch and Zhou (2022), instead of removing the irrelevant variables in the step before the estimation by the RF like is done in this paper. They state that adding extra random noise features to the model can improve the out-of-sample predictive accuracy over even the most optimally tuned model on the original design. We would expect that adding random noisy variables would make a model only worse, but the opposite seems to be true. It seems very interesting to investigate if this is indeed the case and especially in what situations it would be beneficial to add noise variables and in what situations it would be beneficial to remove them.

A limitation of this research is that only one-step-ahead forecasts are considered. Galbraith and Tkacz (2007) conclude that forecasting macroeconomic variables for longer forecast horizons can be very hard, so maybe when considering a longer forecast horizon, the results would be different. Another limitation of this research is that the results obtained for the TRF based on LASSO regression were not the same as in the paper by Borup et al. (2023). A reason for this can be that the data sets were not the same, because in that paper not all details needed for exact replication were given; they claim to have obtained 100 predictors subsequent to the data cleansing process, but they do not mention how they arrived at that number of predictors. In this article, the data set actually contained 104 predictors. It is also not clear which vintage they used, so that could be another reason for the difference in the results.

6 Appendix

A Programming information

Attached is the R code. The data is collected in the downloadingdata.R file. The required packages are **fbi**, **tidyverse**, **TTR** and **readxl**. All ordinary RF estimations are made in the ordinaryRF.R file, using the **ranger** package. The number of trees is set to 500 and the maximum tree depth is set to 3. The TRF estimations with LASSO as the targeting technique are made in the TRF_LASSO.R file and the **ranger** package and **glmnet** package are used. In order to specifically implement the LASSO regression, α had to be set equal to 1. TRF based on permutation importance and variance of the responses can be find in the files TRF_PI.R and TRF_VOR.R respectively. In both files, the **ranger** package is used. For TRF based on the VSURF package, of course the **VSURF** package is used, together with the **ranger** and **parallel** packages. The code can be find in the TRF_VSURF.R file. To decrease the computation time, the RFimplem parameter is set to "ranger", clustertype is set to "ranger" and parallel is set equal to "TRUE". For the DM-test, the **forecast** package is loaded where the alternative parameter is set to "greater", h is set to 1, power is set to 2 and varestimator is set to "acf". The code can be found in the DMtest.R file. Finally, for the calculation of the MCS, the estMCS.quick function of the **modelconf** package is used, with B = 5000 and a "t.max" test. Also the **devtools** package is required. For the code, see the MCS.R file.

B Descriptive table

Table 4: Predictive ability of TRF based on LASSO versus different versions of TRF using shallow trees for a forecast horizon of 1 and MSE loss function. The numbers are the ratios of MSE of different versions of TRF to that of TRF using LASSO regression. Bold indicates that the TRF model based on another targeting technique than LASSO performed better.

s'	**Permutation importance**	**Variance of the responses**	**VSURF**
5	1.142	1.172	1.037
10	**0.998**	**0.995**	**0.948**
20	**0.952**	**0.951**	**0.936**
30	**0.941**	**0.941**	**0.926**
50	**0.956**	**0.961**	**0.944**

C Additional figures

Figure 2: Forecasting performance of TRF based on VSURF package against forecasting performance of TRF based on LASSO, both with 5 selected predictors.

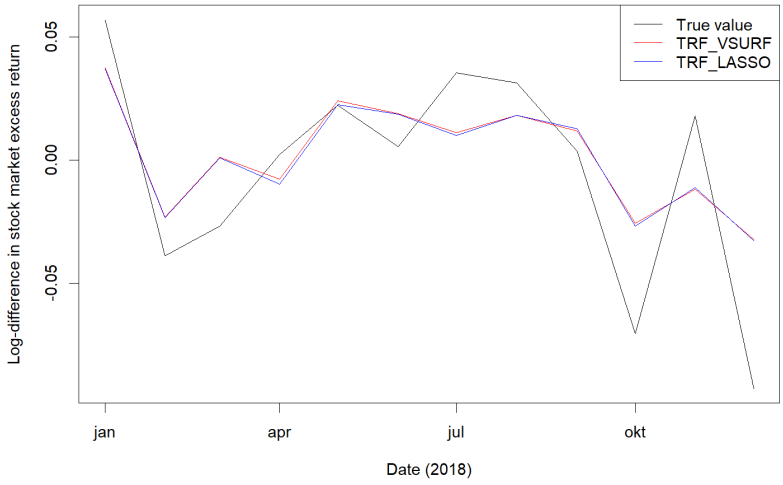


Figure 3: Forecasting performance of TRF based on VSURF package against forecasting performance of TRF based on LASSO, both with 10 selected predictors.

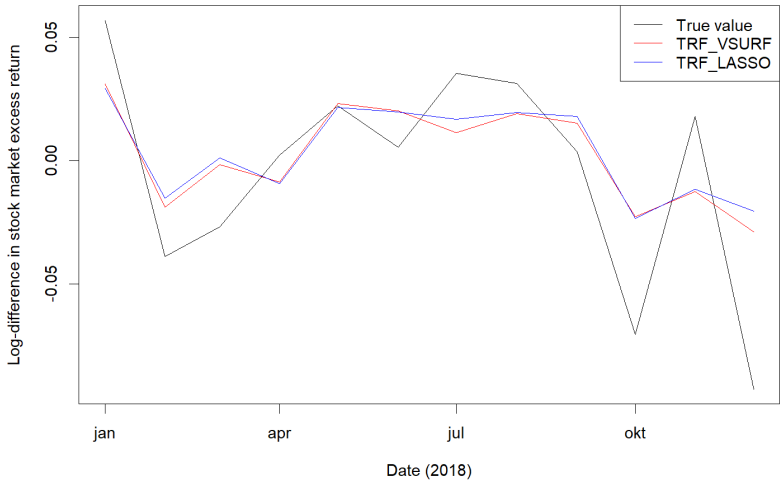


Figure 4: Forecasting performance of TRF based on VSURF package against forecasting performance of TRF based on LASSO, both with 20 selected predictors.

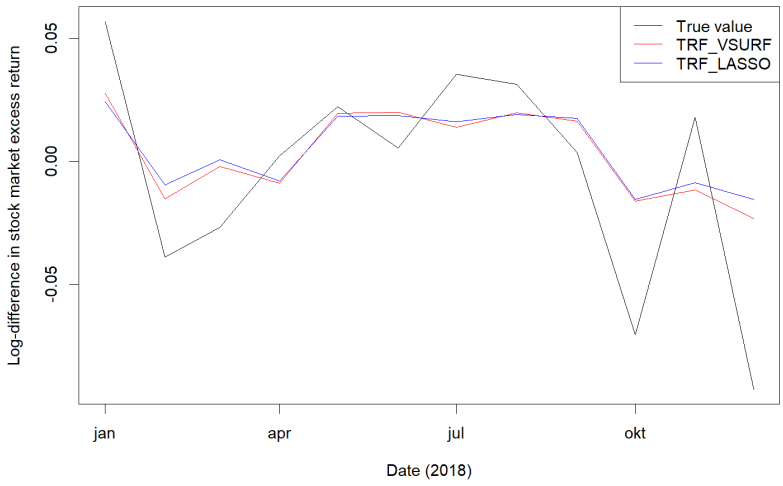
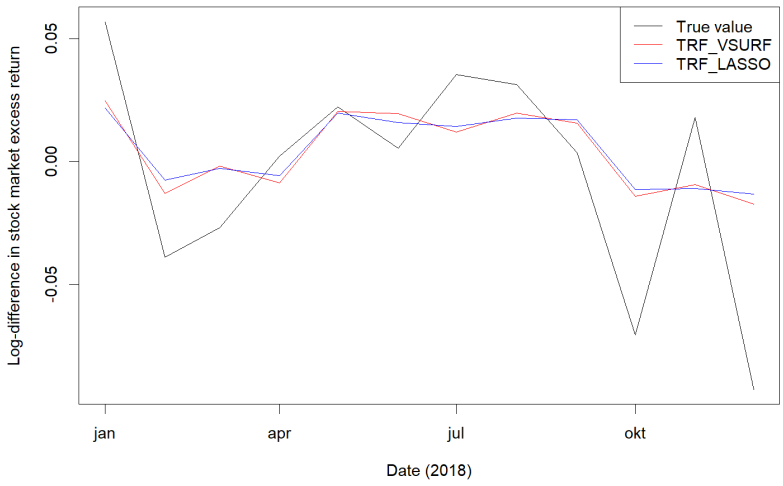


Figure 5: Forecasting performance of TRF based on VSURF package against forecasting performance of TRF based on LASSO, both with 30 selected predictors.



References

Bai, J. & Ng, S. (2008). Forecasting economic time series using targeted predictors. *Journal of Econometrics*, *146*(2), 304–317.

Borup, D., Christensen, B. J., Mühlbach, N. S. & Nielsen, M. S. (2023). Targeting predictors in random forest regression. *International Journal of Forecasting*, *39*(2), 841–868.

Borup, D. & Schütte, E. C. M. (2022). In search of a job: Forecasting employment growth using google trends. *Journal of Business & Economic Statistics*, *40*(1), 186–200.

Breiman, L. (1996). Out-of-bag estimation.

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

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

Charles, A. & Darné, O. (2022). Forecasting macroeconomic time series using sparse random forest models. *Available at SSRN 4111995*.

De Ville, B. (2013). Decision trees. *Wiley Interdisciplinary Reviews: Computational Statistics*, *5*(6), 448–455.

Diebold, F. X. & Mariano, R. S. (2002). Comparing predictive accuracy. *Journal of Business & economic statistics*, *20*(1), 134–144.

Dietterich, T. G. (2000). Ensemble methods in machine learning. In *Multiple classifier systems: First international workshop, mcs 2000 cagliari, italy, june 21–23, 2000 proceedings 1* (pp. 1–15).

Fan, J. & Lv, J. (2008). Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *70*(5), 849–911.

Fortin-Gagnon, O., Leroux, M., Stevanovic, D. & Surprenant, S. (2022). A large canadian database for macroeconomic analysis. *Canadian Journal of Economics/Revue canadienne d'économique*, *55*(4), 1799–1833.

Galbraith, J. W. & Tkacz, G. (2007). Forecast content and content horizons for some important macroeconomic time series. *Canadian Journal of Economics/Revue canadienne d'économique*, *40*(3), 935–953.

Genuer, R., Poggi, J.-M. & Tuleau-Malot, C. (2010). Variable selection using random forests. *Pattern recognition letters*, *31*(14), 2225–2236.

Genuer, R., Poggi, J.-M. & Tuleau-Malot, C. (2015). Vsurf: an r package for variable selection using random forests. *The R Journal*, *7*(2), 19–33.

Hansen, P. R., Lunde, A. & Nason, J. M. (2011). The model confidence set. *Econometrica*, *79*(2), 453–497.

Lundberg, S. M. & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, *30*.

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

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

Mentch, L. & Zhou, S. (2022). Getting better from worse: Augmented bagging and a cautionary tale of variable importance. *Journal of Machine Learning Research*, *23*(224), 1–32.

Rapach, D. E. & Zhou, G. (2020). Time-series and cross-sectional stock return forecasting: New machine learning methods. *Machine learning for asset management: New developments and financial applications*, 1–33.

Tan, Z., Yan, Z. & Zhu, G. (2019). Stock selection with random forest: An exploitation of excess return in the chinese stock market. *Heliyon*, *5*(8), e02310.

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

Tyralis, H. & Papacharalampous, G. (2017). Variable selection in time series forecasting using random forests. *Algorithms*, *10*(4), 114.

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

Zou, H. & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, *67*(2), 301–320.