Empirical Asset Pricing via Machine Learning

Li Zhihan
Huang Fan
Zhang guangyu
Shangguan Wenfang
Wang Siyu

Abstract

We present a comparative analysis of machine learning approaches to a typical problem of empirical asset pricing: measuring asset risk premiums. We can find the most important characteristics for each mode we selected and we use machine learning to identify the best-performing methods (Random forests and neural networks). All methods agree on the same set of main forecast signals, which includes changes in momentum, liquidity, and volatility.

Keywords: Machine Learning, Asset Pricing, Prediction Model

1 Introduction

1.1. ***Background***

Risk premiums are notoriously difficult to measure: market efficiency forces return changes to be dominated by unpredictable news that masks the risk premium. Our study highlights what can be achieved in identifying the most informative predictor variables. This helps to solve the problem of risk premium measurement and thus contributes to more reliable research on the economic mechanisms of asset pricing.

Machine learning accommodates a far more expansive list of potential predictor variables, which enables gains that can be achieved in prediction and identifies the most informative predictor variables.

2 Literature Review

In solving the problem of measuring risk premiums, some scholars have adopted various methods, and the following are prevailing common and currently advanced methods:

2.1. ***Classical methods***

Machine learning method has already applied in financial industry, especially for return prediction. Norbert Keimling (2016) used a basic regression model to correlate stock market returns with the Shiller CAPE index (the long-term mean of the P/E ratio) to predict risk premiums. The conclusion is that CAPE index can predict the potential long-term returns of the S&P 500 index over the next 10 years [2]. Also, There're some research revealed that Conditional bias had achieved great success in explaining the expected risk premium of global investment portfolios. Campbell R. Harvey and Akhtar R. Siddique (2000) decompose expected excess returns into components caused by conditional variance and skewness in time series models. The research results indicate that conditional skewness is important and, when combined with the return on skewness throughout the economy, helps to explain the temporal changes in expected market risk premiums [3].

2.2. ***Recent Research***

As machine learning methods have gradually evolved, more diverse machine learning and deep learning research have emerged in recent years. Hsu et al. (2018) equip the Black-Scholes model and a three-layer fully-connected feedforward network to estimate the bid-ask spread of option price [4]. They argue that this novel model is better than the conventional Black-Scholes model with lower RMSE.Krauss et al. (2017) apply DNN, gradient-boosted-trees, and random forests in statistical arbitrage. They argue that their returns outperform the market index S&P500 [5]. SHIHAO GU (2021) devoted a significant amount of space to the application of various machine-learning methods in measuring risk premiums. For example, linear models, tree models, and neural network models [6]. The work analyzes the predictability of various model in detail, which benefit the work of replication. A novel non-parametric method has been introduced under the present value framework to estimate and test the time changes of equity premium prediction channels (Deshui Yu, 2023) [7].

3 Methodology

This section describes the collection of machine learning methods that we use in our analysis. In each subsection, we introduce a new method and describe it in terms of its three fundamental elements. Firstly, we characterize excess returns of assets with a model of an additive prediction error:

$$r_{i,t+1} = E_t(r_{i,t+1}) + \epsilon_{i,t+1}, \tag{1}$$

bonds are indexed as $i = 1, , N$, and months by $t = 1, , T_i$. Then We assume the same conditional expectation functional form for all bond returns and the function defined as:

$$E_t(r_{i,t+1}) = g(z_{i,t}). \tag{2}$$

We model the dynamics of individual corporate asset's excess returns using a large set of predictors in (2), which include stock characteristics $z_{i,t}$.

3.1. *Splitting and Tuning via Validation*

For training and testing purposes of our model, we should have our data broken down into three distinct dataset splits.

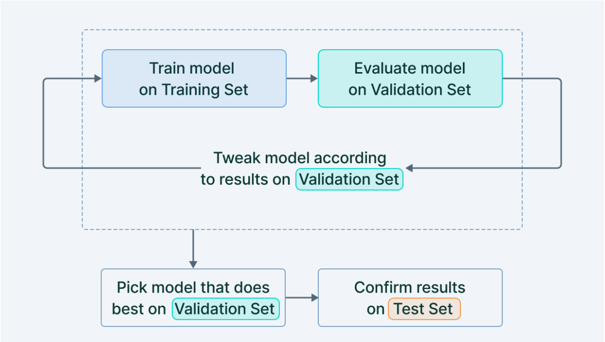

Figure 1: The process of splitting sample data.

3.2. *OLS*

Our baseline estimation of the simple linear model uses a standard least squares objective function:

$$L(\theta) = \frac{1}{NT} \sum_{i=1}^{N} \sum_{t=1}^{T} (r_{i,t+1} - g(z_{i,t}; \theta))^2 \tag{3}$$

Minimizing $L(\theta)$ yields the pooled OLS estimator. The convenience of the baseline objective function is that it offers analytical estimates and thus avoids sophisticated optimization and computation.

3.3. *ELastic Net*

The elastic net method overcomes the limitations of the LASSO (least absolute shrinkage and selection operator) method which uses a penalty function based on

$$||\beta||_1 = \sum_{j=1}^{p} |\beta_j| \tag{4}$$

The use of this penalty function has several limitations and the LASSO selects at most n variables before it saturates. Also if there is a group of highly correlated variables, then the LASSO tends to select one variable from a group and ignore the others. To overcome these limitations, the elastic net adds a quadratic part $||\beta||^2$ to the penalty, which when used alone is ridge regression. The estimates from the elastic net method are defined by

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}}(||y - X\beta||^2 + \lambda_2 ||\beta||^2 + \lambda_1 \sum_{j=1}^{p} |\beta_j|) \tag{5}$$

The quadratic penalty term makes the loss function strongly convex, and it therefore has a unique minimum. The elastic net method includes the LASSO and ridge regression: in other words, each of them is a special case where $\lambda_1 = \lambda, \lambda_2 = 0$ or $\lambda_1 = 0, \lambda_2 = \lambda$.

3.4. *Dimension Reduction: PCR*

The concept of predictor averaging, in contrast to predictor selection, lies at the core of dimension reduction. Creating linear combinations of predictors aids in reducing noise, thereby enhancing the isolation of valuable information within the predictors. Additionally, it assists in mitigating the correlation among predictors that might otherwise be highly interdependent. Principal Components Regression (PCR) is a well-known dimension reduction techniques.

PCR involves a two-step process. Initially, Principal Components Analysis (PCA) amalgamates the regressors into a compact set of linear combinations that most effectively preserve the covariance structure among the predictors. Subsequently, a few leading components are utilized in the standard predictive regression. In essence, PCR regularizes the prediction task by nullifying coefficients related to low variance components.

However, one limitation of PCR is its failure to encompass the ultimate statistical goal—forecasting returns—within the dimension reduction phase. PCA condenses the data into components based on the co-variation among the predictors. This occurs prior to the forecasting step, and it doesn't take into account how predictors are linked to future returns.

3.5. *Gradient Boosted Regression Tree(GBRT)*

Generalized linear models become computationally infeasible without prior assumptions involving interactions. Regression trees are an alternative solution widely used in machine learning for interacting with multiple predictors. GBRT mainly consists of the following two parts:

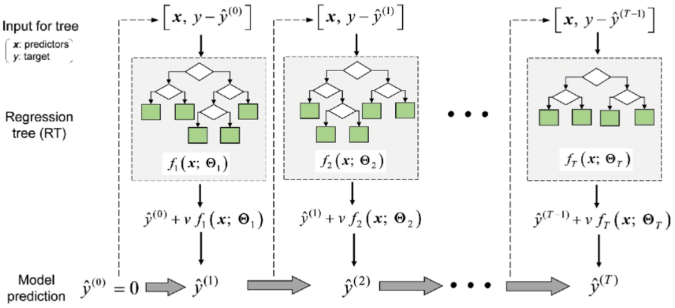

Figure 2: Schematic diagram of the gradient boosted regression tree.

- Regression Tree (RT): Regression tree is one of the decision tree categories used to predict actual values.

- Gradient Boosting (GB): Gradient enhancement involves iterating through multiple trees to jointly determine the final result. Each tree is the conclusion and residual of learning from the previous tree.

b) Gradient Boosting (GB): xGradient enhancement involves iterating through multiple trees to jointly determine the final result. Each tree is the conclusion and residual of learning from the previous tree.

The detailed process is:

a.Initialize tree:

$$f_0(x) = argmin \sum_{i=1}^{N} L(y_i, c) \tag{6}$$

b.Solve the residual of the previous model using the gradient descent method, and then use the minimum value of the residual of the previous model as the residual of the current model:

$$r_{mi} = -[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}]_{f(x)=f_{m-1}(x)} \tag{7}$$

c.Calculate the output values on each leaf node region:

$$c_{mj} = argmin \sum_{i=1}^{N} L(y_i, f_{m-1}(x), c) \tag{8}$$

Update:

$$f_m(x_i) = f_{m-1}(x_i) + \sum_{j=1}^{J} c_{mj} I(x \in R_{mj}) \tag{9}$$

d.Calculate the decision tree:

$$h_m(x) = \sum_{j=1}^{J} c_{mj} I(x \in R_{mj}) \tag{10}$$

e.Obtain regression tree:

$$f_M(x) = \sum_{m=1}^{M} \sum_{j=1}^{J} c_{mj} I(x \in R_{mj}) = \sum_{m=1}^{M} h_m \tag{11}$$

3.6. Random Forests

Random Forest is an ensemble machine learning algorithm that is used for both classification and regression tasks. At the core of the RF, a decision tree is a flowchart-like structure in which an internal node represents a feature (or attribute), the branch represents a decision rule, and each leaf node represents the outcome. The idea of random forests is to improve the variance reduction of bagging by reducing the correlation between trees, without increasing the variance too much.

3.7. Performance Evaluation

We are going to evaluate the performance of each model on the testing set by using this equation of out-of-sample R^2:

$$R^2_{oos} = 1 - \frac{\sum_{(i,t)\in T_3} (r_{i,t+1} - \widehat{r}_{i,t+1})^2}{\sum_{(i,t)\in T_3} (r_{i,t+1})^2} \tag{12}$$

where T_3 is the testing data, $\widehat{r}_{i,t+1}$ is the model's prediction. A higher out-of-sample R^2 indicates better predictive performance, with 1 being the ideal value representing a perfect fit to the unseen data, and values below 0 indicating poor performance.

3.8. Neural Networks

Neural networks, also known as artificial neural networks (ANNs), are comprised of node layers, containing an input layer, one or more hidden layers, and an output layer. Each node, or artificial neuron, connects to another and has an associated weight and threshold. If the output of any individual node is above the specified threshold value, that node is activated, sending data to the next layer of the network. Otherwise, no data is passed along to the next layer of the network.

We will focus on traditional "feed-forward" networks. These consist of an "input layer" of raw predictors, one or more "hidden layers" that interact and nonlinearly transform the predictors, and an "output layer" that aggregates hidden layers into an ultimate outcome prediction. Layers of the networks represent groups of "neurons" with each layer connected by "synapses" that transmit signals among neurons of different layers.

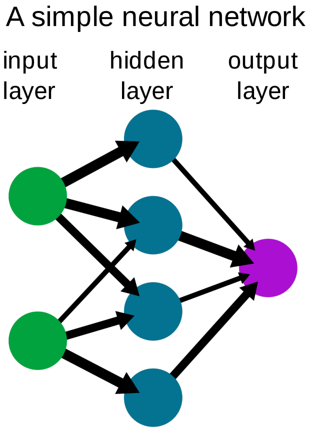

Figure 3: Simple neural networks.

4 Data Preparation

Our dataset includes monthly total individual equity returns for all firms listed in NYSE, AMEX, and NASDAQ. There are approximately 30,000 stocks over 60 years from March 1957 to December 2016. We divided the 60 years of data into 18 years of training sample (1957-1974), 12 years of validation sample (1975-1986), and the remaining 30 years for out-of-sample testing (1987-2016).

Based on the cross-section of stock returns literature, we also obtained 94 predictive firm characteristics about stock level. There are 61 of them which are updated annually, 13 quarterly, and 20 monthly. In addition, we include 74 industry dummies referring to the first two digits of SIC codes. The 8 macroeconomic predictors are constructed following Welch and Goyal (2008), which are not directly provided in the original dataset including dividend-price ratio(dp), earnings-price ratio(ep), book-to-market ratio(bm), net equity expansion(ntis), Treasury-bill rate(tbl), term spread(tms), default spread(dfy), and stock variance (svar).

5 Data Analysis

5.1. *Prediction Performance*

Figure 4 demonstrates the out-of-sample predictive R^2 of different models for predicting monthly return. For all 11 models we have tested, we could see that linear models showed poor performance and were dominated by nonlinear models. Among all models, Neural networks perform the best, especially the NN2 which produces an R^2 close to 0.8 percent, and NN1 which produces an R^2 over 0.4 percent, while all other models produce R^2 lower than 0.3 percent. We could learn from these observations that learning “deeper” may not be that helpful for predicting the monthly return, this may be because of the low signal-to-noise ratio in these kinds of financial data that may lead us to learn some “wrong” information when we learn “deeper”.

In Figure 4, the green and yellow columns show the R^2 for large and small stocks respectively. For large stocks, the results are about the same as for all stocks. However, when it comes to small stocks, the predictive power of NN2 has sharply dropped, and the NN1, random forest and XGBoost became the top 3 models that have clear advantages over other models, one possible reason is the lack of information on small stocks cause further lower signal-to-noise ratio so that the predictability quickly decrease even we just learn to two-layer model.

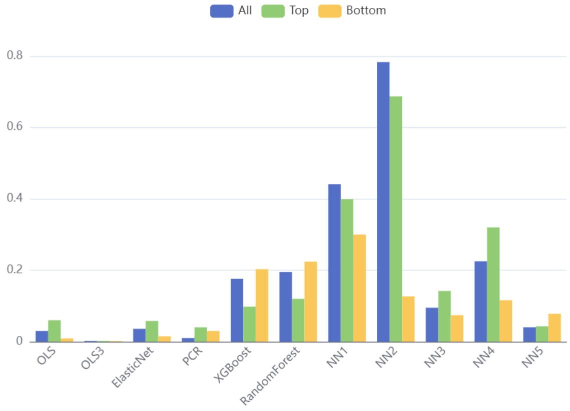

Figure 4: Out-of-sample predictive R2 of different models for predicting monthly returns.

We also compute the out-of-sample predictive R^2 of different models for predicting annual return (Figure 5). Compared with the model for monthly return, almost all models have better performance. This global improvement tells the models we used have predictive ability for longer windows. There are several interesting findings compared to the monthly return R^2: the OLS produces the highest R^2 while it performs almost the worst in the monthly return setting; the PCR ranks second in predicting the large stock while it produced almost the lowest R^2 when predicting all stocks and small stocks; the NN2 show great performance power in predicting small stocks while it is the clear shortcoming of this model in monthly return setting.

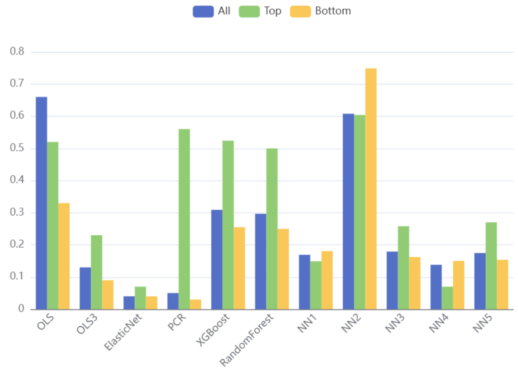

Figure 5: Out-of-sample predictive R2 of different models for predicting annual returns.

5.2. *Variable Importance*

Now we compute the variable importance for each model we selected. These scores are used to determine the relative importance of each feature in a dataset when building a predictive model. It can provide a way to rank the features based on their contribution to the final prediction, and the variable importance within the model is normalized to sum to one.

These figures show that characteristic importance magnitudes for ordinary least squares models and principal component regression models are highly skewed toward momentum and reversal.

Figure 6 shows the feature importance of the OLS model for the top 20 firm characteristics and it is clear that the ’secured’ has the most feature importance to the model prediction. The characteristic importance magnitudes for the OLS model are highly skewed toward momentum and reversal which means the performance of this model is relatively poor.

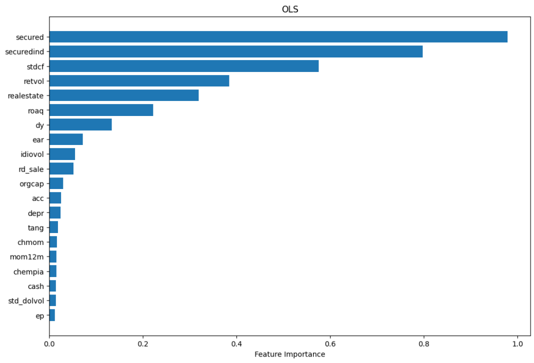

Figure 6: Feature importance for OLS model for firm characteristics.

Then we also obtained the graph of the feature importance of the PCR model for the top 20 firm characteristics(Appendix A.11) and we got the standard deviation of dollar volume(’std_dolvol’) as the most significant feature in this model. But the same problem occurred in this model which is the magnitudes are highly skewed toward momentum and reversal which means the performance of this model is also relatively poor.

Appendix A.12 demonstrates the feature importance of the Elastic Net model for the top 20 firm characteristics and it is obvious that the dollar volume('dolvol') has the most effect on the model prediction. The characteristic importance magnitudes for the OLS model are slightly skewed toward momentum and reversal which means the performance of this model is better than PCR and OLS models.

Next, we drew the figures for the feature importance of the neural networks model for the top 20 firm characteristics. For this part, we obtained 5 figures(Figure 7, Figure 8, Appendix A.13-15) for the NN1 to NN5 models. As we can see, all five models have the same characteristic which has the most importance which is log market equity('mvel1'). From these five Figures, we can easily find that neural networks are more democratic than the previous three models, and they can draw predictive information from a broader set of characteristics.

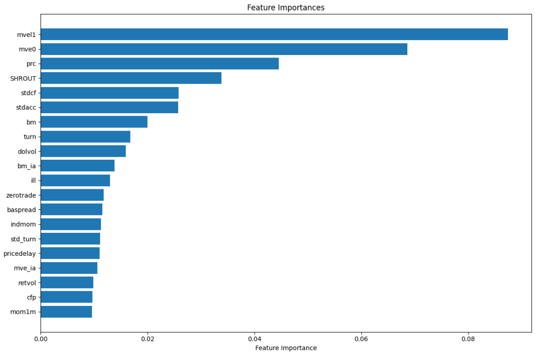

Figure 7: Feature importance for NN-3 model for firm characteristics.

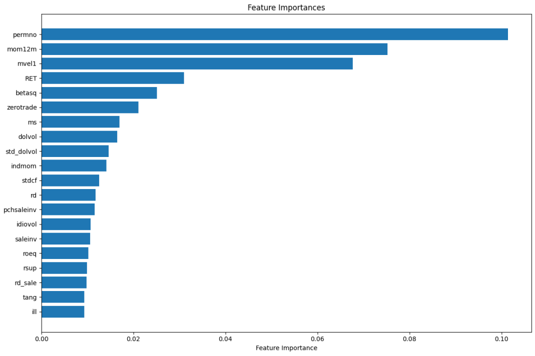

Figure 8: Feature importance for NN-5 model for firm characteristics.

Consider Appendix A.16 for the characteristic importance of the random forests model, 'mve0' has the biggest value of significance. In this figure, we can not directly see the importance of the other 16 features which indicates the RF has an average performance.

The last model we chose was XGBoost, and Figure 9 shows the feature importance of this model for the top 20 firm characteristics. 'mve0' becomes the most important characteristic, and 'mvel1' is the second one. Overall, the XGBoost is more democratic than other skewed models, and it also can draw predictive information from a broader set of characteristics in this project.

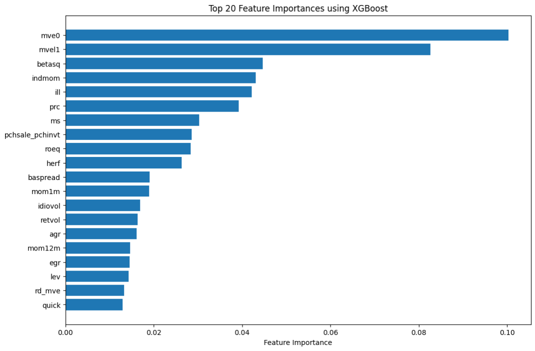

Figure 9: Feature importance for XGBoost model for firm characteristics.

5.3. *Model Performance Comparison and Analysis*

Figure 10 shows the R2-based importance measure for each 8 macroeconomic predictor variable (again normalized to sum to one within a given model). It demonstrates that OLS and PCR place similar weights on all predictors, potentially because these variables are highly correlated. And stock variance('svar') is the most important variable for these two models. The random forests model strongly favors bond market variables, including the default spread('dfy') and the Treasury rate('tbl'). All models agree that the aggregate stock variance('svar') is a critical predictor, whereas the book-to-market ratio('bm') has little role in any model. In conclusion, neural networks place great emphasis on exactly those predictors ignored by linear models, elastic nets, and random forests, such as the earnings-price ratio('ep') and book-to-market ratio('bm'). And random forests model is the second-best for attaching great importance to predictor variables that are easily ignored.

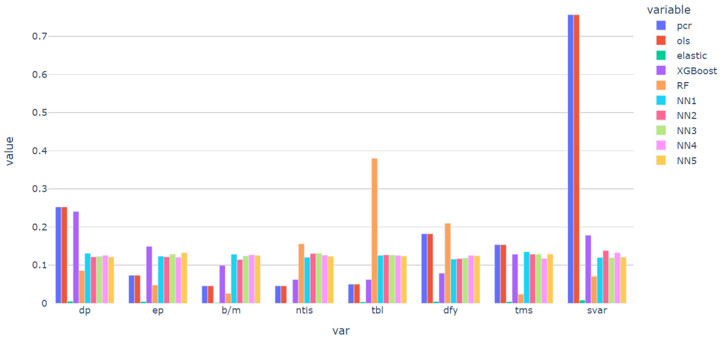

Figure 10: Feature importance for 8 macroeconomic predictor variables.

6 Discussion

6.1. *Strength*

- We have used many nonlinear models, which better capture complex nonlinear relationships compared to traditional linear models, thereby improving the accuracy of predictions.

- The modeling method we use can effectively handle large-scale datasets and utilize a large amount of historical data for modeling and training, thereby providing more accurate prediction results.

- We use models to automatically select and extract features with significant correlations, helping to identify key factors affecting risk premium and asset pricing, reducing the subjectivity of manually selecting features.

- The machine learning method we use has high flexibility, which is not only suitable for this paper, but also can meet various types of data and model requirements in other fields. It can adaptively learn and adjust according to the characteristics and patterns of the data, improving the predictive performance of the model.

6.2. *Drawback*

- Machine learning methods are often presented in the form of black box models, which pose certain difficulties in interpreting the prediction results and internal mechanisms of the model.

- Our work has some obvious flaws, such as not using PLS due to library issues.

- The output results of machine learning methods may be very sensitive to small changes in input data, leading to model instability. This means that there may be significant differences in the prediction results of the model under different training sets and parameter configurations. In practical operation, we used model parameters and training times that were not exactly the same as the original author, which resulted in some differences between our output and the original paper.

7 Summary and conclusions

In summary, neural networks and tree models perform well. Their advantage lies in supplementing the nonlinear interactions missed by other methods. However, in the actual process of machine learning modeling and prediction, simply using to evaluate the performance of a model is not entirely accurate. Other evaluation indicators such as MSE should also be introduced to comprehensively evaluate the performance of a model.

Meanwhile, with the advancement of technology, especially the application of machine learning in the economic field, risk premiums will be better measured, asset pricing methods will become more diverse, and accuracy will gradually improve. Our research has found that momentum and reversal, which are indicators of stock price trends, are of great significance for our research. As more and more data analysts enter this field, the role of machine learning in the field of financial technology will become increasingly important, and related research results will emerge endlessly

8 Contribution

Coding: Shangguan Wenfang, Li Zhihan;
Report: Wang Siyu, Huang Fan, Zhang Guangyu.

References

[1] Wikimedia Foundation. (2023, January 29). Elastic net regularization. Wikipedia., from *https://en.wikipedia.org/wiki/Elastic_net_regularization*

[2] Norbert Keimling, ”Predicting Stock Market Returns Using the Shiller CAPE — An Improvement Towards Traditional Value Indicators?”.

[3] Campbell R. Harvey, Akhtar R. Siddique, ”Time-Varying Conditional Skewness and the Market Risk Premium”.

[4] Hsu, P. Y., Chou, C., Huang, S. H., & Chen, A. P. (2018). A market-making quotation strategy based on dual deep learning agents for option pricing and bid-ask spread estimation. The proceeding of IEEE international conference on agents (pp. 99–104).

[5] Fischer, T., & Krauss, C. (2017). Deep learning with long short-term memory networks for financial market predictions. European Journal of Operational Research, 270(2), 654–669.

[6] SHIHAO GU, ”MACHINE LEARNING IN EMPIRICAL ASSET PRICING”.

[7] Deshui Yu, Nonparametric Estimation and Testing for Instability of Discount Rate and Cash Flow Channels of Stock Returns”.

[8] What are neural networks? IBM. (n.d.). *https://www.ibm.com/topics/neural-networks#What+is+a+neural+network%3F*

Appendix A Appendix Graph

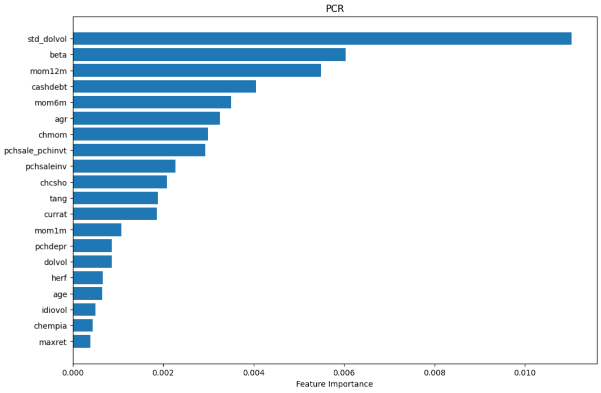

Figure A.11: Feature importance for PCR model for firm characteristics.

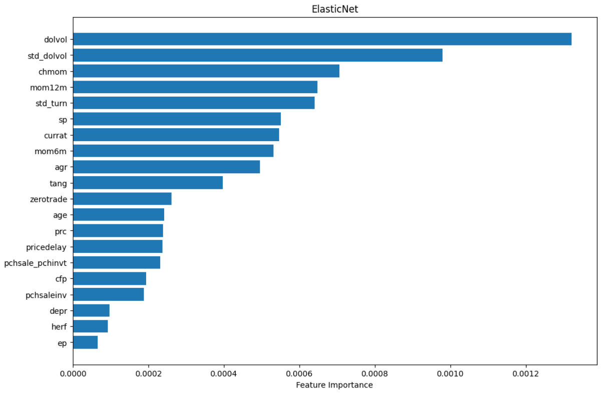

Figure A.12: Feature importance for Elastic Net model for firm characteristics.

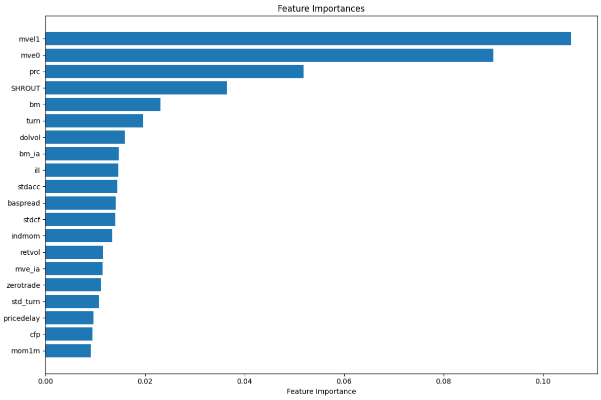

Figure A.13: Feature importance for NN-1 model for firm characteristics.

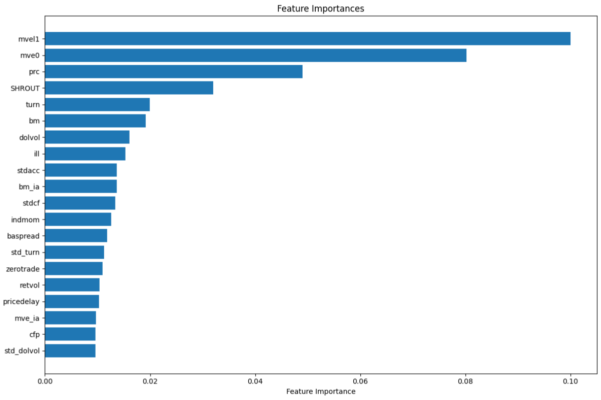

Figure A.14: Feature importance for NN-2 model for firm characteristics.

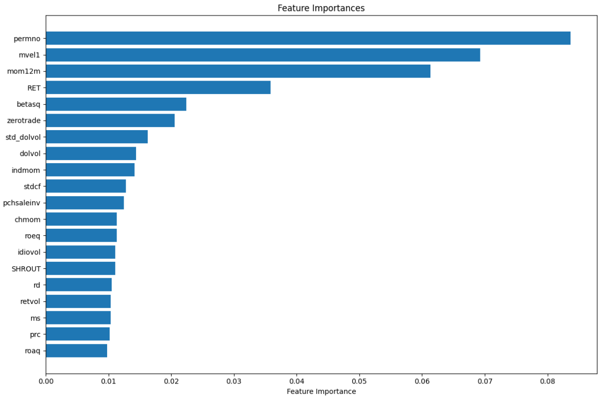

Figure A.15: Feature importance for NN-4 model for firm characteristics.

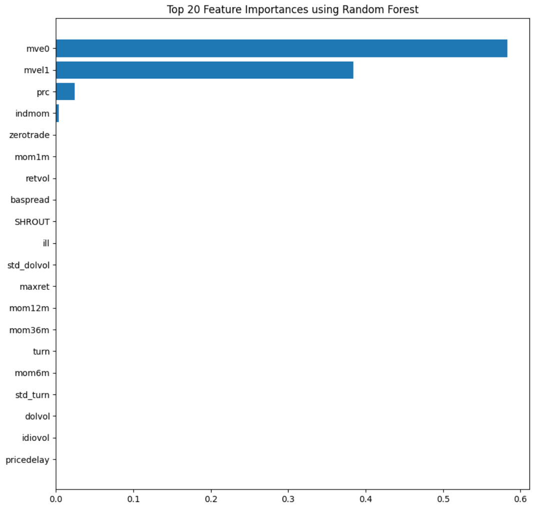

Figure A.16: Feature importance for random forests model for firm characteristics.