CS7646 Machine Learning for Trading
Fall 2024
Project 8: Strategy Evaluation

Shawn Lee
slee3707@gatech.edu

Abstract—This report investigates the in-sample and out-of-sample trading performance of a manual strategy and a classification-based learner, both built using three technical indicators. We also present the effects of different values of market impact on in-sample trading behavior of the classification-based learner.

1 INTRODUCTION

This report consists of 4 main sections: 1) Indicators, 2) Manual Strategy, 3) Strategy Learner and 4) Experiments.

In the first section, we introduce the three technical indicators used to devise our manual strategy and strategy learner: Bollinger Bands, Relative Strength Index and Stochastic Oscillator.

In the second section, we present our manual strategy that combines the indicators to create trading signals. We compare our performance against a buy-and-hold benchmark strategy over the in-sample and out-of-sample time periods.

In the third section, we present how we framed the trading problem as a learning problem for our strategy learner, and the values of the hyperparameters used.

Finally, we perform 2 experiments: one to pit the manual strategy against the strategy learner, and another to investigate the effects of different values of market impact on in-sample trading behavior of the strategy learner.

2 INDICATORS

A combination of volatility and momentum indicators were used. The parameters (e.g. lookback window) used to generate the indicators were not optimized. Each indicator returns a single real results vector that can be interpreted as actionable buy/sell signals.

2.1 Bollinger Bands

Bollinger bands help traders evaluate a stock's price action and volatility (Charles Schwab, n.d.) by providing a dynamic range that adapts to recent price movements and volatility (Thompson, 2024). The bands consist of three lines:

- **Middle Band**: Typically a 20-Day Simple Moving Average (SMA).
- **Upper Band**: Plotted 2 standard deviations above the middle band.
- **Lower Band**: Plotted 2 standard deviations below the middle band.

To obtain an actionable buy/sell signal, we calculate the Bollinger Band Percent (BBP) (Barchart, n.d.) using the formula: $BBP = (Price - Lower\ Band)\ /\ (Upper\ Band - Lower\ Band)$

When BBP > 1, price is above the upper band, representing overbought conditions. When BBP < 0, price is below the lower band, representing oversold conditions. Since we use past price data to construct the bands, Bollinger Bands are lagging indicators (Thompson, 2024). We use BBP in conjunction with leading indicators, such as RSI, to add confluence to overbought/oversold signals.

2.2 Relative Strength Index (RSI)

RSI is a momentum oscillator (Fernando, 2024) computed based on price changes directly, with values ranging from 0 to 100. It is mainly used to identify overbought or oversold conditions and is typically calculated over a rolling window of 14 days. RSI is calculated with the formula: $RSI = 100 - 100\ /\ (1 + (Average\ Gain\ /\ Average\ Loss))$

RSI > 70 signals that the stock may be overbought, and a correction may soon occur. Conversely, RSI < 30 signals that the stock may be oversold, and a rebound is likely (Fernando, 2024).

2.3 Stochastic Oscillator

The Stochastic Oscillator is a momentum indicator that compares adjusted closing price to the absolute highest and lowest prices over a specified lookback window (Hayes, 2024). Like RSI, it aims to identify overbought and oversold conditions. It consists of 2 lines:

- **%K Line**: Measures adjusted closing price relative to price range (lowest and highest price) over specified lookback window (typically 14-Days)
- **%D Line**: SMA applied to %K line (typically 3-Days) to smooth out noise

When %D crosses above 20, the Stochastic Oscillator indicates that a potential bullish reversal, providing a buy signal. Conversely, when %D crosses below 80, it indicates a potential bearish reversal, providing a sell signal.

Since we use the high and low of the lookback window, the Stochastic Oscillator tends to perform better in a ranging market. It generates more frequent signals than RSI, especially in strong trends. Therefore, while the Stochastic Oscillator is useful for giving us early potential buy and sell signals, it should be used in conjunction with slower moving indicators, such as RSI, for added confluence.

3 MANUAL STRATEGY

3.1 Strategy Details

In both the manual strategy and strategy learner, we used the same starting cash of $100,000. Allowable positions are 1000 shares long, 1000 shares short, or 0 shares. We can trade up to 2000 shares at a time, with unlimited leverage, assuming commission of $9.95 and trade impact of 0.005 (0.5%). JPM (JPMorgan Chase & Co.) is the only symbol used. The in-sample time period is from January 1, 2008 to December 31, 2009, and the out-of-sample period is from January 1, 2010 to December 31, 2011.

The manual strategy is essentially a rule-based strategy defining if-else rules for individual indicators values to determine a combined signal representing positions: LONG, CASH, SHORT.

The strategy leverages the combined signals of lagging and leading indicators to enter LONG and SHORT positions when we are in CASH. Once we have trading positions open, we take a more aggressive profit taking approach, requiring only 1 indicator to flash overbought/oversold signals to exit our position. In the case where indicators flash extreme overbought/oversold signals, we immediately flip our position. In all scenarios, if the strategy is already in the desired position (long/short/cash), we do not perform any trade and simply hold the current position. The pseudocode to implement the manual strategy is as follows:

```
Shift all indicators by one day to use previous day's indicator data for today's trades

For each day:
    if in CASH:
        if bbp < 0.2 and rsi < 35 and stoch_d < 20:
            enter LONG
        elif bbp > 0.8 and rsi > 65 and stoch_d > 80:
            enter SHORT
    elif in LONG:
        if bbp > 0.9 or rsi > 80 or stoch_d > 90:
            close LONG, flip SHORT
        elif bbp > 0.8 or rsi > 65 or stoch_d > 80:
            close LONG, enter CASH
    elif in SHORT:
        if bbp < 0.1 or rsi < 20 or stoch_d < 10:
            close SHORT, flip LONG
        elif bbp < 0.2 or rsi < 35 or stoch_d < 20:
            close SHORT, enter CASH
```

3.2 Performance Evaluation

We tuned our rules to get the best performance possible during the in-sample period without peeking at out-of-sample performance. The benchmark used for comparison is a buy-and-hold strategy. We simply buy 1000 shares at the start and hold that position for the entire trading period.

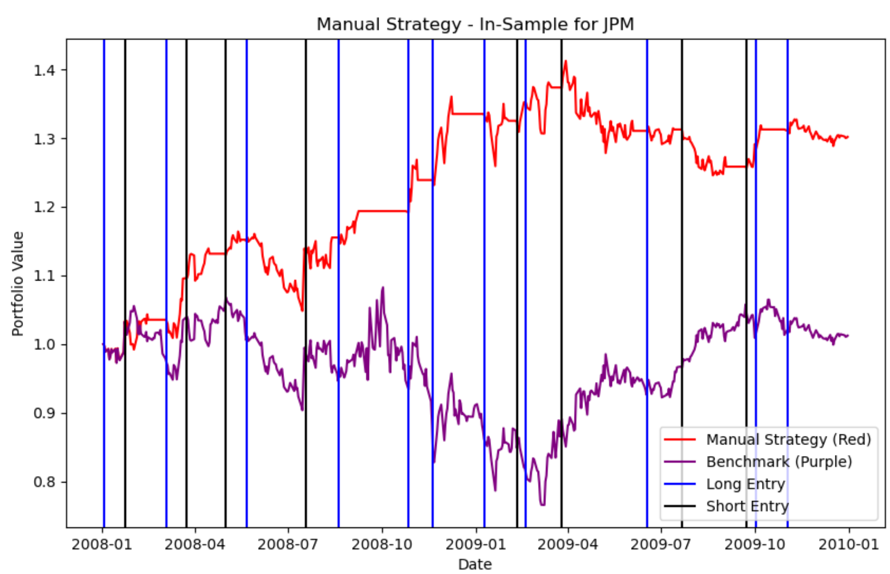

Figure 1 — In-Sample Normalized Portfolio Value Comparison between Manual Strategy and Benchmark

We plot the normalized daily portfolio valuations of the manual strategy (in red) and the benchmark (in purple) in Figure 1 above. Long entries by our manual strategy are indicated by the blue vertical lines, while short entries are in black.

In terms of cumulative returns, our manual strategy clearly outperforms the benchmark (purple) during the in-sample period, which as expected as we tuned our rules specifically for returns during this period. However, the true indication of a successful strategy would be its out-of-sample performance.

Next, we compare their out-of-sample performance in Figure 2 below. While it outperformed the benchmark throughout most of the period, especially towards the end after July 2011, there were periods of time where the manual strategy made bad decisions to long/short, eroding most of the gains that it had over the benchmark. However, it still did reasonably well with a 10% portfolio gain considering the benchmark resulted in an 8% portfolio loss.

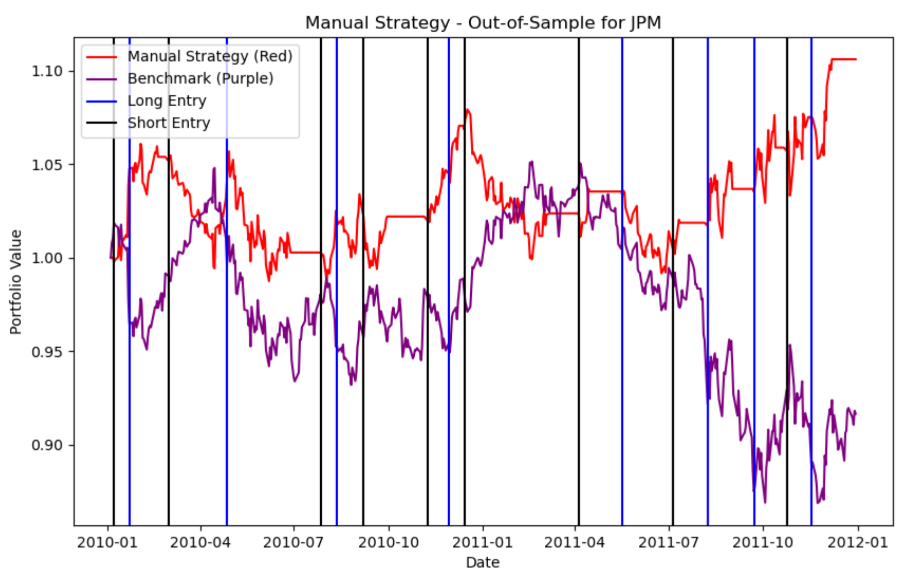

Figure 2 — Out-of-Sample Normalized Portfolio Value Comparison between Manual Strategy and Benchmark

Table 1 below compares the manual strategy and benchmark over the in-sample and out-of-sample period with several performance metrics.

Table 1 — Performance Metrics for Manual Strategy and Benchmark

Metric	Manual Strategy		Benchmark	
	In-sample	Out-of-Sample	In-sample	Out-of-Sample
Cumulative Return	0.301708	0.106182	0.012325	-0.083579
Standard Deviation of Daily Returns	0.009913	0.006556	0.017041	0.008500
Mean of Daily Returns	0.000572	0.000222	0.000169	-0.000137

The manual strategy outperforms the benchmark in every metric for both in-sample and out-of-sample time periods. However, the performance difference during the out-of-sample period was significantly less. For example, the manual strategy beat the benchmark by close to 30% during the in-sample period, but that shrunk to 18% during the out-of-sample period. This could be attributed to overfitting to historical data, where patterns, trends and anomalies were exploited during the in-sample period to tune the thresholds used for our indicators. It could also be due to changes in market conditions, resulting in changes to trends and volatility that our momentum indicators may struggle more with.

4 STRATEGY LEARNER

4.1 Framing the Trading Problem

The chosen strategy is a random forest classification learner implemented using BagLearner and RTLearner. BagLearner utilizes bootstrap aggregation to create an ensemble of Random Tree learners.

In the training phase, the values of the three indicators were used to construct the trainX array, while trainY was constructed using the 5-day change in price. We represent the trading actions (BUY, DO NOTHING, SELL) as the categories (+1, 0, -1) for our learner. To ensure that signals were only given for sufficiently profitable price changes, we settled on a threshold of 0.032. We also took into account market impact in our calculations, so the 5-day change in price had to be >(0.032 + impact) for a BUY signal and <(-0.032 – impact) for a SELL signal.

Otherwise, a DO NOTHING signal was given. The trainX and trainY array were used to train our BagLearner consisting of 20 random trees with leaf size of 5.

In the testing phase, we call our BagLearner with unseen testX data. The BagLearner gives us its prediction testY by taking the **mode** output from the 20 trees. As the RTLearner uses regression algorithms, we do not get discrete 1, 0 or -1 values in testY. To ensure that we trade only on strong signals, we BUY when testY is > 0.5, SELL when testY is < -0.5 and DO NOTHING otherwise. The strategy learner then decides whether to trade 1000 or 2000 shares based on the current holdings, staying within the allowable position limit of 1000 shares.

4.2 Tuning Model Hyperparameters

The random forest learner is controlled using two hyperparameters: leaf size and number of bags. Our optimal values of 5 and 20 respectively were determined by performing a grid search for leaf size = [5, 6, 7, 8, 9, 10] and number of bags = [20, 25, 30, 35, 40, 45, 50], and the combination with the best in-sample performance in terms of cumulative returns was selected.

4.3 Data Adjustment

Unlike the case if a reinforcement learner (QLearner) was used, no standardization or discretization of the indicator values were performed for our random forest learner. The only data manipulation needed was in constructing the trainY array, where the 5-day change in price was used to determine the Y target values.

5 EXPERIMENTS

5.1 Experiment 1

In experiment 1, we pit the manual strategy against the strategy learner over the in-sample and out-of-sample periods. We abide by the same constraints described in Section 3.1. For our strategy learner, we have kept the same hyperparameters (leaf size = 5, number of bags = 20).

Our hypothesis is that the strategy learner will outperform the manual strategy and benchmark over both the in-sample and out-of-sample periods.

In Figure 3 and 4 below, we plot the normalized portfolio values of the manual strategy and strategy learner over the in-sample and out-of-sample periods respectively. We also plot the values of the benchmark using a buy-and-hold strategy for comparison.

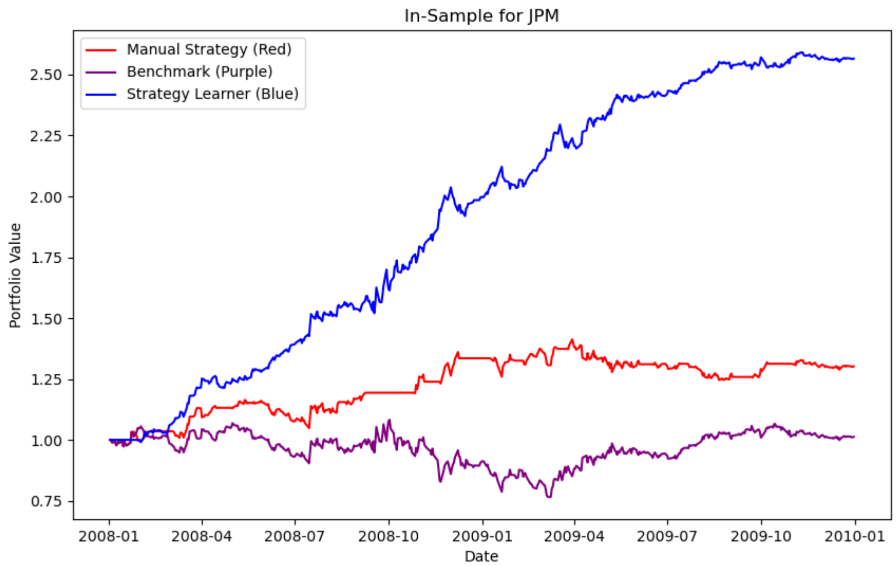

Figure 3 — In-Sample Normalized Portfolio Value Comparison between Manual Strategy, Strategy Learner and Benchmark

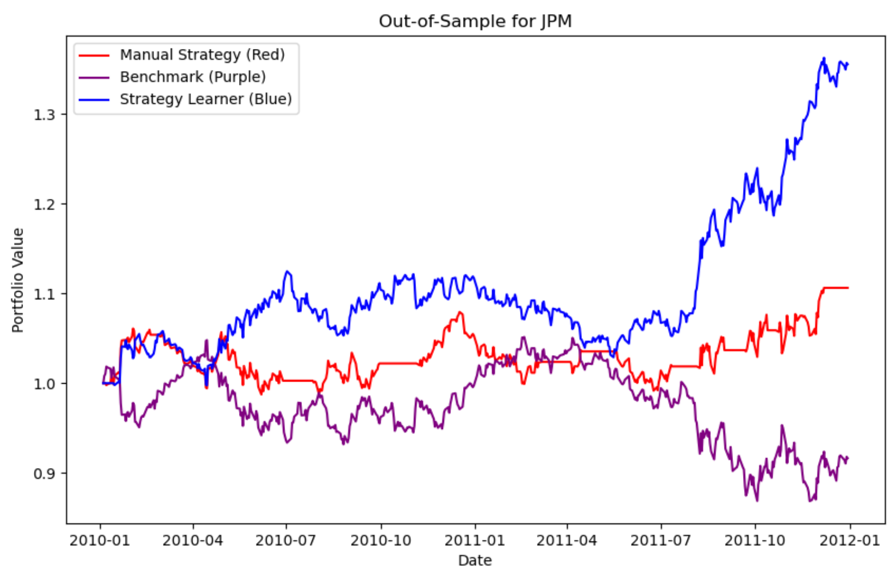

Figure 4 — Out-of-Sample Normalized Portfolio Value Comparison between Manual Strategy, Strategy Learner and Benchmark

The results for both time periods support our hypothesis, providing evidence that machine learning algorithms can potentially give us much better returns. Figure 3 shows that our strategy learner outperforms both the manual strategy and benchmark significantly at almost every point during the in-sample period. The strategy learner returned 2.5 times the initial portfolio value, compared to 1.3 times in the manual strategy and roughly breakeven for the benchmark.

This is unsurprising as machine learning algorithms tend to perform very well on data they have seen and trained on. We would expect this relative result almost every time with in-sample data for the random forest learner due to its flexibility to learn complex, non-linear relationships in the data that we cannot capture manually. However, there could be occasional exceptions due to its inherent randomness introducing performance variability.

Figure 4, however, shows a less rosy result. All 3 strategies remained close in performance for about two-thirds of the out-of-sample period, before they sharply diverged around after July 2011. This could be due the low volatility initially, causing our indicators which rely on momentum to generate false signals. Another reason could be due to overfitting, where our random forest captures the small nuances and noise in the in-sample data. Fortunately, both the strategy learner and manual learner pulled through in the end, giving over 40% and 20% returns over the benchmark.

5.2 Experiment 2

In experiment 2, we investigate the effects of different values of market impact on in-sample trading behavior of our random forest classification learner. We use three significantly different impact values: 0.05, 0.005 and 0.0005.

We have kept the same constraints and hyperparameters used in our earlier experiment. However, commission is $0.00 here.

Our hypothesis is that as the impact value increases, the learner performs fewer trades due to diminished profits per trade. The expectation is that this would result in lower cumulative returns. We plot the normalized portfolio values of the three different impact values in Figure 5 below.

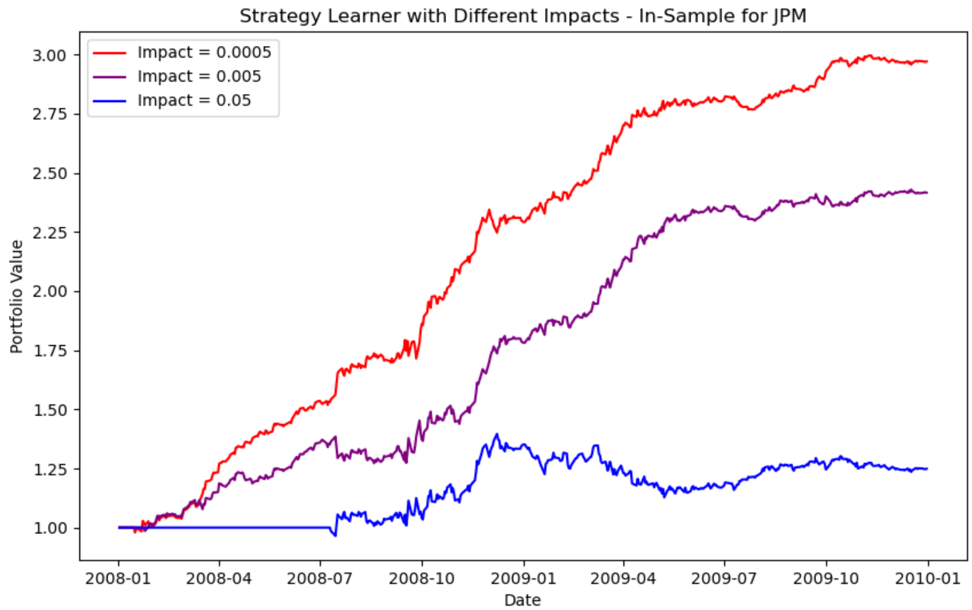

Figure 5 — In-Sample Normalized Portfolio Value for Different Market Impact Values

Table 2 below presents performance metrics for the different impact values over the in-sample period.

Table 2 — Performance Metrics for Different Impact Values

Metric	**i = 0.0005**	**i = 0.005**	**i = 0.05**
Number of Trades	48	42	5
Cumulative Return	1.970698	1.415721	0.248740

The results support our hypothesis. Number of trades decreased from 48 to 42 to 5 as impact increases from 0.0005 to 0.05 to 0.05. Cumulative returns also decreased significantly from 1.97 to 1.42 to 0.25.

This can be attributed to the fact that increasing market impact imposes greater additional cost to our trade execution, reducing the profits of each trade. As a result, more trades fail to meet our expected returns and hence do not get executed by our learner. Ultimately, this results in reduced cumulative returns.

Our findings suggest that we should consider employing our learners in more liquid markets and stocks with larger market capitalizations to reduce the effects of market impact, increasing the reliability and returns of our trade execution.

6 REFERENCES

1. Charles Schwab. (n.d.). *Bollinger Bands: What they are and how to use them*. Charles Schwab. https://www.schwab.com/learn/story/bollinger-bands-what-they-are-and-how-to-use-them. Accessed October 19, 2024.
2. Thompson, C. (2024, April 16). *Bollinger Bands*. Investopedia. https://www.investopedia.com/terms/b/bollingerbands.asp
3. Barchart. (n.d.). *Bollinger Bands Percent*. Barchart. https://www.barchart.com/education/technical-indicators/bollinger_bands_percent
4. Fernando, J. (2024, September 23). *Relative strength index (RSI)*. Investopedia. https://www.investopedia.com/terms/r/rsi.asp
5. Hayes, A. (2024, September 6). *Stochastic oscillator*. Investopedia. https://www.investopedia.com/terms/s/stochasticoscillator.asp