Testing Stock Prediction Viability using Classical Machine Learning Approaches

Tong Zhang
ttz2104@columbia.edu

Abstract

The goal of this study is to show that regression techniques can predict stock price returns with lower mean squared error ("MSE") than predicting using the average return over the period considered. This study examines share price returns for companies listed on the Standard & Poors 500 Index ("S&P 500") from January 2nd, 2009 to September 30th, 2019. This study found there was no significant improvement in MSE between using regression models to predict stock returns and using the mean return over the period considered, but indicates that some techniques show more promise than others.

1 Introduction

Quantitative investment firms have developed trading algorithms using sophisticated machine learning techniques known as deep learning which signal when to buy, when to sell, and how to size a position relative to the overall portfolio. Deep learning models can make accurate predictions with a high dimensional feature space, but require many examples, making them very computationally expensive to train. Deep learning models are commonly referred to as "black box" models, as their inner workings are difficult to interpret, making them difficult to debug in the event of failure. Traditional machine learning techniques such as regression have proven to make accurate predictions with less data and less training time than more complex models created with Deep Learning. We believe these simple techniques can also predict future stock returns, using historical share price return data alone.

Machine learning is a subset of the field of Artificial Intelligence ("AI"), whereby computer software can mimic some behaviors or tasks of naturally intelligent organisms, i.e. humans. First proposed by Alan Turing in 1950 and famously expanded upon in 1956 at a historic conference at Dartmouth College, AI has witnessed an explosion in popularity over the last decade (Mijwil, 2015). A subset of AI known as machine learning ("ML") is responsible for most of the growth of the field. In his textbook Machine Learning, Tom Mitchell, a professor in the Machine Learning Department at Carnegie Mellon University, describes ML as follows: "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience."

Machine learning can be broken down into three broad categories: supervised learning, unsupervised learning, and reinforcement learning. In this paper, we will limit our discussion to supervised learning, where modelers work with labeled training examples to learn the relationship of a set of dependent

variables, known as features, with some dependent variable, known as a target. For example, regression algorithms can predict the price of a house based on the number of bedrooms, amount of square footage, and age of the building, which are the features. Supervised ML algorithms learn by making a prediction based on the features, measuring the error between the prediction and the true label, and updating weights for the features to lower that error. By iterating over a dataset many times and updating the weights after each pass, an algorithm can determine the best weights given the features to predict with the lowest loss.

2 Models

This study will explore the efficacy of regression algorithms to predict stock returns, or the percentage change of the price of a share of stock from day n to day n+1. Regression algorithms predict continuous numerical targets, and are thus well suited to the problem of predicting returns. We will discuss the models employed in our analysis before turning to previous research.

2.1 OLS Regression

Ordinary Least Squares regression, commonly known as linear regression, is one of the oldest statistical learning techniques which is still widely used in practice. If a practitioner believes there is a linear relationship between the dependent and independent variables, he or she can fit a line to the data which minimizes the squared residuals between the true value of the dependent variable and the value predicted by the line. It models the target variable as a linear combination of the features in the form $y = wx + b$. The optimal weights and bias terms can be solved via the normal equation, or by using the iterative weight update algorithm gradient descent. Gradient descent updates weights based on the loss function, calculated over the errors at each step, where the loss function is

$$L = \frac{1}{n}\sum_{n=1}^{N}(y_i - wx_i + b)^2$$

2.2 LASSO Regression

LASSO (least absolute shrinkage and selection operator) regression is the same as OLS regression, but which performs L1 regularization to the weights. An additional penalty is added to the loss function:

$$L = \frac{1}{n}\sum_{n=1}^{N}(y_i - wx_i + b)^2 + \lambda|w|$$

The effect of this regularization term is to shrink the weight parameter, where the magnitude of the shrinkage parameter is set by the scalar λ. For higher values of λ, the shrinkage can push weights to zero, effectively performing feature selection, in addition to reducing model complexity.

2.3 Ridge Regression

Ridge regression is also a modification of OLS regression, but performs L2 regularization instead of L1 regularization.

$$L = \frac{1}{n}\sum_{n=1}^{N}(y_i - wx_i + b)^2 + \lambda w^2$$

L2 regularization has the effect of squeezing the weight parameters without pushing them all the way to zero.

2.4 Random Forest Regression

Ensemble techniques make predictions by averaging the outputs of many weak models which, on their own, perform only slightly better than random. The Random Forest model averages the outputs of Decision Trees,

which make predictions by recursively splitting a dataset into sub-groups based on a single feature at each split. The feature chosen at each step is the one which will result in the greatest decrease in entropy at each step. To make predictions on new data, the tree asks a series of logical questions like "is feature x_i greater than or less than some threshold" and goes left or right down to the next level accordingly. If a node is pure, or contains examples with labels only from one class, no further splits can be made, and unlabeled examples which land in this node are assigned the label of the examples in that node.

The simple decision trees which comprise a Random Forest are trained on a subset of features, and a bootstrapped subsample of the data, which individually perform poorly on new examples, but when taken as a whole can make very accurate predictions. Because each weak learner is trained on a subset of data and features, ensemble models avoid overfitting. Random Forests can be applied to both classification and regression tasks.

3 Previous Research

There is heated debate about whether or not it is possible to consistently generate alpha in the stock market, where alpha is a return above the returns of the index. Eugene Fama published research in 1970 stating that all available information about a stock is baked into the price, or that markets are strongly efficient, only to revise that claim in 1991 to allow for some temporary inefficiencies in prices, or that prices are weakly efficient (Fama, 1991). He stated, however, that markets are efficient enough that the cost to make a profit on extra information is greater than any potential gains, meaning the inefficiencies which do exist cannot be profitably traded upon. Price movement in the absence of news, then, can be modeled as a sort of random walk driven by investor emotion.

The true test of whether it is possible to beat the market is whether professional investors have been able to consistently beat the market. A notable standout is the hedge fund Renaissance Technologies, whose Medallion fund returned an average of 66.1% per year from 1988 to 2018 (Wigglesworth, 2019). The outstanding, long-term performance of Renaissance, an early adopter of machine learning techniques for pattern recognition, indicates there are patterns in stock price movements which can be exploited for financial gain.

3.1 A Non-Random Walk Down Wall Street

Some researchers emphatically reject the random walk hypothesis. Andrew Lo, a finance professor at MIT Sloan Business School, and his team, tested the random walk theory for weekly stock prices, and showed with strong significance that they do not follow a random walk (Lo and Mackinlay, 1988).

While the returns of Renaissance indicate prices are not random, the public has no way of verifying their methods. Lo's research gives us the rigorous proof we need to say that stock prices, and thus returns, exhibit some non-random behavior, which can be exploited.

3.2 Regression vs Neural Networks

The deep neural networks employed by top quantitative trading firms are very computationally expensive to train, and require orders of magnitude more data to train than lower complexity models. While top hedge funds have implicitly been showing the effectiveness of high level neural networks at return prediction, Assis et al. found consistent returns with these complex neural netorks as well. Specifically, they used an ensemble of Neural Networks to predict stock price

changes in the Brazilian stock markets (Assis et al., 2018). By using this ensemble of many high level machine learning networks such as feed forward, cascade forward and elm networks, they were able to predict price changes in the market with statistical significance. However, because of the complexity of the models and the need to for multiple training sets, they were unable to train the model or make predictions on small time frames of information. However, it is often essential to find models which could perform well with limited training examples and low training costs. As early as 1998, researchers were aware of the potential of neural networks to overfit small datasets, and to make poor predictions on unseen data. Comparatively, simple regressors were found to train on smaller datasets and still perform well on unseen data (Desai and Bharati, 1998). This gave us confidence that we could still make predictions with low error using regression models and smaller datasets.

3.3 Ensemble Techniques for Directional Prediction

Machine learning algorithms are tools, and, like hammers and screwdrivers, are better suited to different use cases. In a typical quantitative investing process, different algorithms can be used for different parts of the investing pipeline. A practitioner might use one algorithm to predict whether the price of a stock will go up or down, and another to predict the magnitude of that movement. Random Forests have been found to be effective for predicting the direction of stock price movement (Khaidem et al., 2016). Our study is not to predict stock prices, but rather the daily return of the stock as a percentage of the stock price. The magnitude of returns is so small, that just getting the direction of the movement correct could lead to successful strategies.

3.4 Forecasting Using LASSO

Our research seeks to identify trading strategies which perform well across different market environments using data at short, medium, and long term time horizons. To predict future returns of a single company, at, for example, 250 trading days, we would have 249 features to predict returns on the 250th day. High dimensional feature spaces can lead to algorithms which overfit, or learn the noise of a specific dataset. LASSO Regression, by its very nature, is effective at reducing the complexity of a model by limiting the impact of weights on features, and was found to perform well in stock price prediction tasks (Roy et al., 2015). LASSO regression satisfies our requirement that our model be relatively simple, so that a team can use it for prediction with minimal training costs.

3.5 Comparing Classifiers

Part of any model building process is model selection. As mentioned previously, different models perform differently for different tasks. We wanted to explore the research on how different models stacked up against each other when it came to stock price prediction. In the case of predicting the direction of stock prices, comparisons have been made between modern, neural network approaches, and older ensemble algorithms. An analysis on 5767 European equities found that random forests outperformed several competing algorithms, in predicting stock price directional movement one year ahead (Ballings et al., 2015). This research served to validate our inclusion of the Random Forest regressor the with OLS, LASSO, and Ridge regressors.

4 Research Question and Hypothesis

The successful performance of quantitative hedge funds and the academic research sam-

pled lends itself to the view that machine learning algorithms can beat the market. We centered our experiment around the following research questions:

- Is it possible for simple algorithms to predict future stock returns?
- Can those algorithms be successful if trained only on previous price data?
- What is the best simple algorithm to use?
- Are the algorithms more successful over a different time horizon?

Based on our research, we hypothesized that classical machine learning algorithms, using only historical price data, can predict future returns more accurately than by simply predicting the average value of returns over a given time period.

5 Data and Methodology

5.1 Data

For this study, we pulled data using Wharton Research Data Services (WRDS) collection of databases. Specifically, we accessed the CRSP stock database and pulled all the daily stock price data from the S&P 500 from January 2009 to September 2019 (the end of the third quarter of the financial year).

We then cleaned the data and removed any company that was not a part of the S&P 500 for the full duration of our sample period. This left us with 269 companies to examine.

Due to the nature of all these stocks existing withing the S&P 500 for the duration of the sample, all these stocks would be considered Large Cap Stocks. Fama and French found in their seminal paper that the degree of future movements in stock prices depending on past prices was dependent on if the company was considered ”large” or ”small” (Fama and French, 1996). Accordingly, we ensured that our data set did not then mix stock price data of companies that were drastically different in size. This hopefully will reduce the excess noise in the data and allow our price prediction models to be better trained and function better.

5.2 Data Cleaning and Preparation

The data pulled from WRDS contained daily stock prices, CUSIP code, date, and the percentage return between each time step. We were interested only in the returns and the company name, so all other features were dropped. The original data structure used one row to list the price and return for each company for one day. In order to make predictions for each company, we reorganized the data such that each row was one company, and each feature was the return on different trading days. The resulting data frame was one with 269 rows (companies) and 2704 columns (trading days).

To maximize our chances of making good predictions, we needed to use all of our data to train our models. As we also wanted to test the effectiveness over different time horizons, we needed to reconstruct our data frame for each period. To do this, we split the training data column-wise at each time horizon, and concatenated each subsection so our model would learn to predict returns across the entire time range. For example, for the 10 day time horizon, we split the training data into 270 subsections, and stacked them on top of each other to use as training data, where returns for the first 9 days were features, and the return on the 10th day was the target. This process was repeated for all time horizons.

5.3 Methodology

To establish a baseline prediction, we calculated the average return in the training data

over a given time horizon. To test our hypothesis, we compared the squared error for each of the models to the squared error of the baseline average guess strategy. We then performed a one-sided t-test to see if the error using the baseline was greater than the error using the model. Our null hypothesis was that there was no difference between error, and our alternative hypothesis was that the error of the mean predictions was higher than that of the trained model predictions. This experiment was repeated for each model at each time horizon.

We selected our time horizons to test effectiveness at several typical investor time horizons: short-term, medium-term, and long-term. For these categories we chose to check performance using data for 10 trading days (two weeks of trading), 45 trading days (two months of trading), 250 trading days (one year of trading), and 2,704 trading days (all data).

6 Results

The results of the t-tests for the different models at different time horizons can be found in the table.

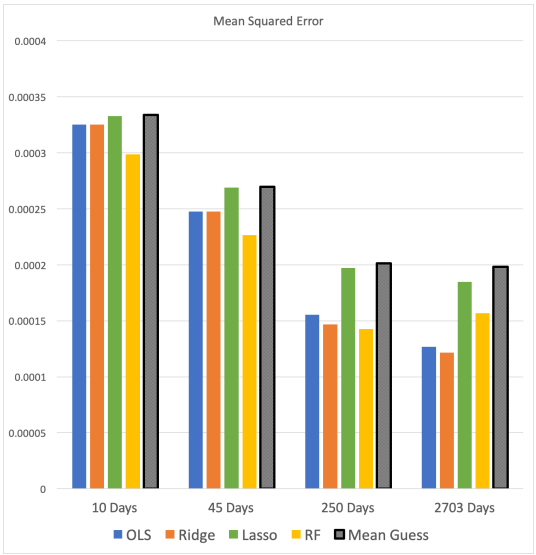
models	10 Days (t-val, p-val)	45 Days (t-val, p-val)	250 Days (t-val, p-val)	2703 Days (t-val, p-val)
OLS	(0.316, 0.376)	(0.774, 0.219)	(0.928, 0.177)	(0.797, 0.215)
Ridge	(0.314, 0.377)	(0.783, 0.217)	(1.067, 0.143)	(0.843, 0.202)
Lasso	(0.029, 0.488)	(0.022, 0.491)	(0.076, 0.47)	(0.121, 0.452)
RF	(1.423, 0.077)	(1.482, 0.069)	(1.166, 0.122)	(0.397, 0.346)

We can see that we were unable to reject the null hypothesis for any model at any time horizon. It is worth noting, however, that the random forest approaches significance at the 10 day and 45 day time interval.

7 Conclusion

The results of our experiments indicate that we cannot expect simple regression algorithms to perform better at stock return prediction than the mean over the period considered. While it is disappointing not to have a successful trading strategy, we must recall that models trained in an applied setting would likely have many more features to choose from, and, with more experiments, we can whittle down the number of strategies until we find one that can predict returns.

Of ancillary benefit is the knowledge that more trading days of training data yield diminishing marginal returns in information gain.



This chart comparing MSE for each model at each time horizon shows significant improvement in MSE from 10 days to 45 days, and from 45 days, to 250 days, but only marginal improvement at 2,704 days. In future experiments, we will take this into account for designing our experiments

8 Future Work

8.1 Random Forest Exploration

One key area of future work that our research points towards is a deeper investigation around the effectiveness of the random forest model in price prediction. While we discussed earlier that the random forest model did not produce significant results, there is the previously mentioned trend of almost predict-

ing significantly better. Future work should look into:

- Is the near significance due to the average prediction values being so close to zero?
- Is there a greater trend of random forest predicting significantly well across other periods?

We believe that random forest tended to have better prediction performances than our other models is due to the actual predictions being so close to zero. As discussed earlier in the Previous Research section, it has already been shown that random forests are the best at classification prediction within the models we tested. As a result, if the random forest model assumes a classification problem and learns to predict the return direction with accuracy, it would likely become a relatively accurate model overall.

Further studies could try to replicate our random forest findings but instead change the prediction target to be weekly or monthly return. As a result, the random forest would be predicting percent change in prices that are much further from zero.

8.2 Small Cap Replication

While the results for the large cap S&P 500 companies did not have significance, a future study could replicate our study with small cap companies instead. Lo et al. found in their research that smaller companies more obviously did not follow a random walk than larger companies (Lo and Mackinlay, 1988). From this, it would be expected that historic price data would then provide a stronger signal for small companies. Hence, there is potential that the machine learning methods tested would be more likely to find significance.

The study could utilize the Russell 2000, a prominent index of 2000 small cap companies. However, the process of preparing that study may be difficult, as companies tend to enter and leave the Russell 2000 index with much higher frequency. This may result in having a shorter duration for the sample period to make sure there are enough small cap companies that stayed in the index for the entire period of interest.

8.3 Significance across Business Cycles

While we split our data so that to ensure our data was all coming from a similar market period, a future test should examine if the inclusion of other parts of the business cycle would alter the findings of the paper. Specifically, a future paper should look to include a full business cycle. While future researches might run into the difficulty of properly delineating a full business cycle, a proposed period could start at the beginning of the fourth quarter of 2007 (the high to the market before the financial downturn) to present day.

Acknowledgments

We would like to thank our instructor Michelle Levine for her guidance in the development of this project. Additionally, we would like to thank Tom Zhang for his technical expertise and giving advice on potential avenues for the research project. Specifically, his own previous research helped in our idea generation to be able to land on our thesis and hypothesis for this research project. Thank you.

References

Assis, J. D. M., Pereira, A. C. M., and Silva, R. C. E. (2018). Designing financial strategies based on artificial neural networks ensembles for stock markets. *2018 International Joint Conference on Neural Networks (IJCNN)*.

Ballings, M., Poel, D. V. D., Hespeels, N., and Gryp, R. (2015). Evaluating multiple classifiers for stock

price direction prediction. *Expert Systems with Applications*, 42(20):70467056.

Desai, V. S. and Bharati, R. (1998). A comparison of linear regression and neural network methods for predicting excess returns on large stocks. *Annals of Operations Research*, 78(0):127–163.

Fama, E. F. (1991). Efficient capital markets: Ii. *The Journal of Finance*, 46(5):15751617.

Fama, E. F. and French, K. R. (1996). Multifactor explanations of asset pricing anomalies. *The Journal of Finance*, 51(1):55.

Khaidem, L., Saha, S., and Dey, S. R. (2016). Predicting the direction of stock market prices using random forest. *arXiv e-prints*, page arXiv:1605.00003.

Lo, A. C. and Mackinlay, A. C. (1988). Stock market prices do not follow random walks: Evidence from a simple specification test. *The Review of Financial Studies*, 1(1):4166.

Mijwil, M. (2015). History of artificial intelligence. 3:1–8.

Roy, S. S., Mittal, D., Basu, A., and Abraham, A. (2015). Stock market forecasting using lasso linear regression model. *Advances in Intelligent Systems and Computing Afro-European Conference for Industrial Advancement*, page 371381.

Wigglesworth, R. (2019). The man who solved the market - how jim simons built a moneymaking machine.

9 Contributions

Nick and Billy shared responsibilities for most sections of this project. Nick found most of the helpful research articles which informed our research questions and hypothesis. Billy wrote all the code, but Nick, with his deep expertise in statistics, was instrumental in ensuring accurate and meaningful results from our experiments. Nick created most of the presentation and Billy took the lead in writing most of the sections of the paper. The idea generation, brainstorming, and outlining for both the presentation and paper were done while meeting together.