

Strategy Evaluation

Aarti Laddha
aladdha7@gatech.edu

1. INTRODUCTION

This project required us to generate a trades data-frame containing Buy, Sell or Hold signals by leveraging our historical stock prices. We achieve this in 2 different ways by creating below strategies –

1. **Manual Strategy** – I implemented a set of rules on the values of my indicators to generate trading signals.
2. **Strategy Learner** – I built an ensemble learner using RTLearner and BagLearner that could learn the trading policy using the same set of indicators leveraged in manual strategy.

2. INDICATOR OVERVIEW

Below are the 3 indicators that I have used –

2.1 Simple Moving Average (SMA)

SMA is simply a rolling mean of our stock prices over a specified rolling window (days here), formula for which is as below -

$$SMA = \sum_{i=1}^n \frac{(a1 + a2 + \dots + an)}{n}$$

Where, n = number of periods & a1, a2 ... indicate stock prices.

I have used **Price/SMA ratio** in the strategy implementation.

2.2 Bollinger Bands Percentage

Bollinger bands uses simple moving average and created 2 additional bands, formula for which is as below –

$$upper_{band} = sma + 2 * std$$

$$lower_{band} = sma - 2 * std$$

Where, std = rolling standard deviation

Bollinger Bands Percentage (BBP%) combines the upper and lower band, formula for which is –

$$BBP\% = \frac{(sma - lower_{band})}{(upper_{band} - lower_{band})}$$

2.3 Momentum

Momentum simply measures how much has the price of a stock changed over several days. Formula for momentum is as below –

$$Momentum = \frac{price[t]}{price[t - n]} - 1$$

Where, n is the lookback period. Momentum usually ranges between -0.5 to 0.5.

For all these indicators, the parameter to optimize is the moving window. In my experiment, I have used a moving window of 21 days. In my view, this moving window is neither too short to generate false signals nor too long to not capture the real trends in price movements.

3. MANUAL STRATEGY

For the implementation of manual strategy, I picked the indicators described above.

How did I interpret the indicators & their threshold?

Price/SMA - Utilized this indicator to look for places where the current price is crossing through the simple moving average as these are the special events where we need to focus. I have considered a moving window of 21 which is in between. My basic idea behind generating trading signals via Price/SMA ratio was that the rolling price over a certain period might represent the true price of the stock. So, if we see a large excursion from that price, we should expect that the price will come back to the average price.

- If `price/sma` ratio < 0.6 i.e., current price is 60% of the rolling price, which is a significant deviation, we would expect the price to increase and go up to the `sma`. So, go long when such events occur.
- If `price/sma` ratio > 1.0 i.e., the current price is higher than the average price, we would expect the price to decrease and go down to the `sma`. Go short when such events occur.

Momentum - If the momentum is positive, it's an indication of buy opportunity as the stock is rising fast. Conversely, if the momentum is negative, it indicates a sell opportunity as the price is going down. Momentum usually ranges between -0.5 to 0.5.

- If momentum < -0.1 , this might be an oversold condition and the price is going to go up. Go long when such events occur.
- If momentum > 0.1 , this might be overbought condition & the price would eventually come down. Go short when such events occur.
- I did not consider the range between -0.1 to 0.1 as this is a small deviation from rolling price and might generate a false trading signal.

Per my research, I found that it's often advisable to combine price/sma and Momentum. If the price has strong momentum and it's crossing through that simple moving average, that could be a trading signal.

BBP% - Here is what the BBP% indicates (marketvolume.com) -

- BBP% is below 0 when price is below the lower band.
- BBP% is above 1 when price is above the upper band.
- BBP% readings above 0.80 indicate that the price is near upper band and sometimes can be interpreted as overbought. Overbought is a condition that occurs when prices are considered too high & susceptible to a decline. Go short when such events occur.
- BBP% readings below 0.20 indicate that the price is near lower band and can be interpreted as oversold. Oversold is a condition that occurs when prices are considered too low and ripe for a rally. Go long when such events occur.

To design the strategy, I used a `signal` variable to create a logical expression that would yield -1, 0, or 1, corresponding to a “short,” “out” or “long” position & followed the steps below -

1. Initialize the signal variable to 0 at the beginning of time since we don't have any holdings.
2. I combined the Buy/Sell signals from all the 3 indicators above & based on the type of trade, signal variable is reset each time.
3. Since at any point, we can only have following trade positions i.e., -2000, -1000, 0, 1000, 2000. I factored this in my code such that when signal is 0, I trade only 1000 stocks long/short. However, when the signal is -1/1 and I already have holdings, I trade 2000 stocks log/short depending on the indicator signals.

This rule-based strategy should certainly be better than our benchmark strategy because here we are generating trading signals based on indicator trends and price movements.

In Sample Results –

JPM, 1/1/08 – 12/31/09, Starting cash = \$1000000, Commission = 9.95 & Impact = 0.005

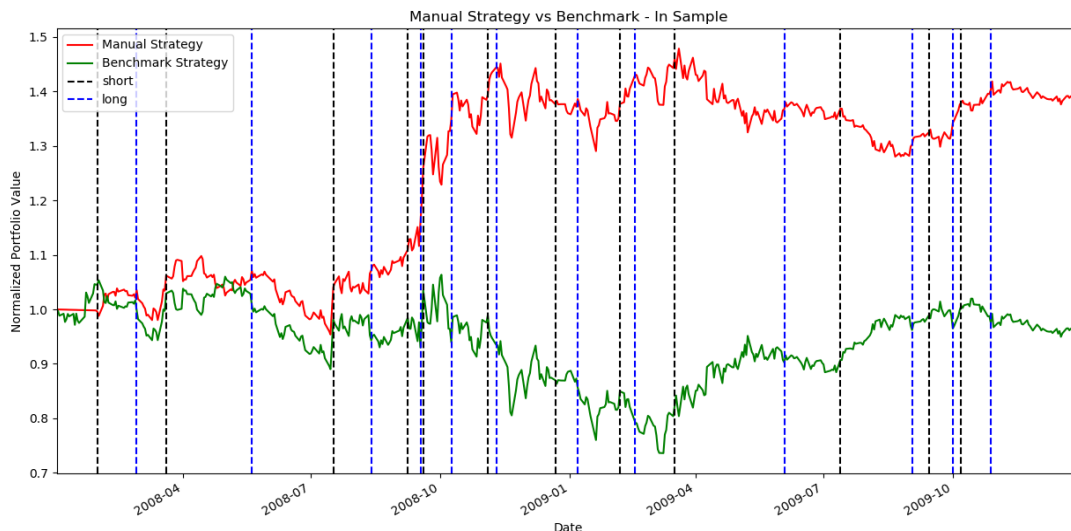


Figure 1. In sample results for Manual vs Benchmark (JPM)

Out of Sample Results -

Parameters – JPM, 1/1/10 – 12/31/11, Starting cash = \$1000000, Commission = 9.95
& Impact = 0.005

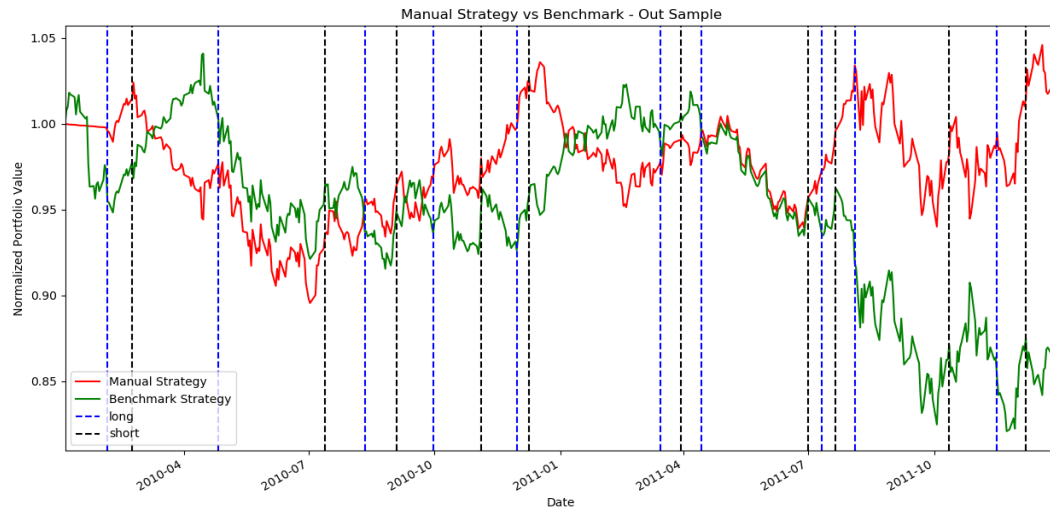


Figure 2. Out of sample results for Manual vs Benchmark (JPM)

Metrics – In Sample vs Out of Sample for JPM

Duration	Type	Daily Return	Cumulative Return	Volatility	Sharpe Ratio
In Sample	Manual	0.0007292	0.3887967	0.0124787	0.9277002
	Benchmark	0.0000710	(0.0399137)	0.0174682	0.0644965
Out of Sample	Manual	0.0000710	0.0202195	0.0079066	0.1425225
	Benchmark	(0.0002506)	(0.1353651)	0.0087819	(0.4530173)

Table 1. Manual vs Benchmark Strategy

We can clearly see from the charts and the table above that the rule based manual strategy beats the benchmark strategy. Except for a few instances in out of sample period, rule based manual strategy has better daily return, cumulative return with

lower volatility in both, in sample & out of sample period when compared to benchmark strategy.

4. STRATEGY LEARNER

Steps to frame the trading problem as a learning problem for your learner -

With the strategy learner, I have converted the trading problem into a learning problem. **This is achieved by implementing an ensemble Bag Learner with a Bag size of 25 and leaf size of 5.** The Bag Learner in turn uses Random Tree Learner (25 of them) to perform the learning task.

Training -

- The `add_evidence` method gets the stock prices for in sample period.
- These values are then used to compute the indicator values i.e., Price/SMA, Bollinger Bands % and Momentum. This in turn is the `Xtrain` data frame.
- `Ytrain` is the future N day return. For my learner, I have used 10-day return.
- Impact will affect the price, so our trade will become less profitable. As such, we need impact to penalize our trading decisions so that we can account for the lost profitability before deciding. The default impact is 0. As such, I have added a market variance of 1.5% to compare with N day return.
- This is a classification learner & will generate labels as the output. For my trading problem, I have converted the recommended actions of Buy, Sell, Hold as 1, -1, 0 respectively.
- The learner uses this data to learn a strategy and will be making predictions for future price changes based on this learning.

Testing –

- `testPolicy` method defined in the code is used to test the learner against the sample indicator data.
- The query method returns the `Ytest` data frame.
- The signals generated in `Ytest` data frame is then used to generate trades_df based on signal values of -1, 0, 1.

- Since at any point, we can only have following trade positions i.e., -2000, -1000, 0, 1000, 2000. I factored this in my code such that when signal is 0, I trade only 1000 stocks long/short. However, when the signal is -1 or 1 and I already have holdings, I trade 2000 stocks long or short depending on the signal value.

Discretization/ Adjusting Data -

I did not have to adjust/discretize my prices data because I am using an Ensemble Bag Learner. In case I had opted for Qlearner, discretization of data points would have been necessary as the states have to be an integer only. The only adjustment that I had to make was to convert my regression learner to classification learner by using mode instead of mean in the query method of Bag Learner.

5. EXPERIMENT 1

In this experiment, we are comparing Manual Strategy, Benchmark Portfolio and Strategy Learner. I have used 'Portfolio_Statistics' function to determine which strategy is the best by comparing the portfolio metrics such as Avg Daily Returns, Cumulative Returns, Volatility and Sharpe Ratio. I have also set seed value to get the same output for each run of Experiment 1 in the main function of 'Experiment 1.py' file.

Parameters - JPM, 1/1/08 – 12/31/09, \$1000000 starting cash, Commission = 9.95 & Impact = 0.005. Moving window = 21

For Strategy Learner, Leaf size of 5 & Bag Size of 25 is used.

Hypothesis – Strategy Learner will beat Manual Strategy which in turn will beat benchmark.

Duration	Type	Daily Return	Cumulative Return	Volatility	Sharpe Ratio
In Sample	Strategy Learner	0.00163247	1.21101088	0.01070809	2.42010441
	Manual Strategy	0.00072925	0.38879669	0.01247869	0.92770016
	Benchmark	0.00007097	(0.03991375)	0.01746825	0.06449646

Table 2. In Sample performance of Strategy learner vs Manual vs Benchmark for JPM

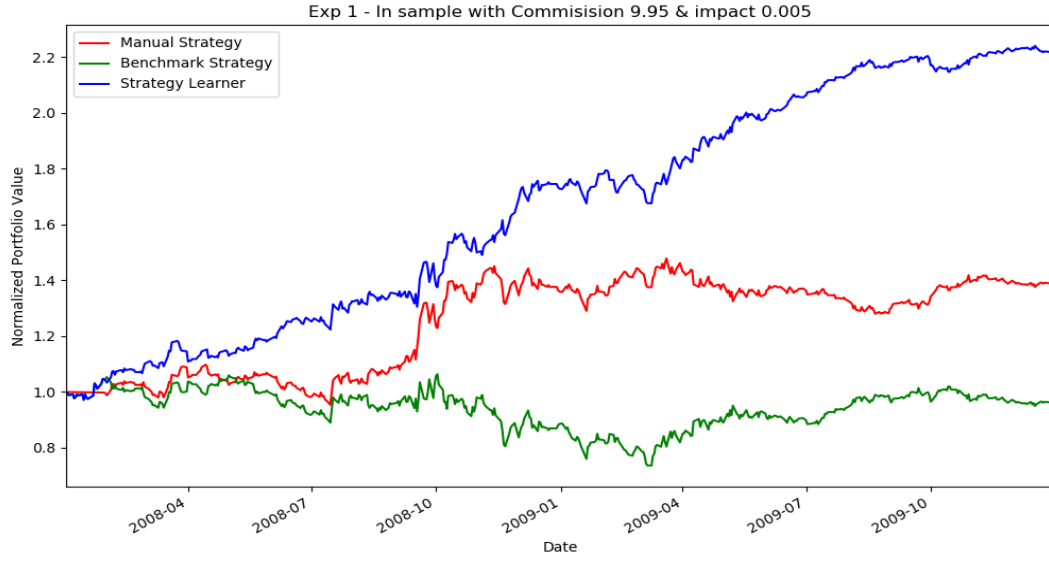


Figure 3. Experiment 1 - Strategy Learner vs Manual vs Benchmark

From the table and chart above, we can see that the experiment outcome is in line with our hypothesis.

1. Daily returns, cumulative returns & Sharpe Ratio – Higher returns and Sharpe ratio (risk adjusted return) is better -

Strategy learner > Manual Strategy > Benchmark

2. Volatility – Lower standard deviation of returns is better -

Strategy Learner < Manual Strategy < Benchmark

Consistency of results -

For the in-sample period, I think that strategy learner would **always** beat benchmark strategy & **almost always** beat the manual strategy. There might be cases where the Strategy Learner performs worse than the Manual Strategy. This is because our ensemble learner is using RTLearner which involves randomness in choosing the best factor for the split. As such, because of the inherent randomness, it might generate a sub optimal tree. But I believe such cases would

be relatively lower & Strategy learner would beat the Manual Strategy most of the times.

6. EXPERIMENT 2

In this experiment, we will determine the effect of impact on the in-sample trading behavior. I have also set seed value to get the same output for each run of Experiment 1 in the main function of `Experiment 2.py` file.

Hypothesis – Impact is the amount the price moves against the trader compared to the historical data at each transaction. Impact of 0.01 corresponds to 1% impact. Hence, it will be favorable to the trader when this impact value is low. **As such, we should expect better performance of Strategy Learner for lower values of impact.**

Parameters - JPM, 1/1/08 – 12/31/09, \$1000000 starting cash, Commission = 0.0 & Impact = 0, 0.002, 0.05, 0.1

For Strategy Learner, Leaf size of 5 & Bag Size of 25, moving window of 21 days is used.

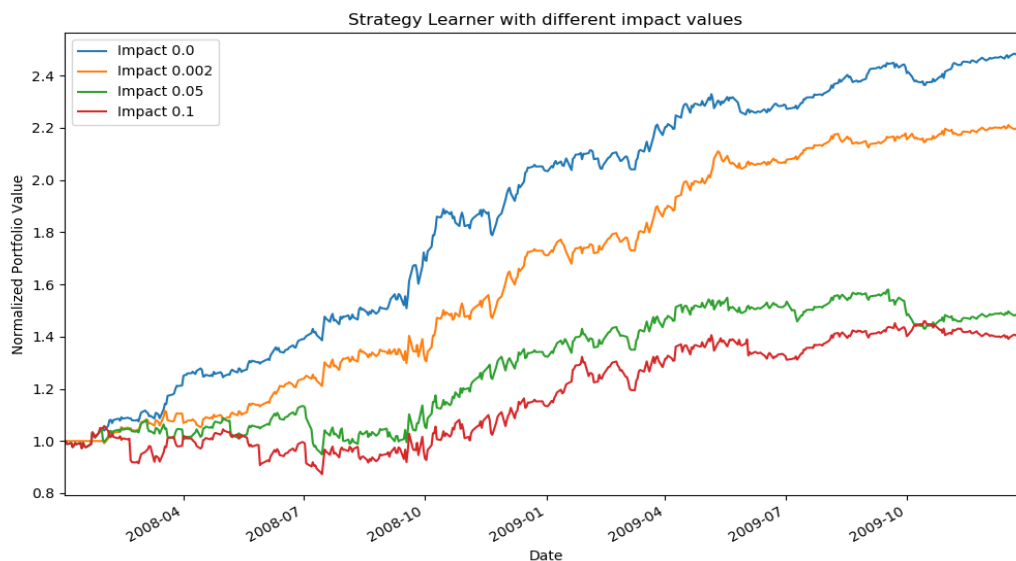


Figure 4. Strategy Learner with varying impact

Duration	Impact	Daily Return	Cumulative Return	Volatility	Sharpe Ratio
In Sample	0.0	0.00184530	1.47530000	0.00956556	3.06237330
	0.002	0.00161426	1.19229960	0.01057461	2.42330756
	0.05	0.00089040	0.48368366	0.01466116	0.96408664
	0.1	0.00081951	0.40462596	0.01700647	0.76496415

Table 3. In Sample performance of Strategy Learner for different impact values

From the table & chart above, we can see that the experiment results are in line with our hypothesis. **Strategy Learner with lower impact performed better than strategy learner with higher impact.** Another interesting thing to look at was how the number of trades reduced as the impact becomes greater. This is expected because higher impact will result in lower expected return & when that is compared against our threshold, the trade might not be profitable & hence not executed.

```

Number of trades with impact 0.0 is 81
Number of trades with impact 0.002 is 66
Number of trades with impact 0.05 is 30
Number of trades with impact 0.1 is 13

```

Figure 5. Comparing # of trades for Strategy Learner with varying impacts

7. REFERENCES

1. <https://www.marketvolume.com/technicalanalysis/bollingerpercent>
2. <https://en.wikipedia.org/wiki/MACD>
3. [https://www.marketvolume.com/technicalanalysis/standard_deviation.a
sp](https://www.marketvolume.com/technicalanalysis/standard_deviation.asp)