Week 9 Progress

# Breakdown of Performance by Time Period

Last time we discussed showing a breakdown of trading bot performance by different time periods.

## Breakdown of Performance by Year

The height of each bar represents the difference in percent of principle for the given year between the trading bot and the SNP 500. For example, if the trading bot started at $100 and ended at $200, and the SNP started at $300 and ended at $150, the difference would be 1.5.

As you can see from the chart, the trading bot is outperformed by the SNP eight of the 15 years, and the trading bot outperforms the SNP seven of the 15 years. However, when the trading bot outperforms the SNP, it is by a larger margin on average than when the SNP outperforms the trading bot. I included the actual trading plot because I think they are interesting to look at side by side.

## Chart, line chart Description automatically generatedChart, waterfall chart Description automatically generated

## Breakdown of Performance by Month

In addition to comparing the performance of the trading bot and the SNP each year, I compared the performance for each month. The heigh of each bar represents the sum of the one-month differences in percentage of principle for a particular month over 15 years. The metric used for the yearly comparison is found for each one-month chunk of time, and the differences are aggregated by month of the year.

Chart, waterfall chart

Description automatically generatedTable

Description automatically generatedThe bar chart shows better performance of the trading bot in the middle of year. However, I wonder how much of this trend is just due to random chance. The table aggregated to produce this data is chaotic.

Source Data

# Model Analysis

I thought that maybe I could improve the performance of the model by using an LSTM because of its effectiveness in processing Time Series data. I had already tried using the LSTM as a classifier, but I decided to try to use it to build a regressor. After lots and lots of tuning, I arrived at a model that achieved similar performance to the Random Forest Regressor used in the Trading bot above. However, the LSTM took much longer to train.

## Chart, text Description automatically generatedA picture containing table Description automatically generatedLSTM

I tried many different network architectures for this model. This one seemed to yield consistent results. One very peculiar thing that I haven’t seen before is a better model performance on the testing set than the training set. I saw this every time I trained this model – with completely random testing sets. The testing MSE is always somewhere around 0.004. This equates to a RMSE of around 0.06 which is very big considering 0.01 is a very large change in price for one week.

## Random Forest

I was running the random forest model in my colab to have a comparison for the LSTM I was working on. I figured I might as well investigate the effects of changing the hyperparameters in the random forest. The heat map on the left shows a grid where the color indicates the MSE. The heat map on the right shows a grid where the color indicates High-Low accuracy. High-low accuracy refers to the percent of the time the model claims a stock will go up and it goes up or the model claims a stock will go down and it goes down. In both grids, yellow indicates a better model.

I found these results somewhat disappointing. It seemed like varying the tree size and tree maximum depth in the random forest had nearly no effect on its performance. You can see that there is no recognizable trend in the plots below. This seems to indicate that we should just choose a small number of estimators, and a small max depth to optimize training time.

Chart

Description automatically generated with medium confidenceA picture containing chart

Description automatically generatedIt is also evident from the heat map on the left that the performance of the random forest is comparable if not a little bit better than the performance of the LSTM in terms of Mean Squared Error. The best random forest performance boasted a MSE of around 0.003 which means the average weekly estimate is still off by around 0.054 on average.