Quantitative Investing with Machine Learning

USC Marshall, FBE 551 Quantitative Finance

Suhas Sridhar

Table of Contents

Executive Summary

The aim of the project is to combine quantitative strategies and machine learning techniques to arrive at an ensemble strategy that can deliver higher returns with reduced volatility. To that end, we have explored assigning different ratios of weights to machine learning models and quantitative strategies focused on delivering higher returns and/or lower volatility. The comparison has been made against the buy & hold strategy of all stocks, all ETFs which is considered the benchmark performance. The key metric we've used for measuring performance is the Sharpe ratio.

On the modeling front, we used 11 years of data from 2000 to 2010 to train the model and predicted performance on the test data for the next 11 years from 2011 to 2022. We used XGBoost and lightGBM techniques for our modeling processes achieving an r-square of -0.3%.

We also explored multiple quantitative strategies including momentum, moving average, and different permutations and combinations of them.

We set our benchmark to buy-and-hold all stocks and ETFs over the past 10 years. After several iterations, we implemented our final strategy recommending an investment with 50% of the investment based on model predictions and 50% based on momentum strategy. This strategy beats the benchmark and we validated it by performing backtesting of the strategies. Although the negative R-square of the model indicates if the strategy is applied with the mean forecast the results would be better, if not the same.

In conclusion, we derived our motivation for this project from Alberto G. Rossi[4]. We believe this is a start and attempt to combine machine learning and quantitative strategies aimed toward achieving higher returns and minimizing investment risks during uncertain times. The poor results conform with the findings of Welch and Goyal [1]

Data Preparation

This project used the monthly "CRSP" dataset and "compustat" dataset that are obtained from Wharton Research Data Services. "CRSP" dataset contains monthly data on returns, prices, volumes, and share outstandings for different security types. The reason for using the monthly dataset instead of the daily dataset is because of transaction cost reduction. The "Compustat" data set contains annual data on key accounting variables, which includes income before extraordinary items (ib). The team merged the two datasets on Date and PERMNO number to get the full dataset that will be used in future steps. In order to better predict the rate of return, the team built a series of potential candidate variables. Ten variables were created in total. The following table shows the definition of each variable:

Variable	Description of Variable
"LAG_IB"	Three months rolling from lagged income before extraordinary items at the 'PERMNO' level.
"LAG_EP"	Three months rolling from lag E/P ratio at the 'PERMNO' level.
"LAG_PRC"	One month rolling from the lagged closing price at the 'PERMNO' level.
"LAG_VOL"	One month rolling from the lagged volume at the 'PERMNO' level.
"LAGRET"	One month rolling from the lagged rate of return at the 'PERMNO' level.
"MA"	Moving Average. Five-day rolling average from the lagged returns at the 'PERMONO' level
"VOLATILITY"	Five-month rolling standard deviation from the lagged returns at the 'PERMNO' level
"MOVING_VOL"	Five-month rolling average from the lagged volume at the 'PERMNO' level.
"MOMENTUM"	Mean of the rate of return during the T-2 to T-12 formation period.
"lagC"	One month rolling from lagged the number of consecutive positive/negative rates of returns at the 'PERMNO' level. An attempt to capture a string of positive/negative returns.

After the feature engineering process, the team used the ten variables in the machine learning process. Data was splitted from 2000 to 2010 as the training set and data from 2011 to 2022 as the testing set. Assuming that clients start investing in January 2011 and they would like to know the investment decision for the next month. The strategy started by using extreme gradient boosting (XGboost) and light gradient boosting (LGBM) to predict the rate of return for each month. We rolled the training set every month with new data as Figure 1. shown below. To illustrate, if we were to predict the rate of return in February 2011, we would use data from January 2000 to January 2011 to train models and get estimations. If we would like to predict the rate of return in March 2011, we would use data from January 2000 to February 2011 to train models, and so on… We averaged the prediction of two machine Learning methods as our final prediction rate of return for each month. Next step, we used the predicted rate of return to build the original investment strategy.

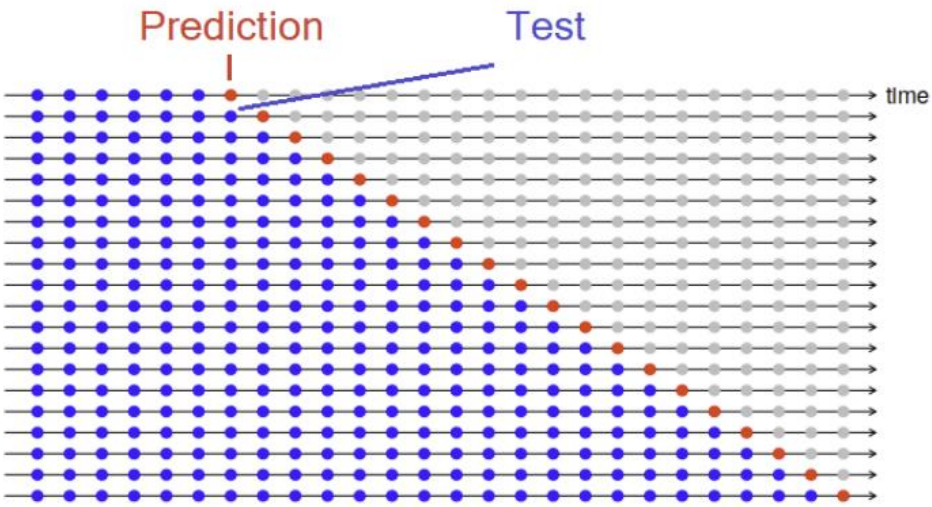

Figure 1. Rolling prediction

Strategy

We explored a strategy that was engineered by gaining inspiration from Lauri Nevasalmi[3], Alberto G. Rossi [4], and Factor Momentum strategies. Our strategy is especially influenced by the concept of factor momentum. The approach we chose was driven by the prospects of prediction through machine learning to improve returns.

Our strategy mainly consists of two parts which are combined to create an ensemble model. The first part of the strategy is to simply utilize momentum to determine which stocks to invest in. This is done by categorizing the stocks into quintiles based on momentum and then investing in the stocks within the fifth quintile (long highest returns) and short the first quintile (short low returns).

The second part of the strategy is to leverage machine learning models to predict returns on a rolling basis for a particular month. Our models draw upon all historic data up until the month the returns are being predicted for, not including the month itself. All of the variables mentioned in the table above are used as inputs for our models. Since the models utilize gradient boosting, it inherently assigns weights to the different variables according to how useful they are and then uses them to predict the expected return. Similar to the first part of the strategy, the stocks are divided into quintiles and we go long on the stocks in the fifth quintile and short those in the first. This cycle is repeated on a monthly basis at the beginning of each month.

We have evaluated our strategy under two distinct investing methods:

- Simple investing
    - We reinvest the same amount every month regardless of gains or losses and realize the returns and get accrual returns at the end of each month. (eg. always invest $100 at the beginning of the month)

- Compound investing
    - We reinvest the principal amount as well as any returns earned during the implementation of the strategy. (eg. invest $100 in the first month and if realized returns are $20, invest $120 in the subsequent month)

Interactive graphs can be found in our python notebook that can be used to better understand the returns and performance of this approach. These calculations do not take transaction costs or capital gains tax into account.

After an extensive review of the code to ensure that there are no data leaks, our model resulted in a slightly negative out-of-sample R^2 (-0.003) of -0.3%. Although on its own, this may not seem very promising, this highlights the potential for improvement of existing strategies by leveraging machine learning techniques.

Results

Statistics

After our machine learning gets finalized, we obtained the following statistics of the portfolio for a long-and-short method (buying the stocks in the highest return quintile and shorting stocks in the lowest return quintile from the model predictions) in Table 1 below. The strategy generates an average monthly return of 1.15% (13.8% annual) and has an annualized Sharpe ratio of 1.71. With a high t-test score, it can be concluded that the test statistics for our strategy are statistically significant.

t-stats: 6.91

	RET
count	137.000000
mean	0.015380
std	0.026065
min	-0.043633
25%	-0.003428
50%	0.013178
75%	0.029129
max	0.099102
Sharpe	1.711741

Table 1. Test statistics for machine learning strategy

By comparing the 100% machine learning strategy to a 100% momentum long-and-short strategy (buying the stocks in the highest momentum quintile and shorting stocks in the lowest momentum quintile) and a 50% machine learning, 50% momentum strategy in Table 2 and Table 3 below, it clearly demonstrates that the 100% machine learning beats the momentum strategy on the average rate of return and Sharpe ratios. And for a 50-50 machine learning and momentum strategy, although it has a similar rate of return with a simple momentum strategy, it has a better performance in reducing the variation/risks which are reflected in the higher Sharpe ratio. Both strategies have high s-stats that are statistically significant for comparison.

t-stats: 2.22

	RET
count	137.000000
mean	0.011704
std	0.061650
min	-0.208805
25%	-0.016021
50%	0.011773
75%	0.042239
max	0.222944
Sharpe	0.517148

Table 2. Momentum Strategy

t-stats: 4.71

	RET
count	137.000000
mean	0.010222
std	0.025378
min	-0.070953
25%	-0.002078
50%	0.009851
75%	0.024779
max	0.091284
Sharpe	1.054071

Table 3. 50% Machine Learning + 50% Momentum

To have a better understanding of the test results, we have also set 2 benchmarks which are buy-and-hold strategies for all stocks or ETFs, and their test statistics are shown in Table 4 and Table 5 below. The two benchmarks have much lower monthly returns than any other strategies that we compared above. In addition, both portfolios have low Sharpe ratios which indicate vulnerability under market fluctuations.

count	138.000000
mean	0.009055
std	0.056338
min	-0.223813
25%	-0.017831
50%	0.010659
75%	0.039548
max	0.205248
sharpe	0.556758
t-test	1.888060

Table 4. Statistics for benchmark 1: stocks

count	138.000000
mean	0.003767
std	0.031682
min	-0.142762
25%	-0.010625
50%	0.006672
75%	0.020208
max	0.100034
sharpe	0.411862
t-test	1.396692

Table 5. Statistics for benchmark 2: ETFs

Backtest

For our final strategy, we have decided on the implementation of putting 50% of investments in the long-and-short strategy and 50% into the momentum long-and-short strategy as a more conservative approach. We have done a backtest for our strategy and the buy-and-hold strategy for stocks and ETFs (benchmark) over the past 10 years as shown in Figure 1 below. While our strategy does not seem to have a significant advantage in the rate of return compared with the benchmarks, it is clear that our strategy avoids some large drawdowns and therefore has a higher Sharpe ratio.

Figure 1. RET comparison among our strategy and benchmark

In addition to that, by comparing our strategy with a simple momentum strategy below, it can also be seen that our strategy has a lesser standard deviation compared to Momentum while not sacrificing returns too much.

Figure 2. Standard deviation comparison between our strategy and momentum strategy

To help better understand the results of our final strategy. We assumed a $100 initial investment into the benchmark portfolio (all stocks or ETFs) and applied a compounding interest rate to get their dollar values at the end of a 10-year investment period. For our strategy, we explored two methods of investment. One is to invest a fixed amount of $100 at the beginning of each month and sell all at the end of each month to get the cumulative earnings from the 10-year test period (shown as strategy cons. investment in Figure 3). Another method is to use the idea of

compounding: invest $100 at the beginning of the first month, sell all at the end of that month, and use all the returns to invest in the next month (shown as a strategy with compounding in Figure 3). A comparison of how much $100 would be at each period has been plotted below in Figure 3. While unable to beat momentum, we have achieved a high Sharpe ratio and it can be observed that at the end of the testing period our compounding strategy catches up and ends up with a similar valuation of momentum strategy.

Figure 3. Money return of our strategy vs. benchmarks with $100 investment

If we become a little more aggressive and adjust our strategy to put 100% weight on the model predictions (0% momentum) and adjust the strategy to buy only instead of buy-and-short, the $100 investment for the new strategy using compounding becomes over $1,000 in 10 years, which clearly beats any other methods as shown in Figure 4 below. In the meantime, a compounding momentum strategy gives us about $500 at the end of the test period, which is the second-best strategy in our comparison. This would mean there is merit to investing based on moving averages of returns rather than model predictions, as the model performs worse than the mean. However, this work can be built upon to generate models that can have a positive predictive capability.

Figure 4. Money return of 100% machine learning (long), momentum, and benchmarks with a

$100 investment

Finally, we adjusted our implementation to a long-short method with a 100% weight on the machine learning model and compared the money return with the momentum strategy and

benchmarks in Figure 5 below. Although not as good as the long-only method above, a long-short method has also given us a fairly high return in 10 years.

Figure 5. Money return of 100% machine learning (long-short), momentum, and benchmarks with a $100 investment

Conclusion

The project aimed to build the best strategy to predict the rate of return. The team merged the two datasets, crsp, and compustat, to get a completed dataset as the initial step. Furthermore, we created ten variables in the feature engineering process, used them in two machine learning processes, extreme gradient boosting and light gradient boosting, and averaged the two predictions from machine learning models as our final predicted rate of return. Next, the team deployed a 50-50 machine learning and momentum strategy and obtained a Sharpe ratio of 1.05 and a mean of monthly return 1.02% as result. The results from this project indicate that an investment strategy based on average returns combined with a momentum strategy might reduce volatility in the investment portfolio. While the team attempted to improve upon existing strategies by leveraging machine learning models, the final results were unsuccessful. However, if time allows, this work can be built upon by leveraging strategies that were successful in leveraging Machine Learning models such as Lauri Nevasalmi [3] and Gu, Shihao and Kelly, Bryan T. and Xiu, Dacheng [2].

Appendix

[1] Ivo Welch, Amit Goyal, A Comprehensive Look at The Empirical Performance of Equity Premium Prediction, The Review of Financial Studies, Volume 21, Issue 4, July 2008, Pages 1455–1508,

[2] Gu, Shihao and Kelly, Bryan T. and Xiu, Dacheng, Empirical Asset Pricing via Machine Learning (September 13, 2019). Chicago Booth Research Paper No. 18-04, 31st Australasian Finance and Banking Conference 2018, Yale ICF Working Paper No. 2018-09

[3] Lauri Nevasalmi Forecasting multinomial stock returns using machine learning methods, The Journal of Finance and Data Science,

[4] Alberto G. Rossi. Predicting Stock Market Returns with Machine Learning. University of Maryland August 21, 2018