Development & Evaluation of a Machine Learning Based Value Investing Methodology

Jun Yi Derek He
Academy of Science and Technology
The Woodlands College Park High School
The Woodlands, TX
derekhe99@yahoo.com

Joseph Ewbank
Academy of Science and Technology
The Woodlands College Park High School
The Woodlands, TX
jewbank@conroeisd.net

Abstract – *The majority of approaches to utilize computers for fundamental analysis in stock investing is plagued with scalability and profitability issues. The present work tested four machine learning algorithms to overcome them. Random Forest and a soft voting ensemble obtained strong risk-reward ratios over 12 test years. An innovative process involving picking exclusively the top 20 stocks based on the algorithms' softmax confidence enhanced returns by approximately 30%. Methods such as comparing in sample and out of sample precision performance as well as distributions of test and training data suggested that the algorithm can be scaled to out of sample data. Finally, the returns of the algorithms were found to generally outperform most mutual and hedge funds.*

Short Research Paper

Keywords – ***Machine Learning, Computerized Investing, Value Investing, Fundamental Analysis***

INTRODUCTION

A. Investing Background

There are 4 main categories for investing analysis: technical, fundamental, top-down, and bottom-up (Chen, 2019). It has been suggested that value investing, which implements fundamental analysis, can be the most effective form of investing in the long term if given enough discipline, skill, and time (Riley, 2010).

The underlying principle behind value investing is to purchase stocks when the investor believes it is undervalued by the market, so that its price will eventually appreciate to its minimum intrinsic value over time, in which it can be sold for a profit (Riley, 2010). An investor uses the company's fundamental data found in their income statement, balance sheet, and cashflow statements to estimate its intrinsic value.

A common method of estimating a company's intrinsic value is using the Discounted Cash Flow calculation, first popularized by John Burr Williams (1956) in *The Theory of Investment Value*. A company's valuation estimation can vary as investors make different judgements on the data.

Because value investing requires advanced financial knowledge and expertise that most individuals do not have, it would be useful to develop machine learning models that learn value investing so everyday individuals can access the algorithm and benefit from the investing technique.

B. Relevant Work

One of the most comprehensive related works is by Rasekhschaffe & Jones (2019). The researchers used multiple machine learning algorithms, ensembles, and feature generation on fundamental data to identify stocks that underperform and outperform the market. One area for further research is to discriminate between large outperformance and small outperformance to further enhance returns.

A similar work was conducted by Quah (2008). The work analyzed soft-computing models on picking stocks among the Dow Jones Industrial Average from 1995 to 2016. The researchers identified optimal training parameters and algorithm evaluation methods that achieved above average returns. However, only 30 companies were analyzed, meaning the data may not be representative of the entire feature space.

Another work was conducted by Sim, K., Gopalkrishnan, V., Phua, C., & Cong, G. (2014). The researchers used 3D Subspace modeling with stock fundamental data to generate models that can identify profitable stocks to purchase. The results suggested that certain models could consistently outperform Graham's method of value investing. The forecast horizons were defined on an annual basis, which many value investors consider too short because the observed price appreciations may have been a result of short term, irrational price movements rather than a return to intrinsic value.

C. Contributions

Below highlights the key contributions presented in this research that either advance or corroborate prior research.

1) Obtain high returns, relatively low volatility, and algorithm precision
2) Utilize data from a large time frame and a large, diverse selection of stocks
3) Establish prediction targets that are balanced and discriminate large outperformance from small outperformance
4) Build prediction targets that emphasize long term price change
5) Determine if the model can reliably predict the profitability of out of sample data

a. Compare precision of in sample data and out of sample data
b. In sample and out of sample data distribution comparison

EXPERIMENTAL SETUP

A. *Dataset Description & Preprocessing*

The dataset was obtained from gurufocus.com and is comprised of 30 years of fundamental data from 1990 to 2019 on the domestic Consumer Discretionary industry according to the S&P 500, the Russell 2000, and the Russell 1000 indexes. The data consists of approximately 200 companies and 4000 data entries. This industry was chosen for its large dataset size and because it is domestic.

The prediction targets for any given company at a certain year were either 0, 1, or 2. To create these labels, first, the return of each company three years after the present year was calculated. Then, the returns of all the companies in the dataset were separated into three percentiles, and 0 was assigned to the bottom third percentile, 1 was assigned to the middle third percentile, and 2 was assigned to the top third percentile. This allows the models to discriminate a large outperformance from a small outperformance, e.g. a company being in the 3rd percentile versus 2nd percentile. The dataset had missing values, so data affected by this were removed. The data were scaled proportionally between 0 and 1.

B. *Feature Selection, Feature Generation, and Supervised Learning*

To prevent overfitting, only a relatively small amount of fundamental data could be used as features to train the model. Based on a combination of references, including Quah (2018) and Buffet & Clark (2011), eleven features were selected: Month End Stock Price, PB Ratio, PE Ratio, YoY Rev. per Sh. Growth, Gross Margin, Net Margin %, Retained Earnings, ROE %, Total Current Assets, Current Ratio, and Capital Expenditure.

The researcher needed to determine how many years in each row of data used to train an algorithm should be given as context before the algorithm makes a prediction. This was done by adding the previous X-1 years to the subsequent row, where X is the number of years of context to be provided to an algorithm. For example, if 3 years of context is to be provided, then the data attributed to a company from 1990 and 1991 are concatenated to the data at 1992.

Features were generated by describing the percent change of each corresponding feature from the first year in a row from the last year.

C. *Data Visualization*

The dataset was visualized to understand the distribution of the features according to a 9 Fold Cross Validation technique, i.e. there was a different test set of graphs for each of the nine folds, each with 12 histograms for each feature and prediction target. This can be seen in Fig 1. Different colors were used to represent the test fold and the training folds.

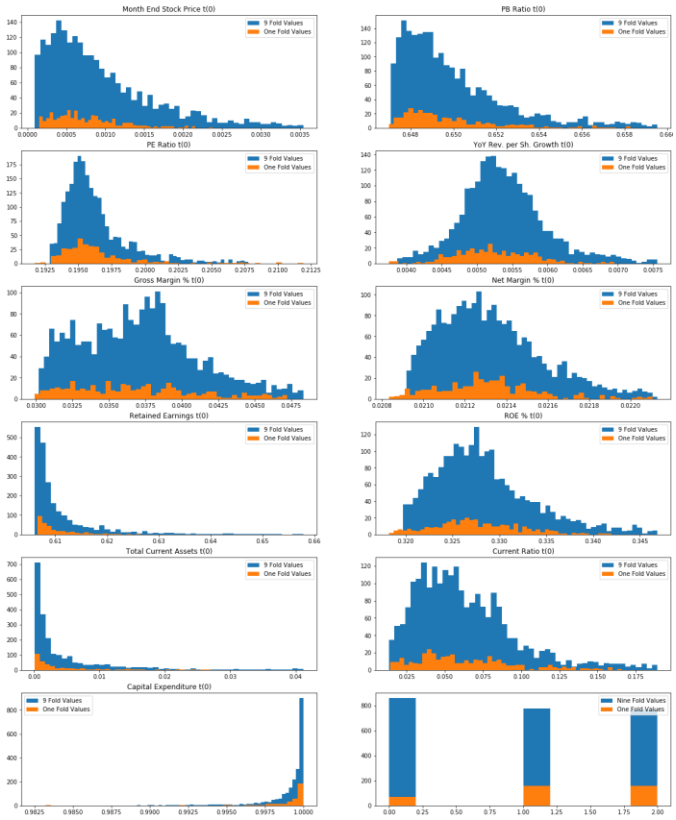

Fig 1. Data Visualization

D. *Data Structuring*

There were multiple ways the dataset was divided and multiple metrics used to evaluate the models. Below are the three ways the dataset was divided.

- Train Test Split

To create the training dataset, 89% of the data were randomly sampled using 42, 24, and 69 as the random seed states. The remaining data were used as test data. However, the training data which featured a company and year that overlapped with the test data (due to shifting the data to create context) were removed to avoid data leakage.

- 2013 Split & Train Test Split

All data before 2013 were treated as “in sample” data, and the train test split method was used. All data after 2013 were considered “out of sample” and were used for validation.

- 15 Year Train to 12 Year Test & Train Test Split

The 15 years between 1990 and 2004 were used for training the algorithms, and the remaining 12 years were used for testing the algorithms. The 15 years for training were also treated with train test split. To suggest that there is not

significant overfitting on the in sample data, the researcher can compare the algorithms' performances on the in sample test data with their performances on the in sample train data.

E. Testing Criteria

- Return

The most important metric is the return of the portfolio on a certain set of data. This is calculated by subtracting the original price from the final price, divided by the original price. The forecast horizon is 3 years, meaning the final price occurs 3 years after the original one.

- Volatility

Volatility was also measured by finding the interquartile range (IQR) of the companies' returns in a given set of data. Standard deviation was not used because it has been repeatedly suggested to not be applicable in the context of stocks, most notably due to its assumption that the return distribution must be normal, which is often times not the case (Adkins, 2020). IQR however, is appropriate for measuring the variation from a skewed data distribution.

- Precision 1 & Precision 2

Precision is a common metric in machine learning, defined as the number of true positives over all data points predicted to be positive. Precision 1 is the number of stocks that are class 2 over the number of stocks predicted as class 2. Precision 2 however is the number of stocks that are class 1 or 2 over the number of stocks predicted as class 2. Precision 2 calculates the percentage of predictions that have at least average return, while Precision 1 calculates that for above average return.

- Benchmark

A benchmark was also calculated for each year in the dataset to compare the algorithms with. The benchmark values were calculated by conducting the appropriate return, volatility, precision 1, and precision 2 calculations on the entire set of data in a certain year. This effectively represents a portfolio that does not have any rules when making an investment and is the market average.

F. Algorithms

Different algorithms were used to learn from this data. Those were Support Vector Machines (SVM), Random Forests (RF), AdaBoost (Ada), an Artificial Neural Network (ANN), and an ensemble of the first three algorithms. The ANN was found to be unsuitable for this dataset, so the results are not reported.

Grid search was used to optimize the SVM. The RF had 100 trees that were a maximum of 2 inner nodes deep. The Ada had 15 weak classifiers. The ensemble used simple voting of the three algorithms.

G. Softmax Ranking Methodology

An innovative method for an algorithm to make a prediction is to first assign a confidence value of each class to each stock by inputting the machine learning algorithm's activations, a set of vectors $z^{(3)}$, to the softmax activation function and output a new set of vectors $\sigma^{(3)}$ (Gao, Pavel, 2017).

$$\sigma(z^{(3)}) = \frac{e^{z_i}}{\sum_{i=0}^{2} e^{z_i}} \quad (1)$$

Next, find the 20th-to-last largest $\sigma_{(2)}$ value and create a set $\sigma_{(2)}^{(20)}$ of all $\sigma_{(2)}$ values larger than that.

$$\sigma_{(2)_{(-20)}} = \max(t \text{ such that } \#\{s \in \sigma \mid s \geq t\} = 20) \quad (2)$$

$$\sigma_{(2)}{}^{(20)} = \{t \in \sigma \mid t \geq \sigma_{(-20)}\} \quad (3)$$

This methodology was hypothesized to produce higher returns due to emphasizing investing in companies that the algorithm is more confident will have high returns.

EXPERIMENTAL RESULTS

A. Experiment 1: Determining Optimal Context Years

The first experiment involved determining the optimal number of years for each data instance. There existed a trade-off between the amount of data per instance and the number of instances available for training that had to be identified. After experimenting with a five combinations ranging from five years per instance, the return on investment (ROI) are reported in Fig 2.

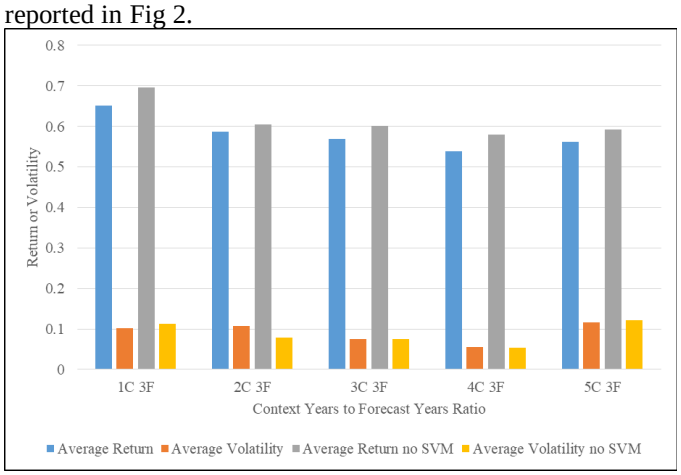

Fig. 2. Average Return & Volatility by Context to Forecast Year Ratio

The average ROI values were calculated by finding the mean of the returns of the algorithms on both the in sample (before 2013) test data and the out-of-sample (post 2013) test data. The blue column includes all four algorithms in the calculation of the mean, while the gray column excludes SVM because it was observed to have the worst performance.

Experiment 1 indicated that one year of context was optimal for the algorithms to make predictions. Figure 2 shows that the average of the algorithms returns is the highest for one year of context and three years of forecast. Having too many features with insufficient instances could perhaps make it difficult to learn the feature space. Additionally, the algorithms are not directly told which features are the same but at different time stamps.

B. Experiment 2: 15 Years (1990-2005) Train - 11 Years (2005-2016) Test

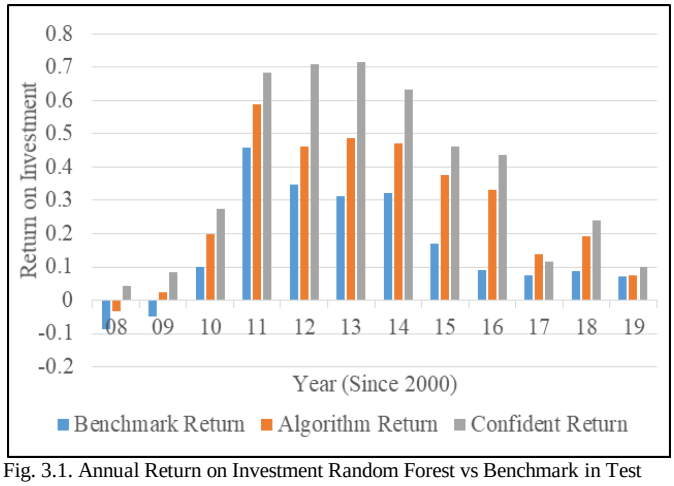

Fig. 3.1. Annual Return on Investment Random Forest vs Benchmark in Test Years (2008-2019)

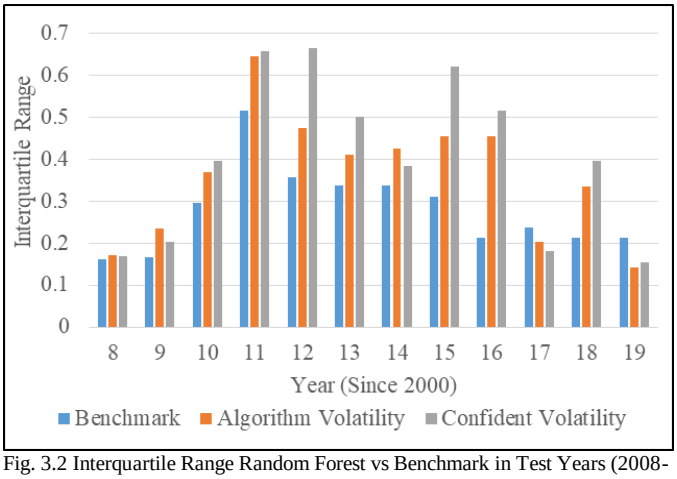

Fig. 3.2 Interquartile Range Random Forest vs Benchmark in Test Years (2008-2019)

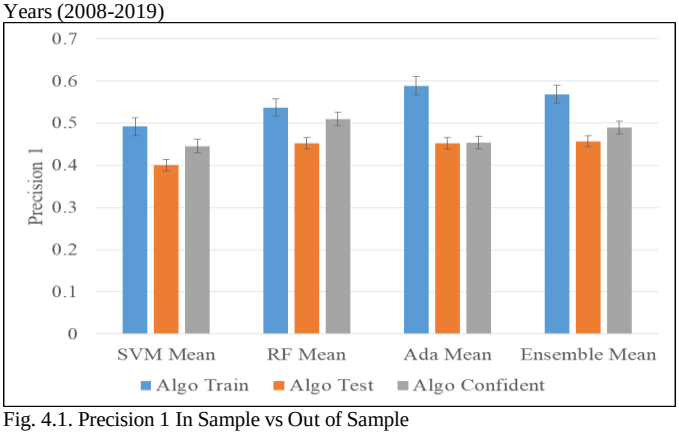

Fig. 4.1. Precision 1 In Sample vs Out of Sample

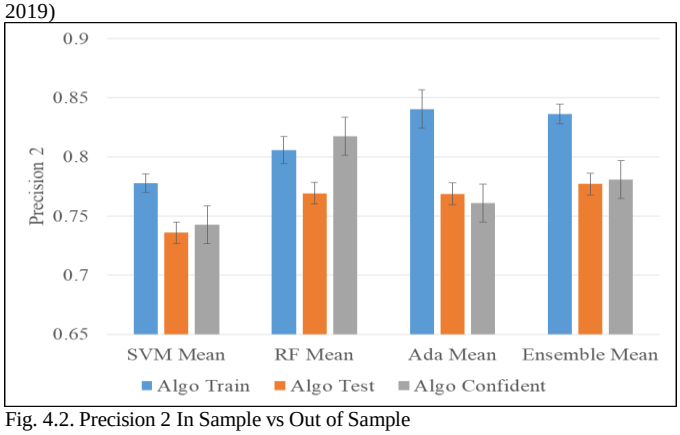

Fig. 4.2. Precision 2 In Sample vs Out of Sample

Fig. 3.1 and 3.2 showcase the RF returns and volatility obtained from the second experiment. Fig 4.1 and 4.2 are the precision 1 and precision 2 values obtained during experiment 2. Figure 4.1 compares the precision 1 of 4 algorithms. Each algorithm has 3 values tracked: the algorithm's precision on the training data, the algorithm's average precision on the out of sample test data, and the algorithm's average precision on the out of sample test data using confidence predictions. The same values were tracked in Figure 4.2 but using precision 2.

From Fig. 4.1, it can be seen that RF shows the least deviation from the training precision and both the in sample and out of sample testing precision. From Fig. 4.2, a similar conclusion can be drawn.

To compare the average ROI of each model across the twelve test years, Fig. 5 was created. The ROI of the algorithms using only their confident predictions was included.

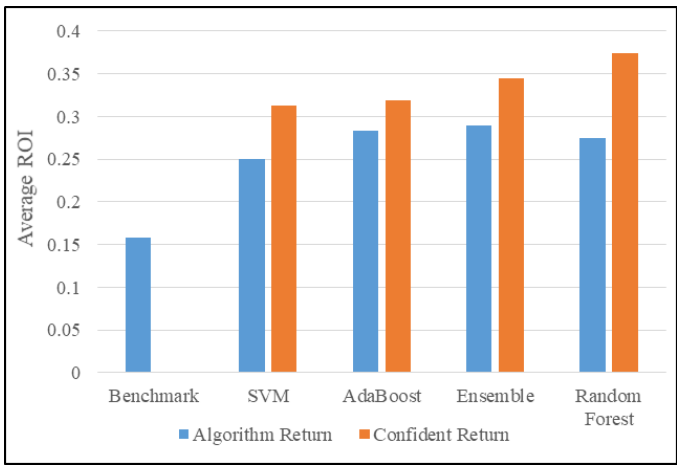

Fig. 5 Four Machine Learning Algorithms vs Benchmark ROI

For the random forest algorithm, the ROI by roughly 35%, for the ensemble roughly 21%, for Adaboost roughly 22%, for the Support Vector Machines roughly 25%. This unanimously shows the increased performance when using the confident prediction methodology.

Below is a table that contains hypothetical portfolio returns based on return on investment and interquartile range values of each algorithm observed in out of sample testing.

TABLE I. Gross Asset Worth Confident Predictions from 2005-2019

Year	Benchmark	SVM	RF	Ada	Ensemble
ROI Percent Change	246.6	1063.	1471.	841.9	1161.
ROI Annual Growth	**0.1003**	0.2077	**0.236**	0.1882	**0.2153**
IQR Percent Change	1165.	2018.	2291.	1821.	2101.
IQR Annual Growth	**0.2155**	0.2647	**0.2766**	0.2552	**0.2685**

Table 1 indicated that RF and ensemble were similar in performance and produced the best risk-reward ratios. Using confidence predictions, RF obtained a 23.60% gross annual return but a cumulative volatility exposure of approximately 28.77%. The ensemble had a 21.53% and return 26.19% volatility exposure respectively. The benchmark had 10.03% and 21.42% respectively.

Fig. 6 is visualization of each feature for 3 data distributions for each variable. The blue represents the feature distributions of the in sample training data. The orange represents those of the out of sample data. The green represents those of the out of sample data that any of the four algorithms predicted using the confidence ranking methodology.

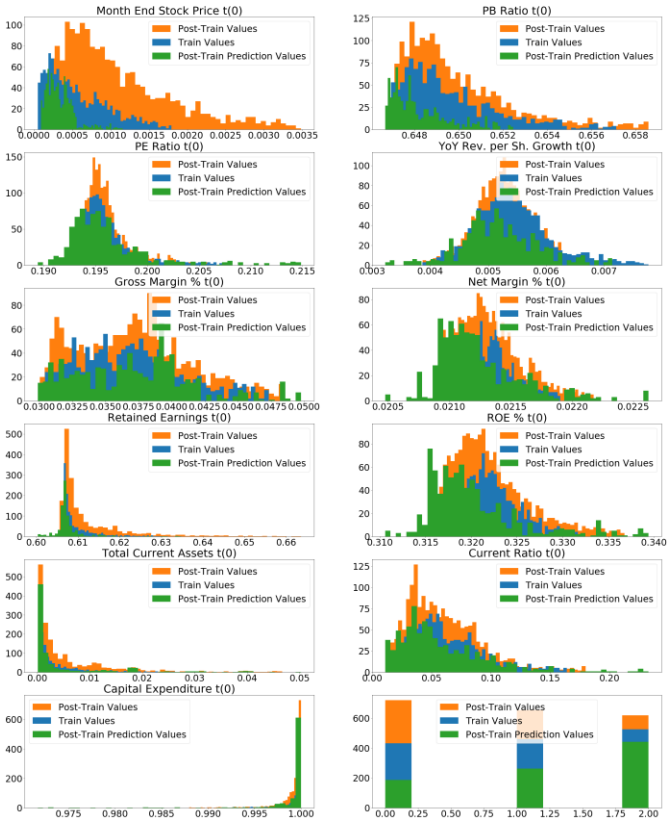

Fig. 6. Data Visualization of Train vs Out-of-Sample Data vs Out-of-Sample Data Predicted using Confidence Methodology

As seen in Fig. 6, the distribution of features from the companies predicted to be most profitable in the out of sample data tend to fall within the distribution of the training data. This may suggest that the algorithms are confident that companies in the future will be class 2 if the features are also similar to the data they were trained on. Out of sample data perhaps may be reliably and correctly predicted to be profitable investments if the algorithm is confident the company is class 2 and the features fall within the distribution of the original training data.

Fig. 7 below shows the relative feature importance of each variable as reported by the random forest algorithm. These relative feature importance values can be used to better understand what variables plays a larger role when assessing the favorability of an investment option. It can also be used to do further analysis, such as apply weights to calculations that relate to data analysis.

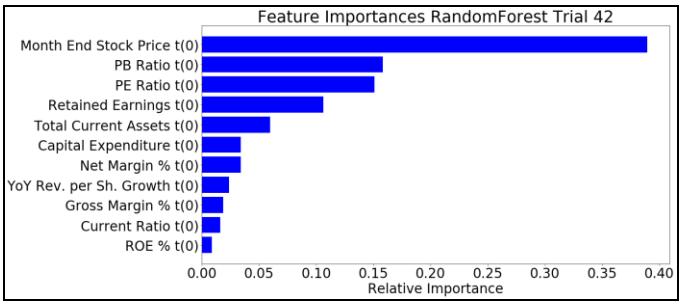

Fig. 7 Feature importance assigned using the random forest algorithm.

It can be seen that the most favored metrics relates to the price of the business, given that the favored variables are the "Month End Stock Price", "Price to Book Ratio", and "Price to Earnings Ratio".

One application of such data analysis is to determine the relationship between the algorithm's precision and the dissimilarity between the average of the out-of-sample data distribution with that of the training data of the algorithm, as depicted in Fig. 8.

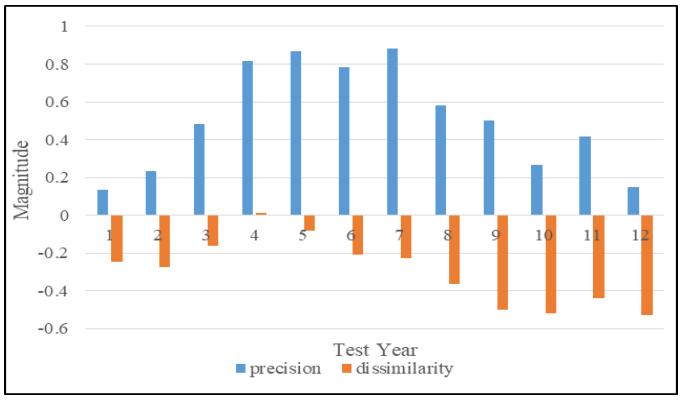

Fig. 8 Precision vs Weighted Dissimilarity for Random Forest

For each out of sample year and for each independent variable, the z-score between the training data distribution and the out of sample year data distribution was determined

using their means and standard deviations. The net z-score of the year was computed using the calculated z -core multiplied by a weight value associated with each variable, as shown in Fig. 6. Finally, the net weighted average z-score was compared with the precision for each year.

From Fig. 8, it can be seen that when there is smaller dissimilarity (signified by a smaller magnitude in dissimilarity), the precision of the model tends to increase. The converse is true. This implies that one can first calculate the dissimilarity between the data being used and the training data to gauge how confident they can be that the model will produce successful results.

Table 2 compares the machine learning algorithms to the best performing funds of the previous decade currently in operation. The gross returns of the RF and ensemble outperform the net of the hedge fund A. It is difficult to estimate the net return of the algorithm considering the intricacies of tax rules, so these results are approximations.

current year's data distribution with the training data's distribution. The random forest algorithm generally found valuation metrics like the "price to earning ratio" and "price to book value ratio" to be influential in its decision making process.

ACKNOWLEDGMENT

The researcher would like to thank his mentor at The Woodlands College Park High School for offering advice and support throughout the research. The project was self-funded.

TABLE II. Comparison between proposed machine learning algorithms and mutual and hedge funds in the real world.

Type	13F/Portfolio Date	# stocks	10-Y Average Return %	Q/Q Turnover %	Value ($Mil)
RF	N/A	20	**~23.60% (Gross)**	0%	N/A
Ensemble	N/A	20	**~21.53% (Gross)**	0%	N/A
SVM	N/A	20	**~20.77% (Gross)**	0%	N/A
Hedge Fund A	9/30/2019	24	**20.10% (Net)**	42%	3,380
Ada	N/A	20	**~18.83% (Gross)**	0%	N/A
Mutual Fund A	12/31/2019	29	16.10% (Net)	12%	3,339
Mutual Fund B	12/31/2019	31	14.80% (Net)	5%	2,623
Mutual Fund C	9/30/2019	350	14.70% (Net)	4%	24,309
Mutual Fund D	9/30/2019	144	14.60% (Net)	4%	7,506
Mutual Fund F	9/30/2019	48	14.50% (Net)	6%	8,135

Considering the data sampled in table 2 is the best performing mutual and hedge funds out of roughly 400 total around the world, this suggests that the algorithms generally outperforms human investors. It is reasonable to expect the algorithms can still have high net returns due to the three year time horizons that usually allows for more tax efficiency.

CONCLUSION

The researcher concluded that the random forest and ensemble are strong candidates for individuals who would like to utilize machine learning to help aid in picking stocks using a value investing approach. The algorithm was able to generally outperform top mutual funds for over twelve years, and evidence was gathered that the results are scalable to future years if the user calculates the dissimilarity between the

REFERENCES

[1] Adkins, Troy. "Calculating Volatility: A Simplified Approach." Investopedia, Investopedia, 25 Jan. 2020, www.investopedia.com/articles/basics/09/simplified-measuring-interpreting-volatility.asp.

[2] Williams, J. B. (1956). The theory of investment value. Amsterdam: North-Holland.Buffett, M., & Clark, D. (2011).

[3] Buffett, M., & Clark, D. (2011). Warren Buffett and the interpretation of financial statements the search for the company with a durable competitive advantage. London: Simon & Schuster.

[4] Chen, J. (2019, April 23). Investment Analysis: The Key to Sound Portfolio Management Strategy. Retrieved from

https://www.investopedia.com/terms/i/investment-analysis.asp

[5] Gao, B., & Pavel, L. (2017). On the properties of the softmax function with application in game theory and reinforcement learning. arXiv preprint arXiv:1704.00805.

[6] Rasekhschaffe, K. C., & Jones, R. C. (2019). Machine Learning for Stock Selection. Financial Analysts Journal, 75(3), 70–88. doi: 10.1080/0015198x.2019.1596678

[7] Quah, T.-S. (2008). DJIA stock selection assisted by neural network. Expert Systems with Applications, 35(1-2), 50–58. doi: 10.1016/j.eswa.2007.06.039

[8] Sim, K., Gopalkrishnan, V., Phua, C., & Cong, G. (2014). 3D Subspace Clustering for Value Investing. IEEE Intelligent Systems, 29(2), 52–59. doi: 10.1109/mis.2012.24

[9] Riley, E. (2010, June). Advisor Today. Advisor Today.