Predicting Stock Returns with Machine Learning

Mykola Pinchuk*

20 February 2022

Abstract

This paper examines performance of modern machine learning models for the task of predicting expected returns of the cross-section of stocks. I consider 10 linear and nonlinear models. Traditionally used linear regression underperforms modern machine learning models, achieving only 0.36% R^2 out of sample. Artificial neural networks and tree-based methods have the best performance, reaching 1.9% and 1.1% R^2 respectively.

*Simon Business School, University of Rochester. Email: Mykola.Pinchuk@simon.rochester.edu.

1 Introduction

Predicting which stocks will have higher returns is probably one of the most common problems in finance industry. Existence of active asset management industry to some extent relies on the hope that asset managers can identify and invest in stock with higher expected returns. No wonder that predicting expected returns is one of the the most thoroughly-studied problems in academic finance research. Empirical research in asset pricing has long history of building predictive models for cross-section of expected excess stock returns. Since early 1980s, this research attempted to predict and explain cross-section of expected stock returns by characteristics of these firms.

Until recently, all academic and most industry research on predicting returns in cross-section of stocks relied on ordinary least squares (OLS) regression. While its simplicity facilitates interpretation, the rigidity of linear model constrains its ability to fit complex relations. In particular, linear model struggles in detecting nonlinear patterns as well as interaction effects between explanatory variables. There is trade-off between explanatory and predictive power of models of varying complexity. Thus it is important to consider wide range of predictive models to understand how much predictive power we can get by giving up some explanatory power.

In this paper I study predictive performance of 10 machine learning model in predicting the cross-section of expected returns. I consider both linear and nonlinear models. Apart of generalized linear models, I use KNN, SVM, tree-based models and neural networks. I document that neural network model dramatically outperforms linear models, having 5.5 times better predictive performance on test sample. Overall, for tasks valuing predictive performance over interpretability linear regression should not be considered as baseline model due to poor predictive performance.

This paper contributes to large body of research in both finance academia and industry. Since seminal paper by Fama and French (1992), academic empirical asset pricing focused on relation between returns and characteristics of stocks. Recent

examples of such research are Harvey, Liu and Zhu (2016) and Hou, Xue and Zhang (2020). Over the past 3 years some attempts have been made to employ machine learning models for predicting cross-section of stocks returns. Both academic (Shihao, Kelly and Xiu 2020, Chen, Pelger and Zhu 2020, Bianchi, Buechner and Tamoni 2021) and industry-oriented research (Christian and Jones 2019, Gao et al 2020, Rasekhschaffe) on application of machine learning to finance is emerging as a separate field.

2 Data and summary statistics

This paper uses data on stock returns as well as characteristics of publicly-traded firms and their stock at monthly frequency. I combine these data from standard sources in academic finance: Center of Research in Security Prices (CRSP) and StandardPoor's Compustat. In total, I have more than 3,200,000 observations over 1964-2018. Figure 2 shows that sample size kept increasing before 1990s and somewhat declined after that.

After performing data cleaning, standard for empirical asset pricing research, I have a sample with 1,336,352 firm-month observations. This is the sample used throughout this analysis. I use stock return in current month as target variable and stock characteristics, known before the beginning of the current month, as independent variables (features). The sample includes 17 explanatory variables, listed in Table 1. I further augment the sample with lagged versions of some of predictor variables. The final sample contains 27 features.

Table 2 reports summary statistics of these variables. To decrease tails in the distribution of features, I took natural logs of size and book-to-market. To limit effects of outliers, I use winsorization at 0.5% and 99.5% levels. At Figure 1 shows, distribution of stock returns (target variable) is approximately normal. Figure 3 describes scatterplots between returns and features.

In order to capture some nonlinear relations using linear models, I transform the sample using second-order polynomial transform. This results in 406 variables. To avoid adding noise from less relevant variables, I then use linear PCA and select the first 50 principal components. Finally, to see how much information I lose from the original sample by using PCA, I employ the sample with first 10 principal components. So I repeat all predictive modeling for 3 samples:

- Original sample, 27 features.
- PCA-10 sample, 10 features.
- PCA-50 sample, 50 features.

I select number of principal components to capture at least 70% of variation in the input sample.

3 Research Design

The main purpose of this analysis is to predict return of the given firm over the month t by using information, known at the end of month t-1. Thus I use current month-return as target variable and all known characteristics of the firm and its stock as predictors (features).

This paper uses 10 predictive models:

1. Ordinary linear regression (OLS).
2. LASSO regression.
3. Ridge regression.
4. Elastic Net regression.
5. K-nearest neighbor (KNN) regression.
6. Support Vector Regression (SVR).
7. Simple decision tree.
8. Random Forest.
9. Boosted Tree.
10. Artificial neural network

For each model with hyperparameters, I select them first via 10-fold cross-validation. That is I use a grid of hyperparameters to pick their value, achieving the best cross-validation performance. Since large parameter grid could lead to multiple hypotheses testing problem, where the best hyperparameters will achieve high performance by chance, I use separate test sample to test performance of the model with chosen hyperparameters.

Selection of test sample in the context of panel data is a difficult problem. Ideally, I would like to select test sample over time period immediately after training sample. This would mean using expanding training sample and estimating each model and its hyperparameters dozens of times. Due to limited access to computing power I can not use this approach. Instead I randomly select 100,000 observations from 1990-2018 sample as test sample. For each model, the paper will report cross-validation R^2 of the model with optimal hyperparameters on train sample and then R^2 on test sample using the same model.

I implement this analysis using Python and sklearn, XGBoost and TensorFlow libraries.

4 Modeling results

Table 3 presents main results. 6 columns describe R^2 from 9 predictive models. Every two columns correspond to one sample type: original sample with 27 features, a sample with 10 PCA features and a sample with 50 features. For each sample, I report cross-validated R^2 on train sample first and then report test sample R^2.

Almost all models perform best on the original sample with 27 variables. The only exception are linear models (OLS, LASSO, Ridge and Elastic net), which have slightly better out-of-sample performance on PCA 50 sample than on original sample. Since PCA 50 sample captures nonlinear effect and interactions between original features to some extend, this result seems to imply that poor performance of linear models is

due to its inability to accommodate nonlinearities. Every nonlinear model shows the best performance on the original sample with 27 features.

Different variations of linear model have very similar performance. This is due to the fact that grid search algorithm usually selects very low penalty parameters, which means that LASSO, Ridge and Elastic net models end up being very close to linear regression. They provide 0.34-0.37% R^2 on either train or test samples. As a sidenote, performance of linear model in this panel regression is much weaker than in Fama-MacBeth regressions, standard in empirical asset pricing literature. This is likely to be driven by the fact that Fama-MacBeth regression overweights observations during early part of the sample with few observations. Since stock characteristics had stronger relation to expected returns in earlier part of the sample, implicitly overweighting them allows Fama-MacBeth regression to achieve much higher R^2 than panel regression.

Tree-based methods and neural networks perform best among considered set of models. Even simple tree with depth of 4 slightly outperforms linear models. Random forest achieves 0.86% out-of-sample R^2, which is more than twice as linear models. Finally, boosted tree reaches 1.11% R^2 on test sample, which is three times better than linear models.

Neural network model performs best and achieves 1.91% R^2 on test sample. due to computational limitations, I did not use grid search to pick hyperparameters of neural network, so it may be possible to achieve large improvement in its performance through further tweaking. I used neural network with 5 dense layers and 256x3, 64 and 16 nodes in each layer. The model uses batch normalization and dropout with probability of 0.4 at all layers. I trained the model with early stopping and it usually took 30-50 epochs.

The results imply that modern machine learning models can achieve very large improvement in predictive accuracy over linear regression. This is likely due to

ability of tree-based models and neural networks to effectively capture nonlinearities and interaction effects between the features.

To gain insight into drivers of the performance of each model, I investigate feature importance. Figures 7-13 describe these results. In case of linear models, feature importance is simply coefficient estimate in front of each regressor. Since regularization hyperparameter is usually very small, all linear models have similar pattern of feature importances. Size and its lagged value are the strongest predictors. Consistent with results from academic empirical asset pricing literature, size is very strong negative predictor, implying that small firms have higher returns.

Figures 11-13 present feature importance in tree-based models. While size remains very important feature, short-term reversal (denoted as mom11), traditional momentum (denoted as mom122) and book-to-market emerge as other important features.

5 Conclusion

Large part of research in asset pricing and investment management comes down to predicting cross-section of expected stock returns using stock characteristics. Traditionally this research has relied heavily on linear models, primarily on linear regression. This paper shows that modern machine learning models can achieve more than 400% improvement in predictive performance over linear regression.

Using simple nonlinear transformations of a sample does not lead to significant improvements in the performance of linear models. Deep neural network achieves 1.91% R^2 compared to 0.36% R^2 for linear models due to its ability to capture nonlinearities and interaction effects. Neural network outperforms other nonlinear models such as tree-based approaches, KNN and SVR.

Using complex machine learning models such as neural networks does not come for free. By gaining more predicative power we give up simplicity and interpretability of simple models like linear regression. Thus the main takeaway is to realize that there is a spectrum of statistical models, ranging across their predictive power and interpretability. It is the job of an analyst to pick the model, most suited to particular task. For tasks valuing predictive performance over interpretability linear regression should not be considered as baseline model due to poor predictive performance.

6 References:

1. Bianchi, Daniele, Matthias Büchner, and Andrea Tamoni. "Bond risk premiums with machine learning." The Review of Financial Studies 34, no. 2 (2021): 1046-1089.
2. Chen, Luyang, Markus Pelger, and Jason Zhu. "Deep learning in asset pricing." Available at SSRN 3350138 (2020).
3. Fama, Eugene F., and Kenneth R. French. "The cross-section of expected stock returns." the Journal of Finance 47, no. 2 (1992): 427-465.
4. Gao, Ziming, Yuan Gao, Yi Hu, Zhengyong Jiang, and Jionglong Su. "Application of deep q-network in portfolio management." In 2020 5th IEEE International Conference on Big Data Analytics (ICBDA), pp. 268-275. IEEE, 2020.
5. Géron, Aurélien. Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. " O'Reilly Media, Inc.", 2019.
6. Gu, Shihao, Bryan Kelly, and Dacheng Xiu. "Empirical asset pricing via machine learning." The Review of Financial Studies 33, no. 5 (2020): 2223-2273.
7. Harvey, Campbell R., Yan Liu, and Heqing Zhu. "... and the cross-section of expected returns." The Review of Financial Studies 29, no. 1 (2016): 5-68.
8. Hastie, Trevor, Robert Tibshirani, Jerome H. Friedman, and Jerome H. Friedman. The elements of statistical learning: data mining, inference, and prediction. Vol. 2. New York: springer, 2009.
9. Hou, Kewei, Chen Xue, and Lu Zhang. "Replicating anomalies." The Review of Financial Studies 33, no. 5 (2020): 2019-2133.
10. Kelly, Bryan, Asaf Manela, and Alan Moreira. "Text selection." Journal of Business Economic Statistics 39, no. 4 (2021): 859-879.
11. Rasekhschaffe, Keywan Christian, and Robert C. Jones. "Machine learning for stock selection." Financial Analysts Journal 75, no. 3 (2019): 70-88.

7 Appendix

Figure 1: Histogram of returns

The figure describes distribution of monthly returns in the sample with 1,357,579 observations.

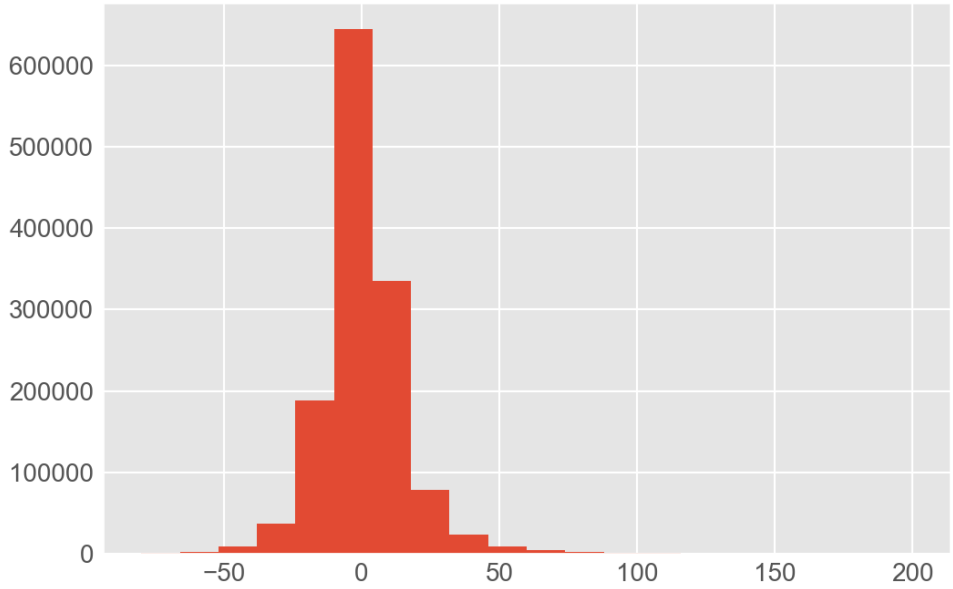

Figure 2: Sample size over time

The figure describes evolution of the number of firms in the sample over time.

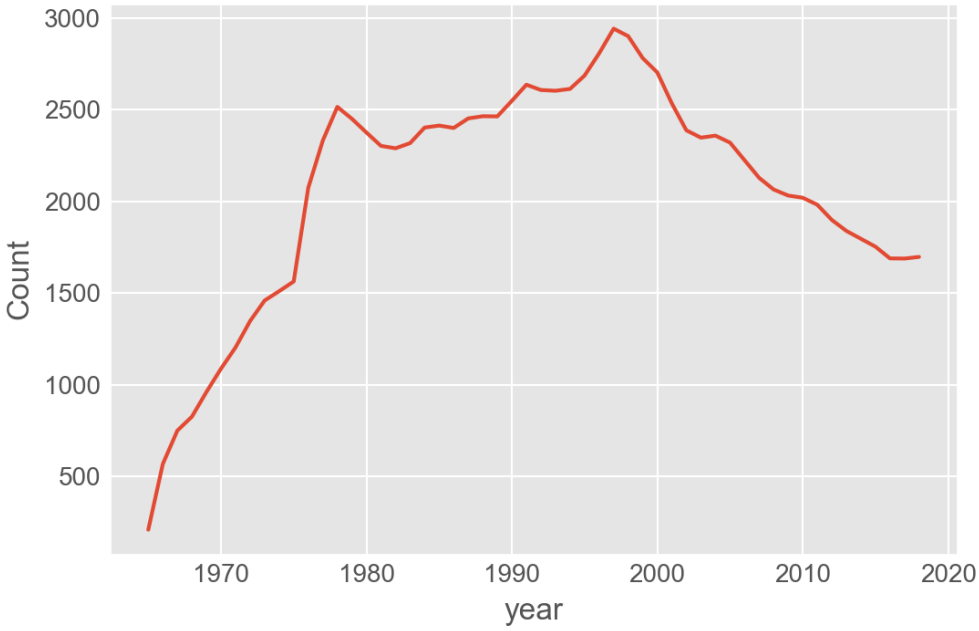

Figure 3: Relation between returns and features

The figures describe joint distribution of returns and features. See Table 1 for feature description.

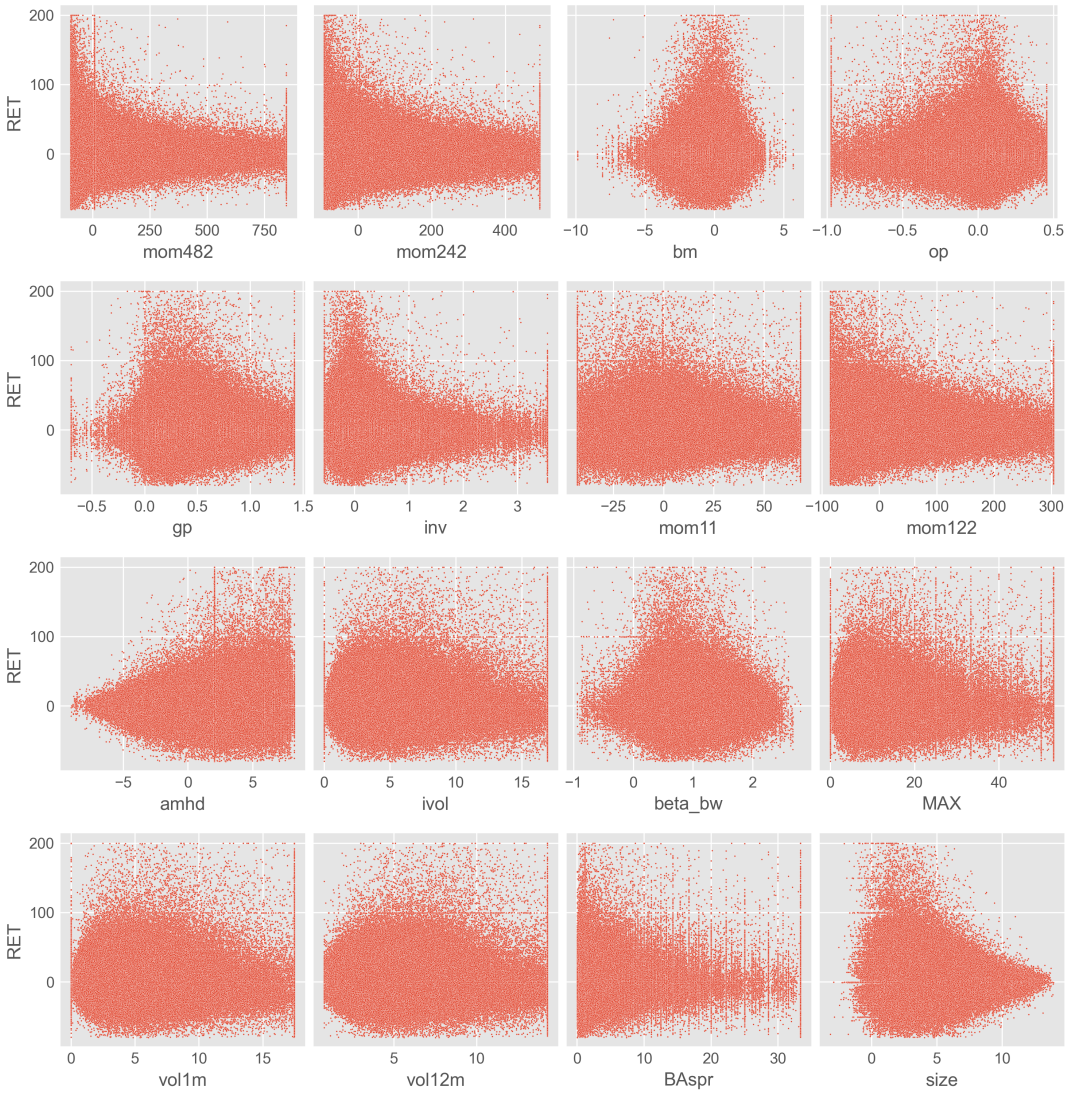

Figure 4: Model performance on original sample

The figures describe R^2 from predictive models on train and test samples. I use 27 features to predict returns.

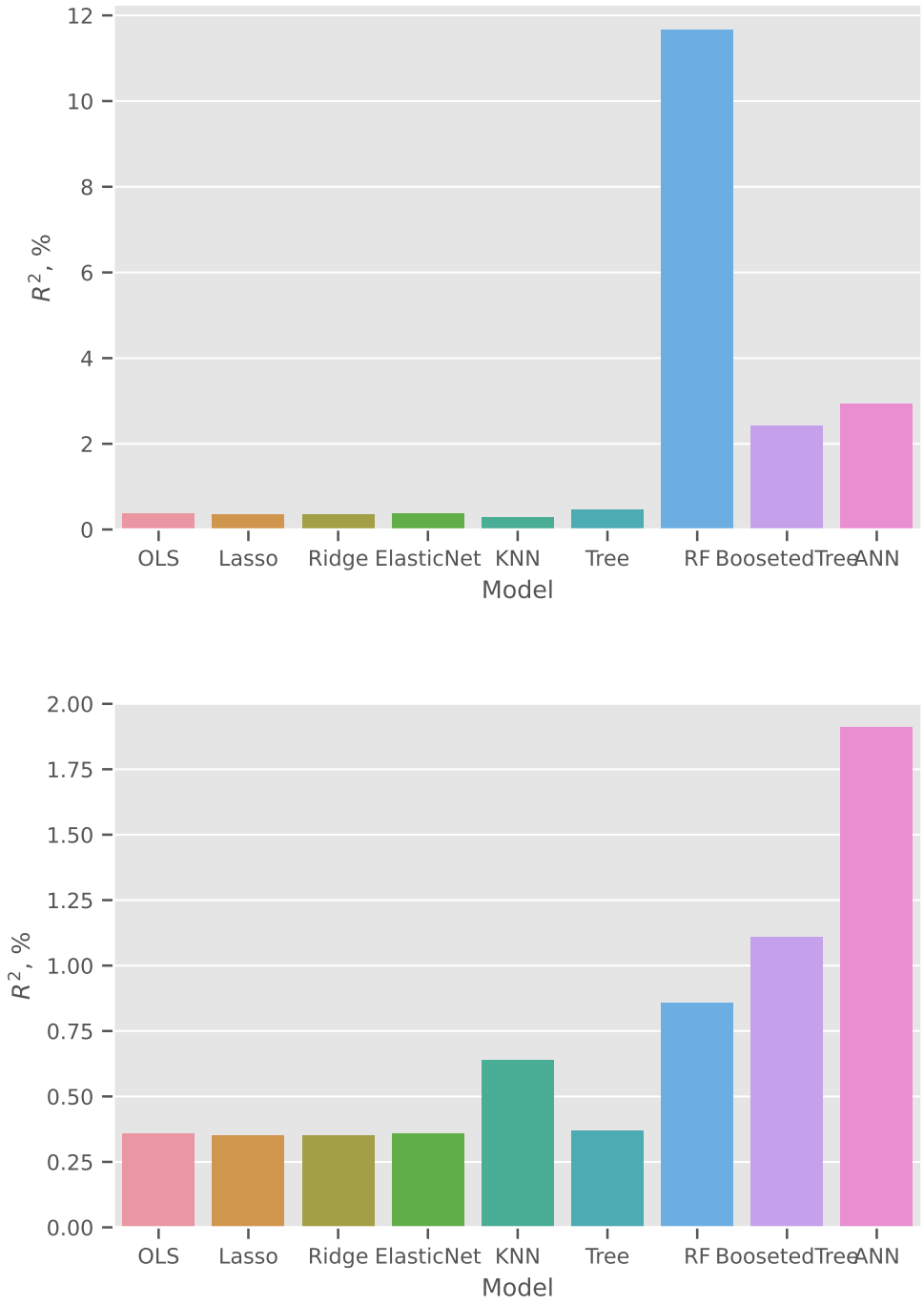

Figure 5: Model performance on PCA10 sample

The figures describe R^2 from predictive models on train and test samples. I use 10 principal components, obtained from the original sample of 27 features to predict returns.

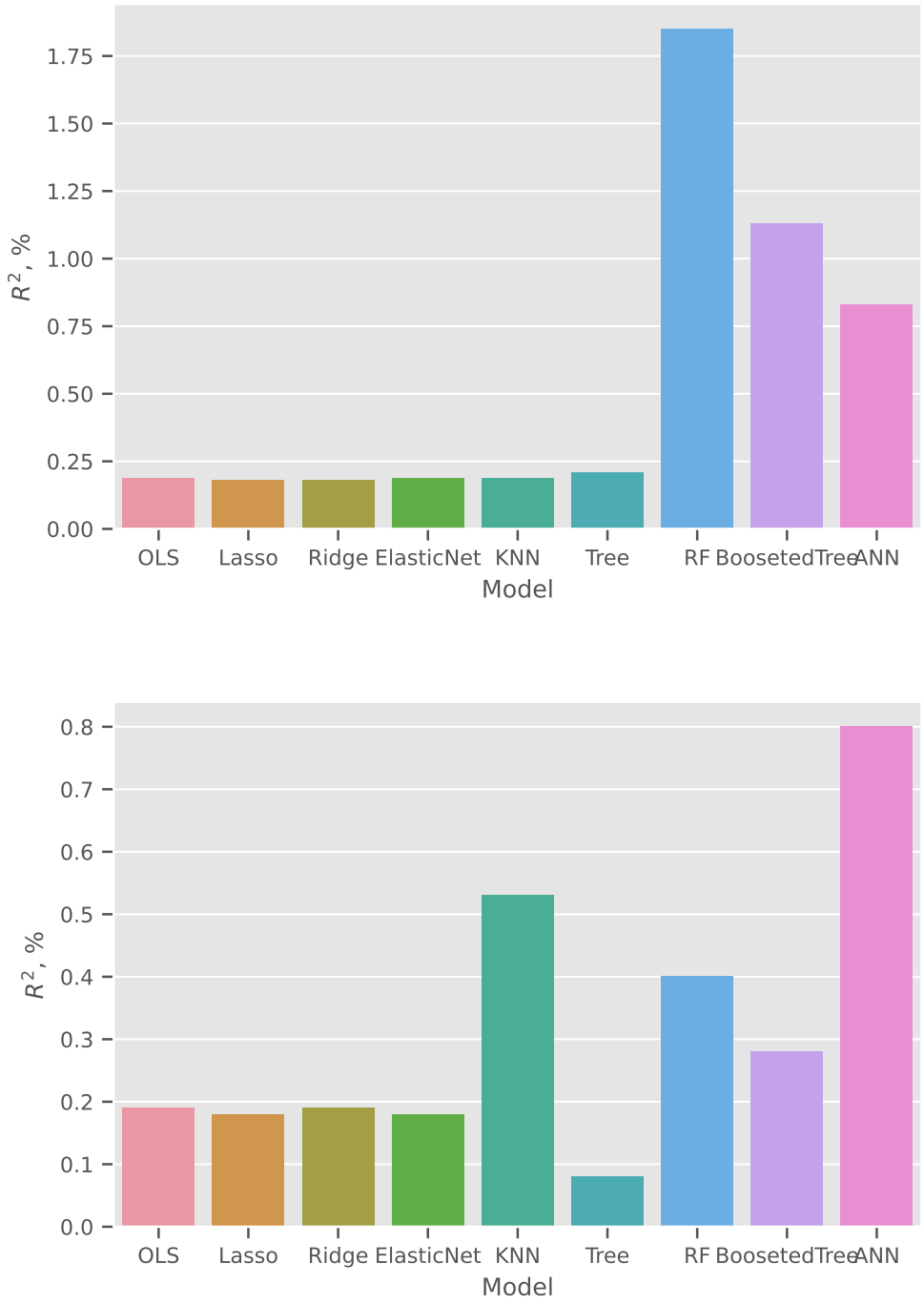

Figure 6: Model performance on PCA50 sample

The figures describe R^2 from predictive models on train and test samples. I use 50 principal components, obtained from polynomial transformation of the original sample of 27 features to predict returns.

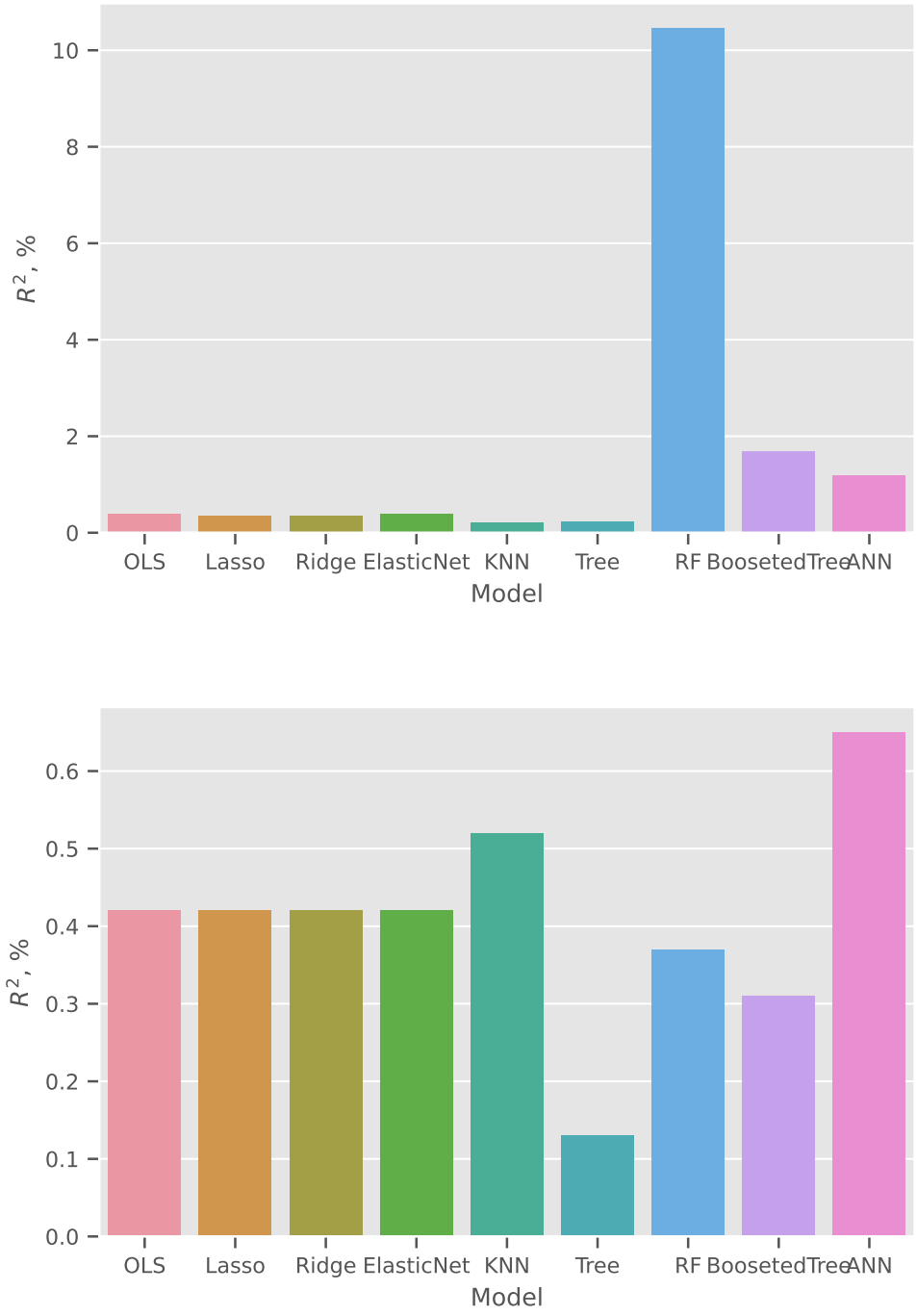

Figure 7: Feature importance in linear regression

The figures describes coefficient estimates for all features in linear regression. See Table 1 for feature description.

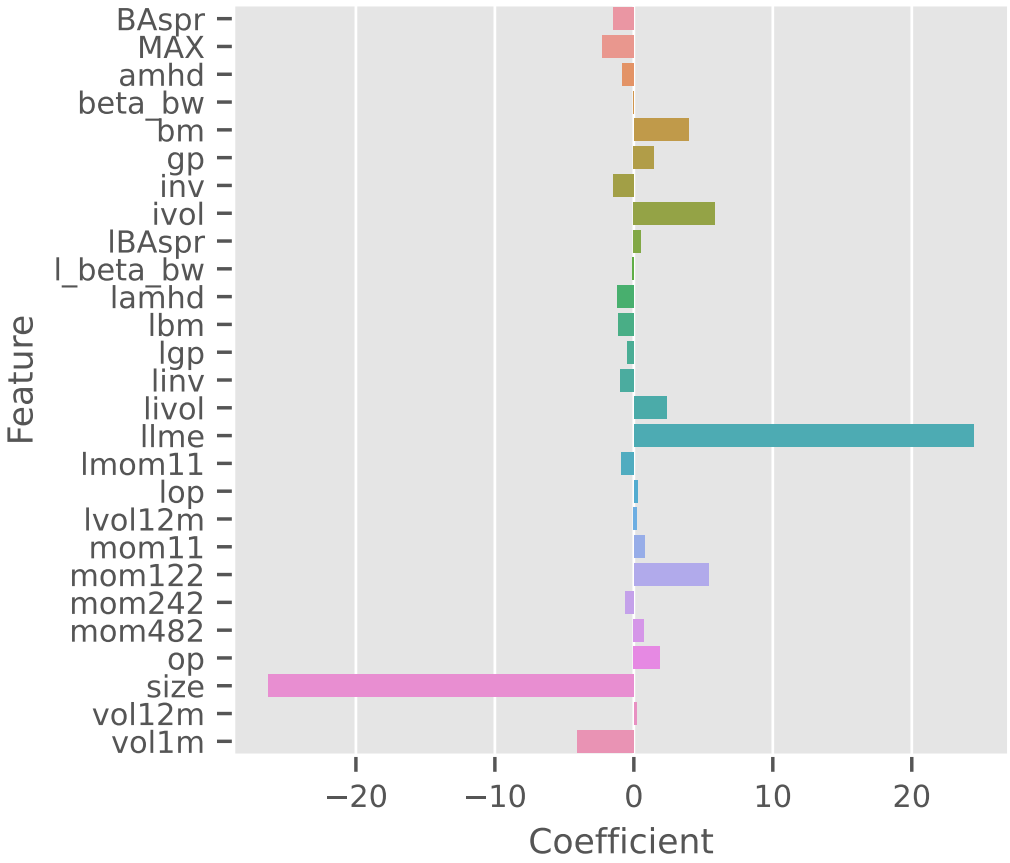

Figure 8: Feature importance in LASSO regression

The figures describes coefficient estimates for all features in LASSO regression. See Table 1 for feature description.

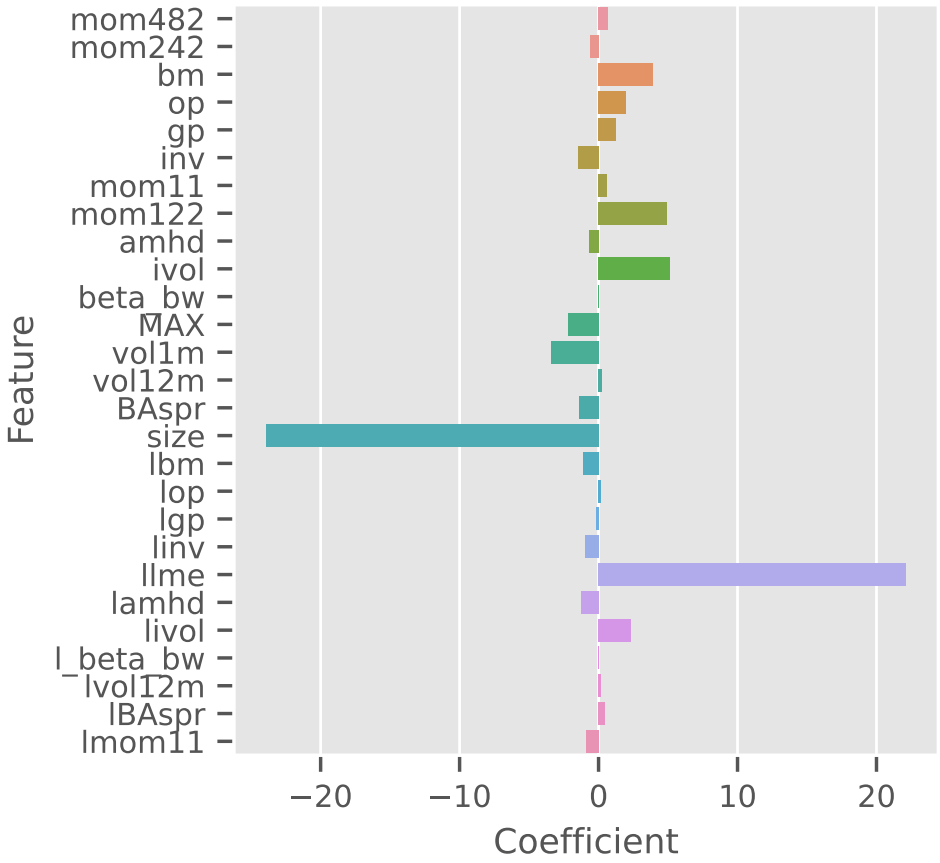

Figure 9: Feature importance in Ridge regression

The figures describes coefficient estimates for all features in Ridge regression. See Table 1 for feature description.

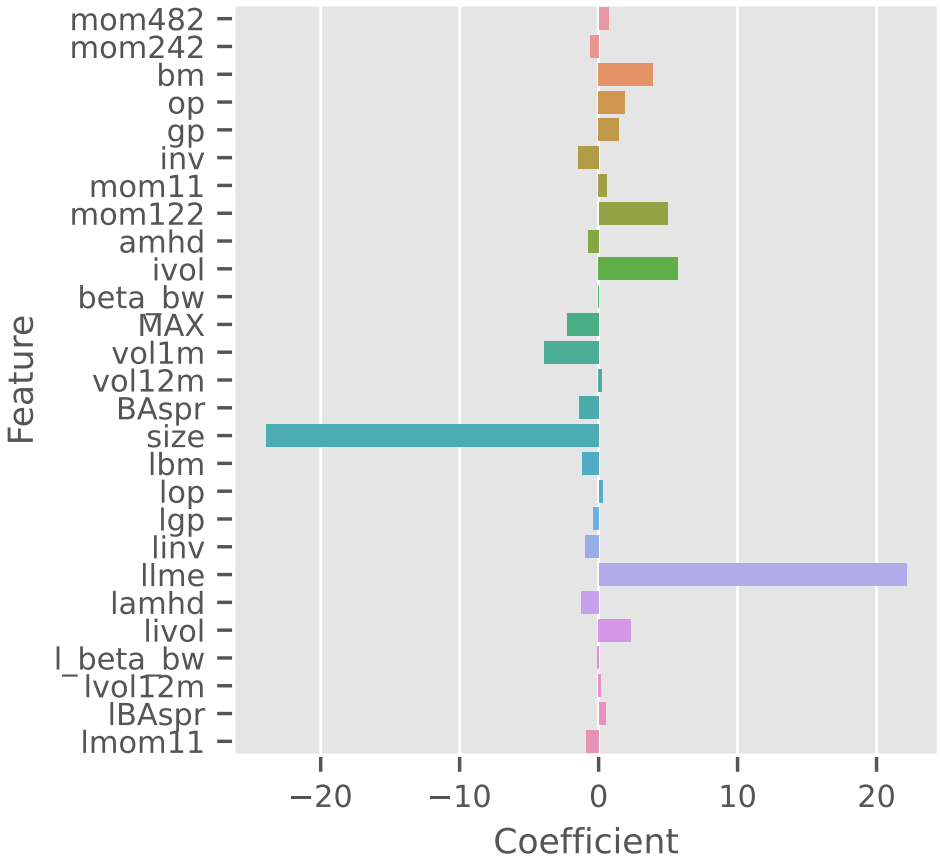

Figure 10: Feature importance in Elastic Net regression

The figures describes coefficient estimates for all features in Elastic Net regression. See Table 1 for feature description.

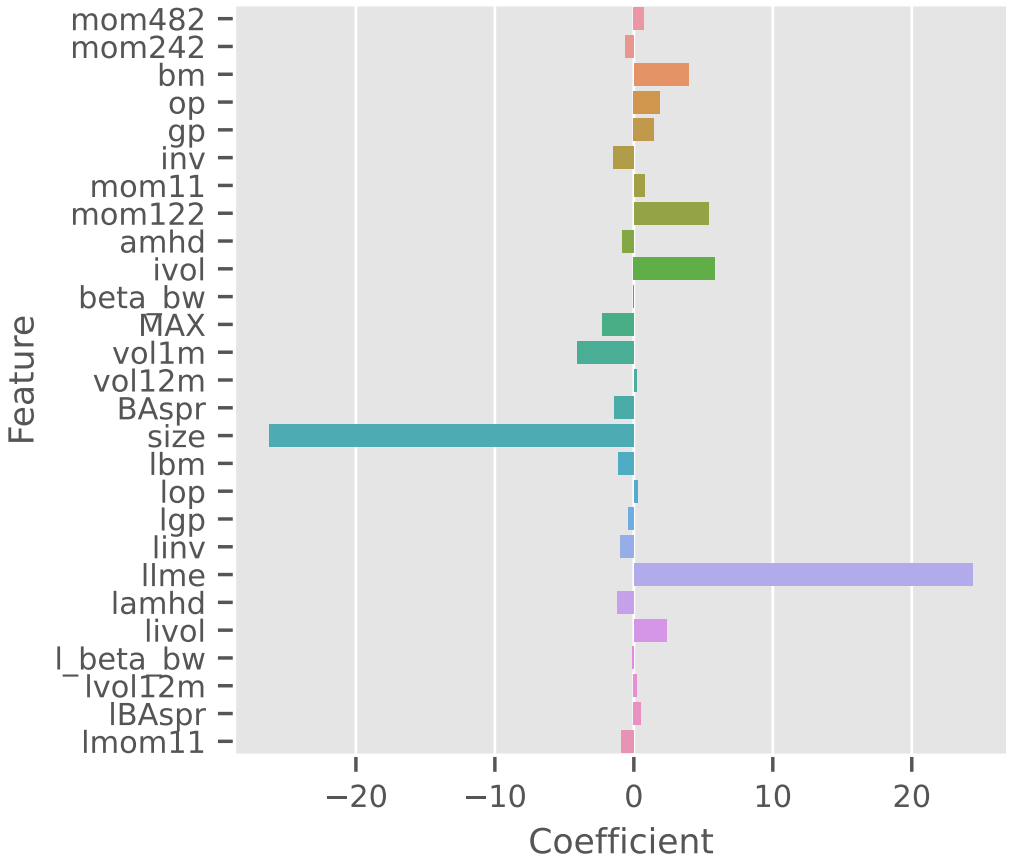

Figure 11: Feature importance in Tree model

The figures describes feature importance in a simple tree model. See Table 1 for feature description.

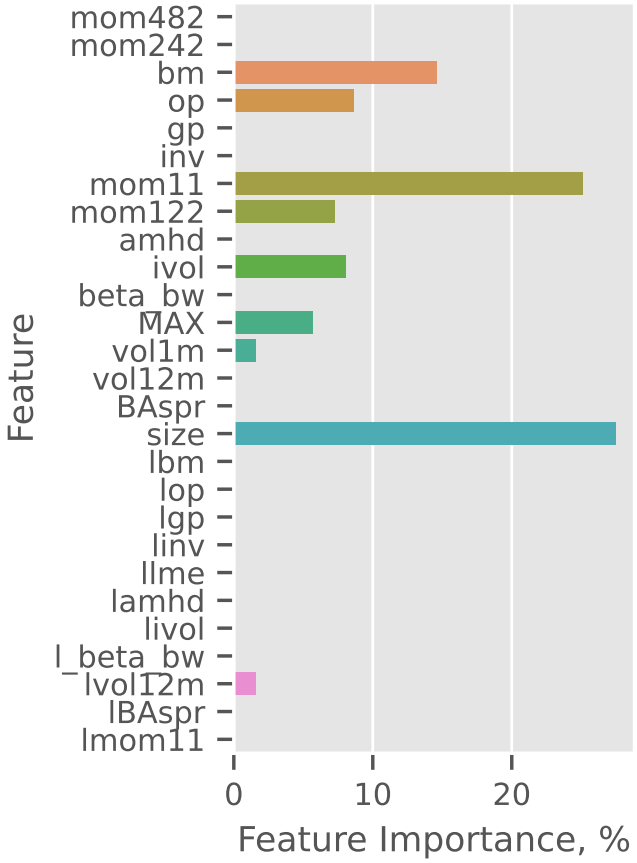

Figure 12: Feature importance in Random Forest regression

The figures describes feature importance in Random Forest model. See Table 1 for feature description.

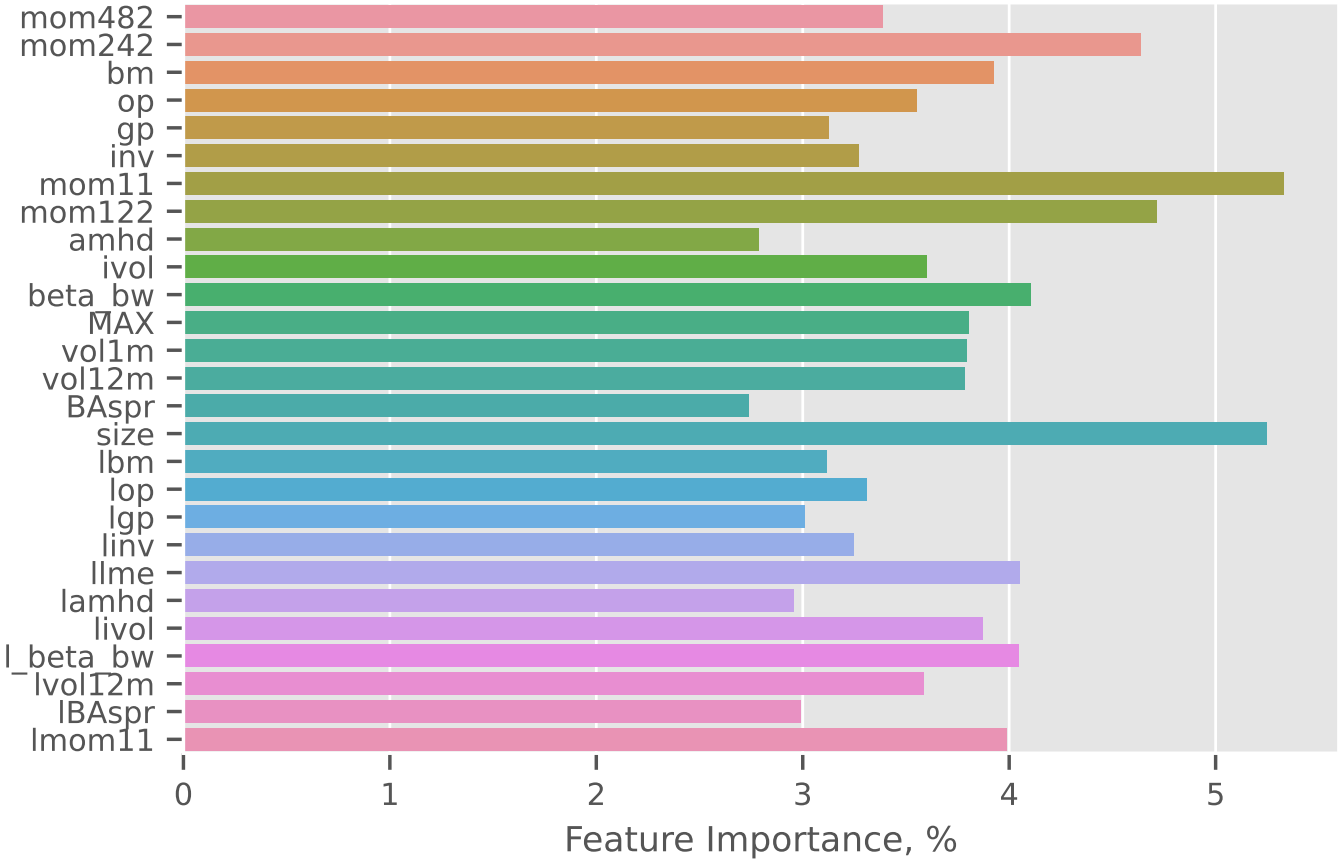

Figure 13: Feature importance in Boosted tree model

The figures describes feature importance in boosted Tree model. See Table 1 for feature description.

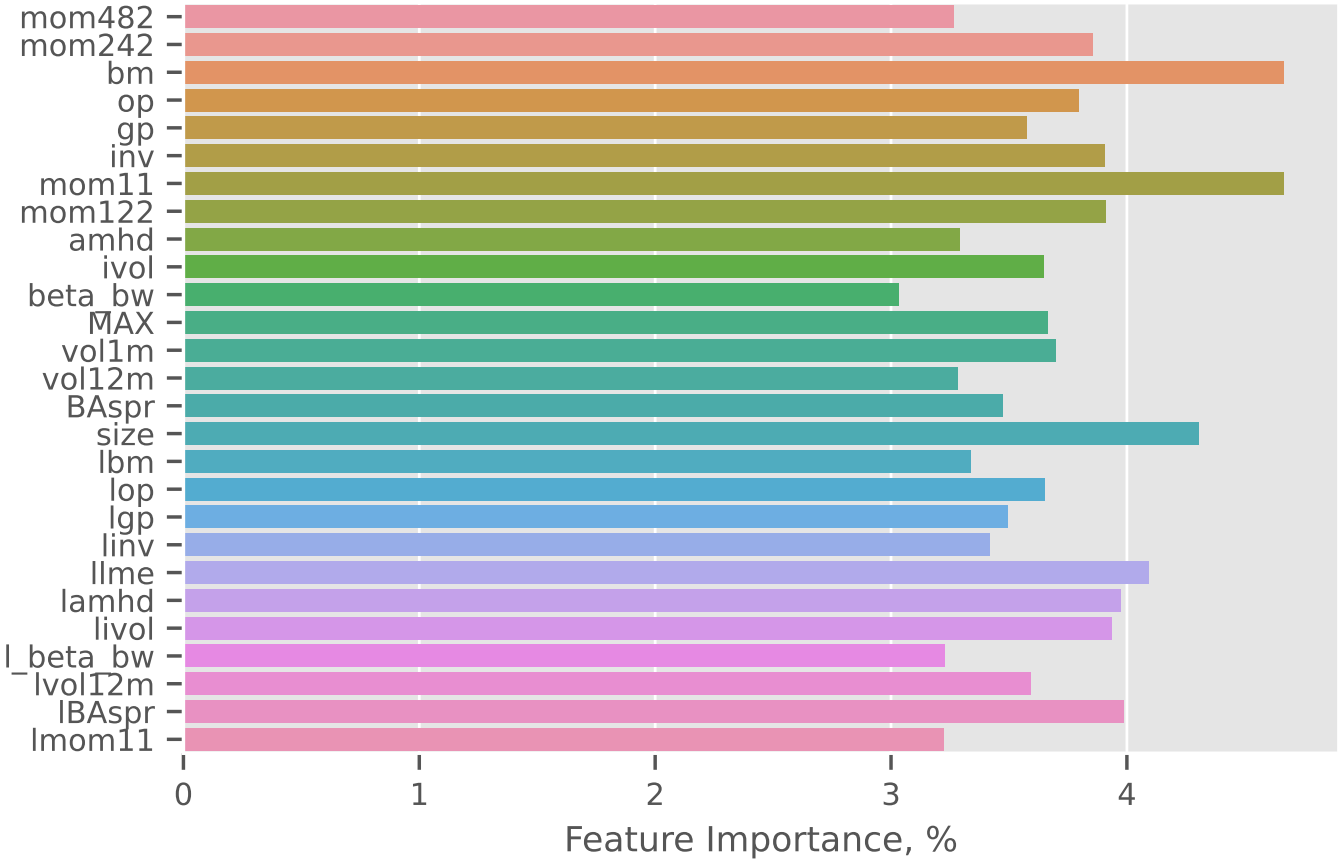

Table 1: Variable Description

The table describes all variables, used in the analysis. I use lagged values for some of these variables as additional features.

Variable code	Variable description
mom482	Past returns over 48 months excluding the last month.
mom242	Past returns over 24 months excluding the last month.
RET	Return in the current month (response variable).
bm	Natural Logarithm of Book-to-Market ratio.
op	Operating profitability.
gp	Gross profitability.
inv	Investment (annual change in total assets).
mom11	Return over the previous month.
mom122	Past returns over 12 months excluding the last month.
amhd	Amihud illiquidity ratio.
ivol	Idiosyncratic volatility from CAPM.
beta	Market beta
MAX	Highest return over the past year
vol1m	Volatility over the previous month.
vol12m	Volatility over preceding year.
BAspr	Bid-Ask spread.
size	Natural logarithm of market capitalization in the previous month.

Table 2: Summary statistics

The table reports summary statistics of all variables.

	count	mean	std	min	25%	50%	75%	max
mom482	1336352.00	38.18	125.21	-95.70	-31.02	7.91	60.90	844.70
mom242	1336352.00	19.89	83.07	-92.12	-31.44	3.16	46.64	493.52
RET	1336352.00	0.85	15.79	-80.00	-6.98	-0.37	7.13	199.87
bm	1336352.00	-0.44	0.93	-9.87	-0.98	-0.40	0.15	5.69
op	1336352.00	0.09	0.15	-0.97	0.05	0.11	0.16	0.45
gp	1336352.00	0.42	0.25	-0.70	0.24	0.38	0.54	1.42
inv	1336352.00	0.14	0.38	-0.56	-0.02	0.07	0.19	3.54
mom11	1336352.00	0.76	14.59	-43.11	-6.98	-0.37	7.16	68.12
mom122	1336352.00	9.19	53.60	-85.71	-23.14	1.18	29.09	303.53
amhd	1336352.00	1.81	2.85	-9.05	0.11	2.06	3.71	8.22
ivol	1336352.00	2.95	2.37	0.00	1.47	2.27	3.60	16.94
beta_bw	1336352.00	0.90	0.37	-0.94	0.64	0.88	1.13	2.79
MAX	1336352.00	7.29	6.96	-0.03	3.21	5.24	8.80	52.92
vol1m	1336352.00	3.18	2.43	0.00	1.66	2.50	3.88	17.45
vol12m	1336352.00	3.38	2.08	0.77	1.99	2.81	4.09	14.28
BAspr	1336352.00	2.23	3.83	0.01	0.77	1.17	1.66	33.33
size	1336352.00	4.83	2.29	-3.16	3.13	4.65	6.40	13.91

Table 3: Prediction accuracy

The table reports R^2 from predictive modeling. IS stands for in-sample and corresponds to R^2 on train sample from the best model, chosen via cross-validation. OOS corresponds to R^2 on test sample. The first two columns report results on the original sample with 27 features. The nest two columns contain results on the sample with 10 principal components of the original sample. Finally, The last columns represent results from 50 principal components of the augmented sample of 406 features, which are second-degree polynomial transformation of the original sample.

Model	s_IS	s_OOS	sp_IS	sp_OOS	lp_IS	lp_OOS
OLS	0.37	0.36	0.19	0.19	0.38	0.42
Lasso	0.34	0.35	0.18	0.18	0.34	0.42
Ridge	0.34	0.35	0.18	0.19	0.34	0.42
ElasticNet	0.37	0.36	0.19	0.18	0.38	0.42
KNN	0.29	0.64	0.19	0.53	0.20	0.52
Tree	0.45	0.37	0.21	0.08	0.23	0.13
RF	11.67	0.86	1.85	0.40	10.45	0.37
BoosetedTree	2.41	1.11	1.13	0.28	1.69	0.31
ANN	2.93	1.91	0.83	0.80	1.19	0.65