Project 8: Strategy Evaluation

Syed Muhammad Husainie (903965504)

ABSTRACT

The Strategy Evaluation project brings together the fundamental concepts and strategies explored in previous projects for this course. The first part of the project aimed to create a set of manual rules for predicting future returns. The second half involved the use of Classification and Reinforcement learners to predict returns, through machine learning, data discretization, and fine tuning.

1. INTRODUCTION

The goal of this project was to develop a strategy to predict stocks and develop a profitable portfolio using two approaches – manual rules and classification learner. Before working on this assignment, my personal expectations were that the manual strategy compared to a classification learner would be difficult to improve by a significant amount, however, by applying simple rules on the different indicators such as Bollinger Band Values etc. to signal buy or sell conditions, it would likely be easier to tune and monitor improvements for it.

2. INDICATOR OVERVIEW

This project incorporated the use of five indicators. These were, Simple Moving Average (SMA), Bollinger Bands (BB), Momentum, K-Smooth, and Moving Average Convergence Divergence (MACD).

The SMA, used to smooth price volatility, was obtained by taking the rolling mean of the normalized prices over a fine-tuned window of 20-days. The results were discretized into 5 bins using the pandas qcut method (discussed more in the discretization section below), to generate meaningful rules for learning.

$$\mathrm{SMA} = \frac{(\mathrm{Price}\ 1 + \mathrm{Price}\ 2 + \mathrm{Price}\ 3 \ldots + \mathrm{Price}\ 12)}{\mathrm{n}}\text{, where } \mathrm{n} = 12 \tag{1}$$

In P6, Bollinger Band upper and lower bounds were calculated, in addition to the Bollinger Band value. On a side note, it was unclear why a single vector was needed and whether these two values could be used separately. During this project, the Bollinger Bands function in indicators.py was updated to calculate the value within the same function and combine the bands using a simple formula to generate a single vector:

$$\text{BB} = \text{BB Value} \times (\text{Upper Band} - \text{Lower Band}) \tag{2}$$

Traditionally, BB is used to analyze the volatility in prices, with high gaps in bands indicating more volatility. Using equation 1 essentially means the BB value was multiplied by 2 standard deviations. In short, this metric helped analyze volatility within the data more effectively. This BB value generated meaningful rules and values for both Manual and Strategy learners.

The momentum measures the state of change in the price of a stock and it was measured by comparing the current price to the past stock price:

$$\text{Momentum} = \frac{\text{current price}}{\text{price from x days ago}} - 1 \tag{3}$$

The stochastic oscillator compares the current price to the high and low prices in the past window. Traditionally, if the value passes 80% it indicates that the stock is overbought, whereas under 20% indicates that it is oversold:

$$\text{Stochastic Oscillator} = \frac{\text{current price} - \text{low price for window}}{\text{high price for window} - \text{low price for window}} \times 100 \tag{4}$$

The last indicator, MACD, is used to identify stock momentum and trend reversals, divergence, and direction of stock buy or sell trends. This is traditionally done by comparing the relationship between the exponential moving averages over a 12 and 26 period. To combine the MACD line, signal, and histogram (difference between MACD line and signal), an untraditional formula was applied as follows:

$$\text{Combined MACD} = \frac{\text{MACD Line}}{\text{MACD Histogram}} \times \frac{\text{Signal Line}}{\text{MACD Histogram}} \tag{5}$$

The formula went through several updates, starting from simple multiplication of the histogram to the line and signal, followed by addition. This did not help generate positive cumulative returns for the manual rules. As a result, it was then updated to add the two ratios instead of multiplying them. This generated meaningful predictions but did not perform up to mark for some symbols e.g., sine fast noise. The last update was changing addition to multiplication, and this improved the performance drastically. It is imperative to say that this only worked with discretized data due to the instability of results when MACD histogram equals 0 (spoken about more in the next section).

3. MANUAL STRATEGY

3.1 Discretization of data

Before applying the manual and classification learner strategies, the data was discretized with the help of Pandas qcut method. Initially, an error message was being displayed as qcut could not generate distinct bins for K% and momentum indicators using the ML4T-200 data. The resulting thresholds were then converted to integer values to further simplify data, and any infinite values were set to zero. Setting infinite to zero was important as it helped improve interpretability of negative outliers although it lost positive outliers. The manual strategy used accounted for this with its rules.

There were several justifications for discretizing the data. Firstly, when the data is not discretized, it can involve many complex relationships adding to its non-linearity and overall complexity over time. For example, the indicators used can never predict unforeseen events. Discretization can prevent capturing this volatility from in-sample data. In essence, since the task came down to whether we need to buy, hold, or sell a stock, smoothing out noise, and making it much more "readable" to the model predictors helped prevent overfitting. The figure below shows that with the required impact and commission, the Random Forest learner performs much better with discretization (first row of charts) compared to without it (second row of charts). It is noteworthy to mention that with 0 impact and commission, the strategy learner outperforms the manual strategy in all conditions (figure 1 manual strategy rules were adjusted for non-discretized values).

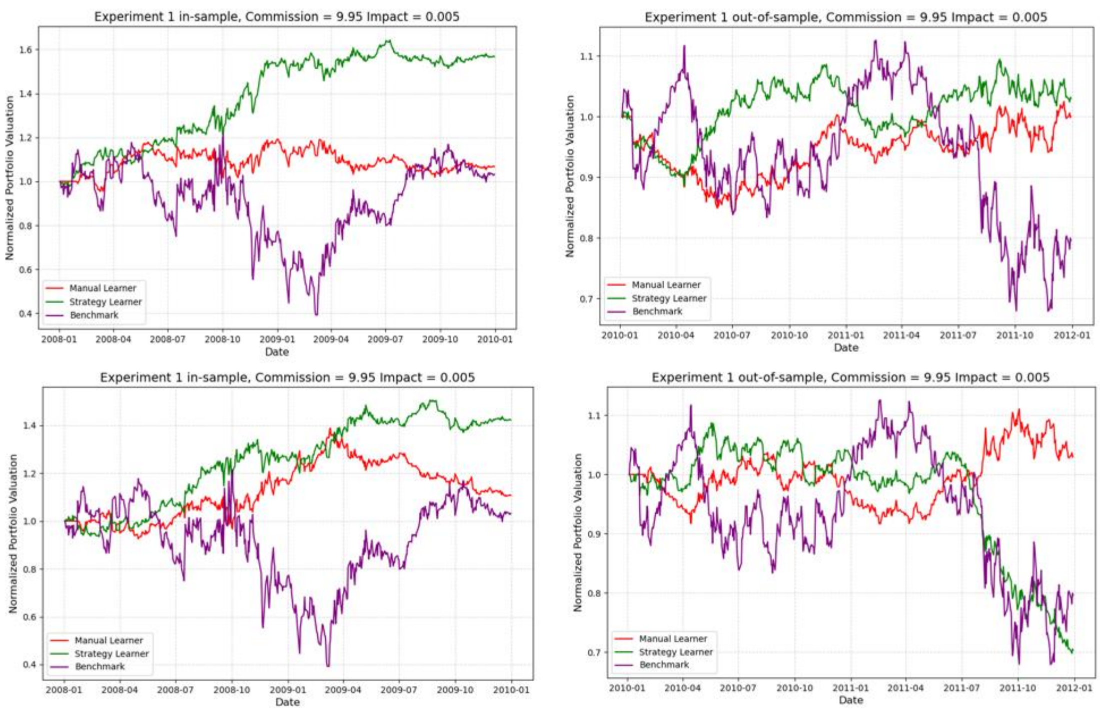


***Figure 1*:** Discretized data (first row of charts) vs. un-discretized data (second row of charts)

3.2 Manual Strategy Rules

The manual strategy involved creating a set of rules to train the data over an in-sample period and generate output for the out-of-sample period using the same rules. The buy and sell algorithms/ rules used were as follows:

1. Only **sell** if net holdings are above the minimum limit of -1000 stocks held at a time and only **buy** if net holdings are below the maximum limit of 1000 stocks.
2. If condition 1 is met, check if MACD OR Momentum is above a threshold of 3 (discretized data). If so, this would indicate a buy. If the MACD OR Momentum is below 1, this would indicate a sell condition.
3. If conditions 1 and 2 are met, check the previous combined BB value and current BB value. If the previous BB is lower than previous SMA AND current BB is higher than current SMA, this would be a buy signal. Moreover, if the previous BB was equal to 0, this would also make (3) a buy signal. Conversely, a previous BB above previous SMA AND current

BB below current SMA, OR just BB equal to 4 (upper level) signals a sell for (3).

4. If conditions 1,2 are met, and 3 is not met, a fourth condition is also used which is to check the K% threshold. If the current K% value is below 1, this would indicate a buy, whereas if the value is above 3, this would indicate a sell position.

This was proven to be an effective strategy as indicated in figure 2 below. Condition 2 above was effective as it used momentum and stock price trends to monitor sudden and/ or drastic changes. Condition 3 helped understand oversold or undersold conditions by comparing the adjusted BB to the normalized and/or discretized SMA values. If the trend showed that BB value was going above the SMA value, this would indicate an overbought position and vice versa. Furthermore, specifying the BB value thresholds also helped in controlling these signals as very high BB values show stock overbuying and vice versa. The last condition helped in effectively buy or sell based on current price comparisons to previous highs and lows. For the discretized data, this was limited to 1 and 3 to match the traditional 20% and 80% threshold values. Also, comparing figure 1,2, and 3, the manual strategy results seem slightly different, which is attributed to model output variation between runs as the data for each plot was generated separately. This was done on purpose to evaluate the model's volatility in results. Figure 2 compares the results for the in-sample data period. The results showed a consistent over-performance of the stock compared to the benchmark (just buying 1000 stocks and holding that position till the end). This highlights the effectiveness of the manual rules as it should only buy or sell to improve the portfolio values. Figure 3 is the out-of-sample results and comparison to the benchmark and shows a largely negative return, except at the end where the portfolio showed a net positive return compared to the benchmark. The manual strategy data was calculated at commission of $9.95 and impact of 0.005.

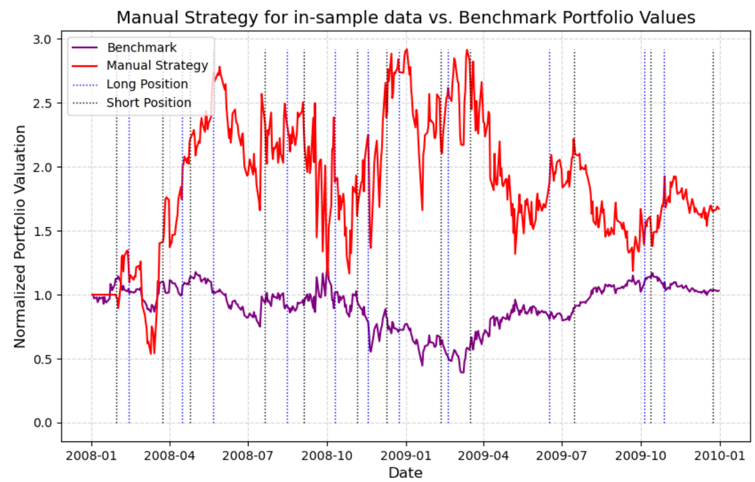


***Figure 2*:** Manual Strategy vs. Benchmark for in-sample discretized data

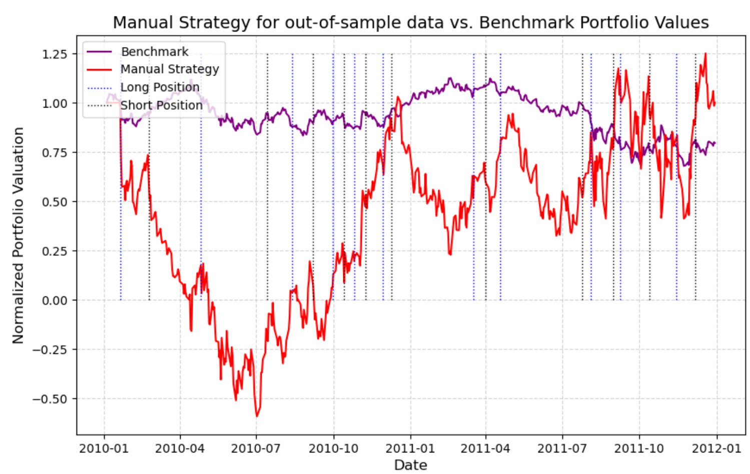


***Figure 3*:** Manual Strategy vs. Benchmark for out-of-sample discretized data

***Table 1*:** In-sample and out-of-sample performance metrics for manual strategy (raw values)

Metric	In-sample	Out-of-sample
Cumulative Return	0.671800	0.000780
Average Daily Return	0.005238	0.093215
Standard Deviation of Daily Returns	0.094289	1.537198
Sharpe Ratio	0.881881	0.962627

To summarize, the performance of the manual rules, despite performing well on in sample data, failed to generate on-par meaningful cumulative returns for out-of-sample. The other metrics were also below par including very high daily return deviation for out-of-sample. This was likely due to the rules capturing complex patterns unrelated to the actual indicators, thus causing overfitting.

4. STRATEGY LEARNER

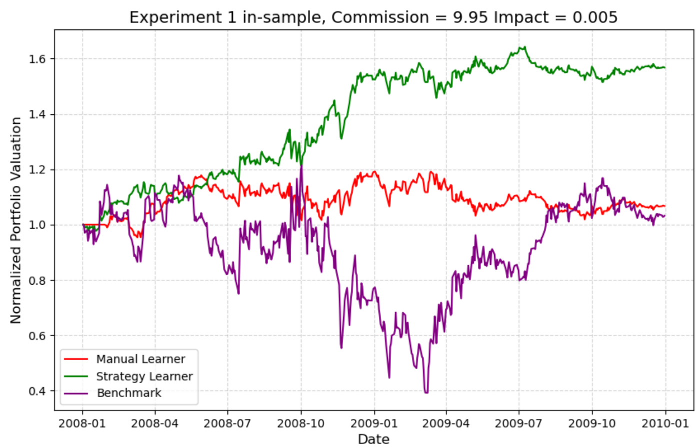
To compare and potentially improve the out-of-sample results, a random tree bag learner was used for this project. The goal was to train the learner to understand the relationship between the discretized indicators data (explained in section 3) and buy/ sell signals (return y values). To generate training return y values, the future in-sample daily returns were used. The returns were adjusted for impact on the price by multiplying with a conservative factor of 4 times the impact. For positive returns above a y-buy threshold, a long signal (value of 1) was set. For negative returns below a y-sell threshold, a -1 indicating a short signal was set. For all other cases, a value of 0 was set indicating a do-nothing command. The learner was then trained using the discretized indicators and the predictions from this train data were used to generate a long or short signal. Essentially, if y equaled 1 for the previous day and net holdings were below the 1000 threshold, this would indicate a long signal for the present day for stocks with a value matching the upper threshold. For shorting the stock, the exact opposite strategy was used. The goal was to either be in the long or short position until the very end where all stocks held would be cashed. For tuning the hyperparameters, a simple for loop with 20 iterations was used to measure cumulative returns for each tuned parameter. Moreover, the random sampling in bag learner was updated to k-fold cross validation, to preserve the data's temporal pattern and predictions were calculated using mode. The table below highlights the hyperparameters that were tuned, and the final values found:

Table 2: Hyperparameter tuning, results, and sensitivity for cumulative return

Hyperparameter	Tuning Range	Final parameter	Return Sensitivity
Indicator lookback window	5 – 50, increased in increments of 1	20	High
Return vector y look ahead window	5 – 50, increased in increments of 1	15	High
Y-sell threshold	0 – 0.04 increased in increments of 0.002 and finely tuned best output	-0.025	Very high
Y-buy threshold	0 – 0.04 increased in increments of 0.002 and finely tuned best output	0.0155	Very high
Discretization bins	2-20, increased in increments of 1	5	Very high
Leaf size	5-100, increased in increments of 2	30	Medium
Number of bags	20-120, increased in increments of 2	103	Medium

5. EXPERIMENT 1

This experiment was conducted to give a comparison of in-sample and out-of-sample performance for the manual and strategy learners. The results were compared to the benchmark. The impact was set to 0.005 and commission was set to $9.95. This was done to mimic real life scenarios. The initial hypothesis was that the strategy-based learner would provide more meaningful predictions without overfitting on training data.



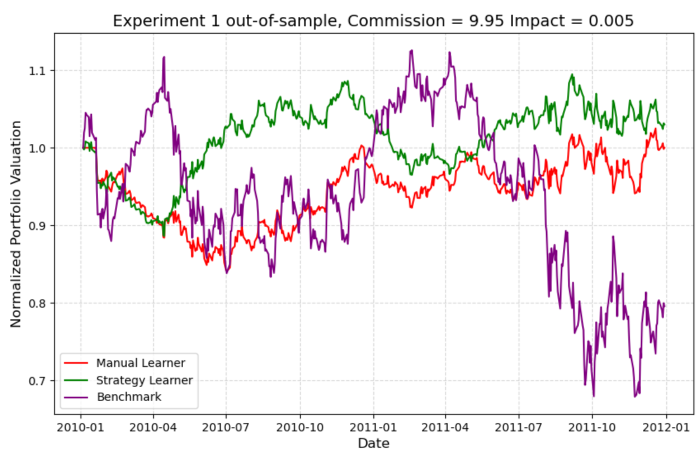


***Figure 4*:** Manual Strategy vs. Strategy Learner vs. Benchmark for in and out-of-sample data

The results from figure 4 showed that the strategy learner performed better than the manual strategy-based model throughout the out-of-sample dates. This was likely due to how finely tuned the rules were for manual strategy which may have caused overfitting. However, comparing the benchmark to figure 2, it is also evident that the benchmark has high deviation in its results showing unpredictability. The model was run several times and gave very consistent results for in sample and out of sample for the strategy learner, however, this was not the case for the manual strategy, which showed high deviation in out of sample due to overfitting. Furthermore, the strategy learner prevents high losses on any day, indicating that it is not likely a more robust predictive model.

(Something to note is that figure 2 data did not match figure 4 data. I believe this relates to how the data was called within the function and the volatility within it. Nevertheless, the cumulative returns were similar.)

6. EXPERIMENT 2

This experiment consisted of setting the impact at different values and generating the portfolio using in-sample data for the strategy learner. Before running the test, it was hypothesized that increasing the impact would lead to lower cumulative returns and performance metrics due to the model not taking risk for potential low returns on a day. The metrics used for testing were Sharpe ratio, which is a measure of the risk adjusted return, and the 100-day rolling/ look back return which helps track rewards as well as steadiness in the stock valuation. The results are shown below:

Table 3: Performance metrics at different impact values (raw values)

Metric	Sharpe ratio	100 day rolling return
Impact 0	2.049327	0.076151
Impact 0.005	0.953533	0.027191
Impact 0.01	0.680998	0.030486
Impact 0.02	0.354619	0.035270

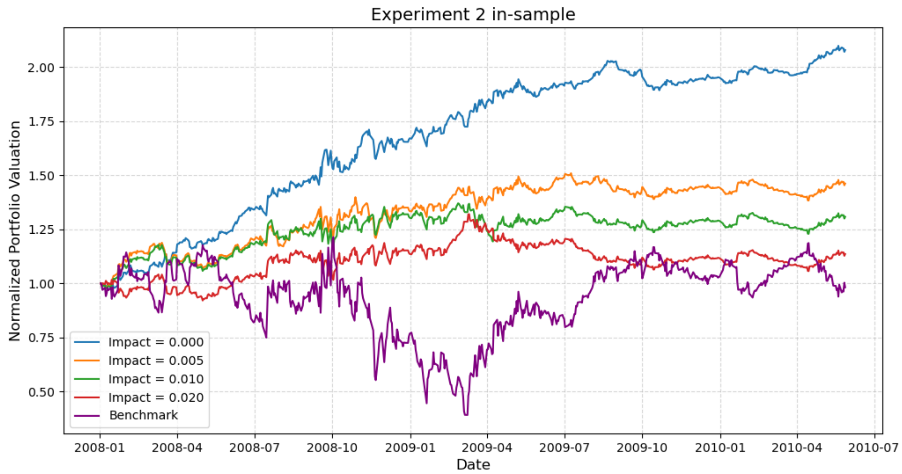


***Figure 5*:** Strategy Learner results at different impact values for in-sample data

The results were in line with the hypothesis. When the impact is 0, the results were very high, and because of less overfitting (verified through out-of-sample testing), the out-of-sample results were promising. However, as the impact increased, the model performance dropped progressively as indicated by the Sharpe ratio. At 0.005, the model still performed fairly reasonably well, indicated by the Sharpe ratio approaching 1 and positive 100 day rolling return. The results further indicate that more volatility in the market through higher impact, leads to less model efficiency in providing positive portfolio values. Nevertheless, by accounting for the impact within the strategy learner, the model still maintained less volatility and positive net returns, exceeding the benchmark performance.

7. CONCLUSIONS

Herein, a random tree classification learner shows high potential to learn effectively and generate meaningful results; compared to a manual approach. To further enhance its performance, the learner can also account for commission costs. Furthermore, other indicators can also be tested and included to measure the viability for this application. Lastly, it is important to note that the learner works well with most but not all symbols, e.g., AAPL when impact is added. It would be beneficial to see why this happens and how it can be accounted for in future research.