Project 8: Strategy Evaluation
CS 7646

Aarushi Gupta
agupta857@gatech.edu

Abstract—This project uses three technical indicators - Bollinger Bands Value (BBValue), Moving Average Convergence Divergence (MACD) and Relative Strength Index (RSI) - to generate the trades. Manually developed strategy and Machine Learning based strategy is used to evaluate the performance. Methods adopted are explained with the results obtained in this report. All results are compared to a benchmark. Strategy evaluation, comparisons and experiments were performed to understand the behaviour on in-sample and out-sample data.

1. INDICATORS OVERVIEW

Three indicators are used for this project - BBVALUE to understand the market trend; MACD to monitor the momentum; and RSI to see when the stock is overbought or oversold. How each indicator generates the BUY or SELL signal is explained in the following sub sections.

1.1. Moving Average Convergence/Divergence Oscillator (MACD)

This indicator turns two period moving average (12 day ema and 26 day ema) into a simple and most effective signal available. It represents the difference between the short and longer trend following indicators. This conjugates both trend following and momentum.

MACD = EMA[12 day period] - EMA[26 day period]
Signal = EMA[MACD, 9 day period]

Now to generate the trading indication we used signal line, which is a 9 day EMA of MACD. Now, traders might BUY the stock if the MACD crosses above its signal line, or SHORT when MACD crosses below its signal line. Thus, we used the MACD and its Signal line to find the entry point and confirmed the trade by looking at RSI and BBValue.

1.2. Bollinger Bands Value (BBValue)

Bollinger Bands Value represents a value which is a ratio of difference between price and its moving mean to twice the moving standard deviation.. It is plotted with price and SMA. If the price grazes or crosses the upper band that indicates the SELL signal and it is said to be bound to resonate back to the moving average. Similarly, if the price goes below or grazes along the lower band it is said to be underbought and hence a BUY signal as it is expected to move back within the bands.

BB = [SMA + STD, SMA - STD]
BBValue = (Price[t] - SMA[t]) / 2* STD[t]

BB when combined with MACD brings volatility along with trend-following and momentum. Breakout from the Bollinger bands present high volatility and high MACD represents an increasing divergence. This can indicate if we need to long the stock or short it.

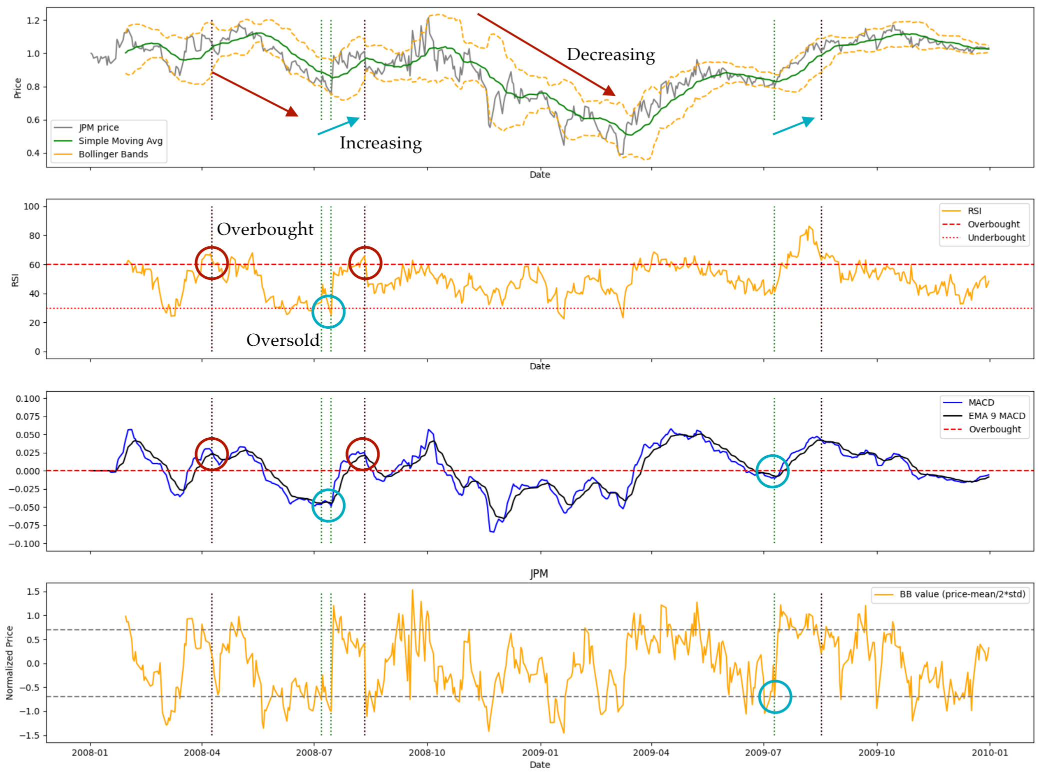


Figure 1- Indicators plotted for JPM InSample data, marked with indicators BUY or SELL signal (combined as explained in section 2.1).

1.3. Relative Strength Index (RSI)

RSI is an indicator which oscillate between 0 to 100 depicting the percentage to measure the recent price changes. It is similar to momentum indicator and signals an overbought or underbought situation. It is calculated for 14 day period here. The RSI will rise as the number and size of positive closes increase, and it will fall as the number and size of losses increase. A reading below 30 will be a BUY signal and above 60 it will show a SELL signal.

The RSI and MACD are both trend-following momentum indicators that show the relationship between two moving averages of a security's price. ... The MACD measures the relationship between two EMAs, while the RSI measures price change in relation to recent price highs and lows, showing when the security is overbought or oversold.

The combined charts for all the indicators and price for JPM in sample data is shown in Figure 1. The indicators are marked for sell and buy signals.

MACD gives crisp signals as we see only crossovers, thus RSI and BBValue boundaries were kept lenient as they are used for confirmation only.

The chart marks the buy or sell signals according to the combination of indicators as explained in section 2.1.

Indicators	BUY	SELL
MACD with Signal line	np.diff(MACD - Signal, axis=0) > 0	np.diff(MACD - Signal, axis=0) < 0
RSI	Below 30	Above 60
BBVALUE	Below 0.7	Above 0.7

Table 1- Indicators Signals with tuned thresholds.

2. MANUAL STRATEGY

2.1. Entry and Exit Positions using Indicators

The BUY and SELL signals for each indicators is already explained in the section 1. Now to perform the trade we have used MACD as the primary indicator to show a signal at each crossover with signal line, which is then supported by either RSI or BBValue.

LONG : **MACD == buy && (RSI == buy | BBV == buy)**

SHORT : **MACD == sell && (RSI == sell | BBV == sell)**

Whenever MACD crosses above the signal line it is sufficient to say that the price will increase but to avoid any long term losses we look at RSI to confirm. RSI tells the how price has risen or fallen with respect to recent changes, that is, if the stock is overbought or oversold looking at the trend following momentum it measures. However, RSI sometimes missed the opportunities which were not clear enough. Therefore BBValue was used to support the MACD at those places. BBValue understands when the stock is more volatile and because of this it gives us a clear indication of breakout from the Bollinger bands which tells us that price is bound to resonate back to the moving average and hence a good opportunity to go bullish or bearish.

Therefore, we used only those MACD indications which were supported by RSI or BBValue. Combined together with thresholds described in Table 1, it generated good enough trades as compared to benchmark.

2.2. Development of strategy

We use the combined buy or sell indications to generate the trades under following cases.

Case 1 - Net holdings == 0 : If signal is bullish we LONG 1000 stocks, else if bearish we SHORT 1000 stocks.

Case 2 - Net holdings == 1000 : We SHORT 2000 stocks if sell signal is True, else we HOLD the position.

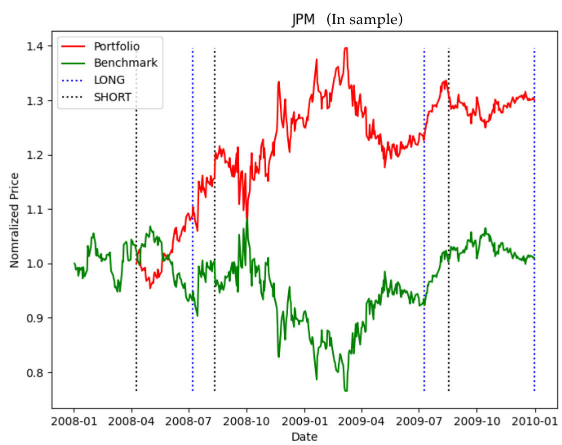
Case 3 - Net holdings == -1000 : We LONG 2000 stocks if bullish, else we HOLD the position.

For the last day it is programmed to have zero net holdings, thus, if we have net holdings = -1000 then we buy 1000, if it is equal to +1000 then we sell. And we have our cumulative return.

There were not any specific assumptions made apart from that the leverage will be unlimited with 9.95 commission and 0.005 impact. However, the trades were generated only for the eligible dates (when SPY traded too).

2.3. Manual Strategy Results

For this report all results are generated for 'JPM' stock with starting value of $100,000. The in-sample period is January 1, 2008 to December 31, 2009 and out-of-sample/testing period is January 1, 2010 to December 31 2011. Each of them is compared to Benchmark Trade - investing in 1000 shares of the symbol in use on the first trading day, and holding that position.



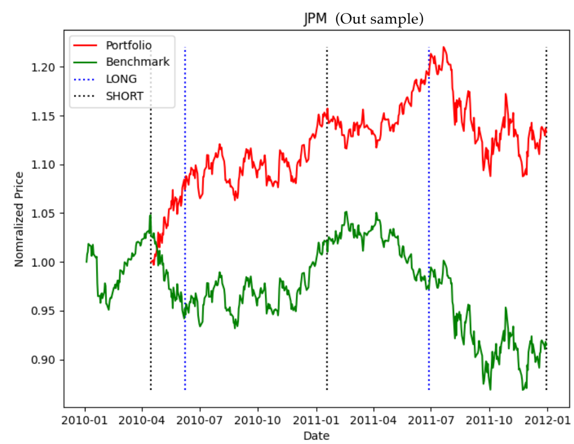


Figure 2- In sample and Out sample portfolio values for trades generated by Manual Strategy as compared to Benchmark trade.

	In Sample	**Benchmark**	**Out Sample**	**Benchmark**
Cumulative Return	0.3003	0.1533	0.133	-0.263
Average Daily Return	0.00069	0.00016	0.000317	-0.000141
Standard Deviation	0.013	0.017	0.007	0.008
Sharpe Ratio	0.806	0.153	0.684	-0.2636
Final Portfolio Value	**130034**	**101023**	**113373**	**91469**

Table 2- Comparison between InSample, OutSample and Benchmark Manual Strategy.

2.4. Interpretation

In sample

Table 2, shows that Manual strategy provided a 30% increase in the start value as compared to the bench-mark which only provided 1% increase. And the results are better not just for final day but for the duration it provides lesser standard deviation and high average daily return too.

The second sell trade in Figure 2 (left) shows that while the price and benchmark portfolio was decreasing drastically, the manual strategy owing to this sell trade was able to gain by shorting the stocks. However, the first sell trade was wrongly decided as it gave us the negative outcomes as compared to benchmark. The buy trade were good as they were placed at right positions where the stock price was going to increase.

Overall, the portfolio value for manual strategy comes out to be much better than benchmark.

Out sample

This data was completely new to the strategy, where all indicators were tuned to suit the In sample data, however, the portfolio value for even out sample data has also given outstanding results when compared to benchmark. It has shown a 13.3 % increase in start value whereas benchmark gave a loss of -26.3%. Hence, when the stock was facing loss the strategy selected correct shorting positions to make a profit. Even the average daily return and Sharpe ratio were much higher and positive to show that buy trades were also good enough and not blindly trusted the shorting strategy here.

Therefore, even if tuned to in sample the strategy worked very well with out sample period too.

In sample V/S Out sample

In sample was tuned to give good trades but for out sample period the strategy failed to select good buy trades, as we can see the buy trades decrease the portfolio value, therefore, the out sample performed better than benchmark only relying on the sell trades. But the in sample period, buy and sell both trades helped in improving portfolio performance.

Overall the manual strategy developed was not able to bring out the best trades in any of the sample period. Still the performance was much better than the benchmark and the strategy was good enough to work with both training and testing sample.

3. STRATEGY LEARNER

Setup of trade problem as learning problem

1. Setting up the learning environment : It is based on a BagLearner which internally implements 25 bags of Random Tree Learner with leaf size set to 5 (avoids overfitting).
2. Training of the data : add_evidence()

- The training data is generated from the stock price and in sample period shared as input.
- The required indicators are generated for each day using indicators.py.
- EMA is calculated over MACD to generate Signal Line. And these 4 values are then concatenated to form a 4 feature set of data representing each day.
- Now the classification of this data is decided by considering N day lookahead in the stock price. If the price increases by 2% in the next N days then the this trade is classified as 1 (LONG), else if the price decreases by 2% then this trade is classified as -1 (SHORT), for rest the trades is classified as 0 (no action taken). Thus, the trainX data obtained in previous step is classified to trainY as one of the three classes (1, -1, 0).

Hence, here we create the data comprising of indicator values and their labels using the classification criteria adopted.

3. Testing of the data : testPolicy()
   - The testing sample period with testing stock symbol is passed as input, which is then again converted into testing data X by calculating the 3 indicator values and Signal Value for MACD.
   - The data generated for the testing sample period passed now serves as the query data for our learner for which classification labels are returned by query() method.
   - The predicted Y returned are the classes - 1 (LONG), -1 (SHORT), 0 (DO NOTHING) - which decided on the trade that we will be executing.
   - Now trades are generated using the labels provided for each legal day.
If the label is 1 we generate a +1000 trade, if net holding == 0 else +2000.
If it -1 we generate a -1000 trade, if net holding == 0 else -2000.
On the last day we sell/buy to make net holdings == 0.

Hyperparameters determination

1. BagLearner Hyperparameters - Iterative executions of the strategy helped to decide the most optimum combination of leaf_size and bags.
   - Number of Bags - we decided to keep 25 bags/instances of RTLearner. RTLearner is fast enough thus computationally we can keep a high number of instances. High number compensates the randomness of RTLearner and explores multiple trade combinations.
   - Leaf_Size - anything greater than 3 should have been optimum to avoid overfitting, we kept 5 and increasing this value resulted in performance degrade.

2. Training/Testing Data Hyperparameters - Iterative executions of the strategy helped in deciding the good combination of N, YBUY and YSELL.
   - N - this the window we look ahead into to decide whether we buy or sell the stock today. The stock is highly volatile and the market trend changes very often that is why a small number was preferred. We initialised with 10 but decreasing it to 5 gave us better results. However, with the combination set for BagLearner N = 3 gave us the most optimum results.
   - YBUY/ YSELL - the threshold was set to +/-0.02 which gives a good 2% percent profit estimation when we decide to long or short the stock. Any value greater than this will result into increased number of trades which is not preferred if the impact is high. (more details shared in experiment 2) Thus to avoid many trades we kept the number low at 2% and between YBUY and YSELL no trade happened (labelled as 0).

Adjustment of Data

Data was not to be adjusted in out case because of the Learner we chose (BagLearner). If it may have been Q Learner that we will have to modify the data to allow it to work with the training/testing. The data was suitable to work with Random Tree learner which based on the indicator values has to just classify them into -1, 1, 0. Bag Learner was modified to use the mode of the values and nothing else was modified to make the data function well with the strategy learner.

4. EXPERIMENT 1

This experiment is done to compare the manual strategy to ML based learner strategy, given a benchmark trade to buy 1000 stocks on first day and hold that position.

Setup

The experiment is performed for in sample period for 'JPM' symbol with a start value of $100,000; 9.95 commission and 0.005 impact.

Steps

1. Generate Portfolio Values for all three strategies -
   Trades are obtained using the ManualStrategy.testPolicy() method for the given input data and then portfolio value is calculated using compute_portvals() method defined in marketsimcode.py. Similarly, benchmark portfolio values are created using get_benchmark() method in ManualStrategy.py. For StrategyLearner, the learner was trained using the learner.add_evidence() method and then trades were obtained using the learner.testPolicy() method. The portfolio values were calculated for the predicted in sample trades.

2. Normalize the portfolio Values - portvals = (portvals[-1] / portvals[0]) - 1.0 This ensured the start of each portfolio at 1.0
3. Calculate the Cumulative Return, Average daily return, standard deviation, Sharpe ratio along with chart generation. The results are shared below.

Assumptions

We assumed that Strategy Learner will perform better than Manual Strategy and Manual Strategy will perform better than Benchmark. We placed random.seed() method to ensure the reproduction of the results. The first trade in strategy needs not be aligned, thus size of portfolio values list is not same for all three.

Tabular Results

	Benchmark	**Manual Strategy**	**Strategy Learner**
Cumulative Return	0.01 (1%)	0.30 (30%)	2.23 (223%)
Average Daily Return	0.00016	0.00069	0.0024
Standard Deviation	0.017	0.014	0.0084
Sharpe Ratio	0.15	0.806	4.544
Final Portfolio Value	**101023**	**130034**	**323993**

Table 3- Experiment 1 Outcomes Statistics.

Outcome Comparison Graph

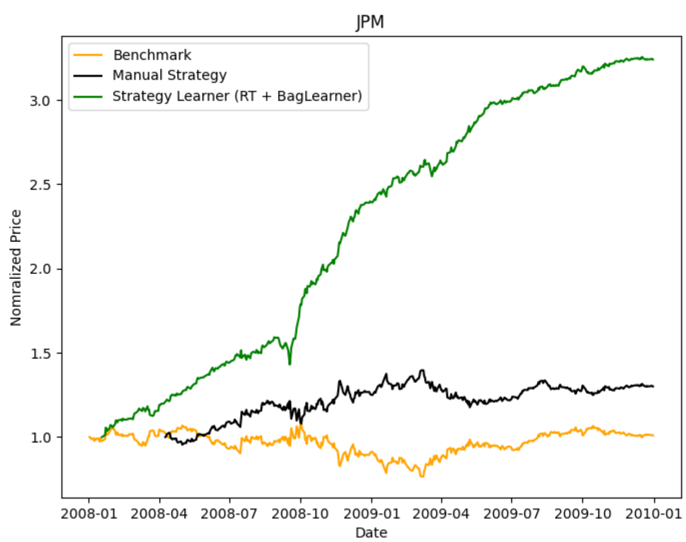


Figure 3- Comparison graphs for strategies on in sample period.

Interpretation

Strategy Learner has performed way better than the other two strategies. The reason is the automation of selection of trades. The number of trades have increased much more than the manual strategy which has resulted into more shorting and longing benefitting at each dip and rise in market.

However, the strategy learner and manual strategy will always perform better than benchmark, but strategy learner possess randomness and therefore can sometimes be outperformed by manual strategy. Without the random seed placed, we obtained a few results where manual strategy outperformed. But the final portfolio value was always highest for learner. The majority of the outcomes were best for strategy learner.

5. EXPERIMENT 2

This experiment is to be performed to understand the effect of changing the value of impact for the strategy learner while computing portfolio values. Impact is the transactional cost incurred by the trader. Impact is the estimation of the price change that results due to the trade executed by trader, that is, if trader buys then the stock price is intended to go up and if the trader sells the price will go down. Thus, lower impact will be better for the trader.

HYPOTHESIS - "As the impact goes higher the performance of strategy learner will reduce, as a result, cumulative returns will go down and sharpe ratio will decrease."

Setup

The strategy learner is executed for symbol 'JPM' with start value of $100,000 on in sample period. The following steps are executed with different values of impact [0.05, 0.025, 0.01, 0.005, 0.0025, 0.0005] with commission set to 0.0.

Steps

1. All parameters initialised with values mentioned in Setup.
2. For each value of impact:
   i. Initialize StrategyLearner object sl.
   ii. Train the learner for in sample period - sl.add_evidence().
   iii. Test the learner for the same in sample period - sl.testPolicy()
   iv. Compute portfolio values using method compute_portvals() with the commission = 0.0 and impact set to current iteration value.
   v. Normalize the portfolio values (start with 1.0) and calculate the statistics (Cum return, Avg daily Ret, Sharpe Ratio, Std Dev.).
   vi. Store the stats and plot the portfolio values obtained.

The results obtained and plotted are shared below. The steps to generate the trade are described in section 3.

Results

Impact Values	0.05	0.025	0.01	0.005	0.0025	0.0005
Cumulative Return	-1.058	0.793	1.888	2.25	2.431	2.575
Average Daily Return	-0.029	0.001	0.002	0.002	0.003	0.003
Standard Deviation	0.457	0.012	0.009	0.008	0.008	0.008
Sharpe Ratio	-1.02	1.646	3.889	4.563	4.872	5.104
Final Portfolio Value	-5848.08	179322.585	288814.409	325047.623	343115.122	357545.63

Table 4- Experiment 2 Outcomes Statistics.

Statistics show that the hypothesis is true as the cumulative returns are decreased to a negative value of -105%, where as very low impact value of 0.0005 gives a 257% cumulative return. Similarly, sharpe ratio increases with such difference also. Hence, the initial hypothesis is correct. One interesting finding is that the Standard Deviation increases by a huge factor as the impact increases.

It can be observed that the difference in portfolio is increases exponentially as the impact value doubles. This is evident from the graph plotted in figure 4.

portvals(0.0005) - portvals(0.0025) >> portvals(0.025) - portvals(0.05)

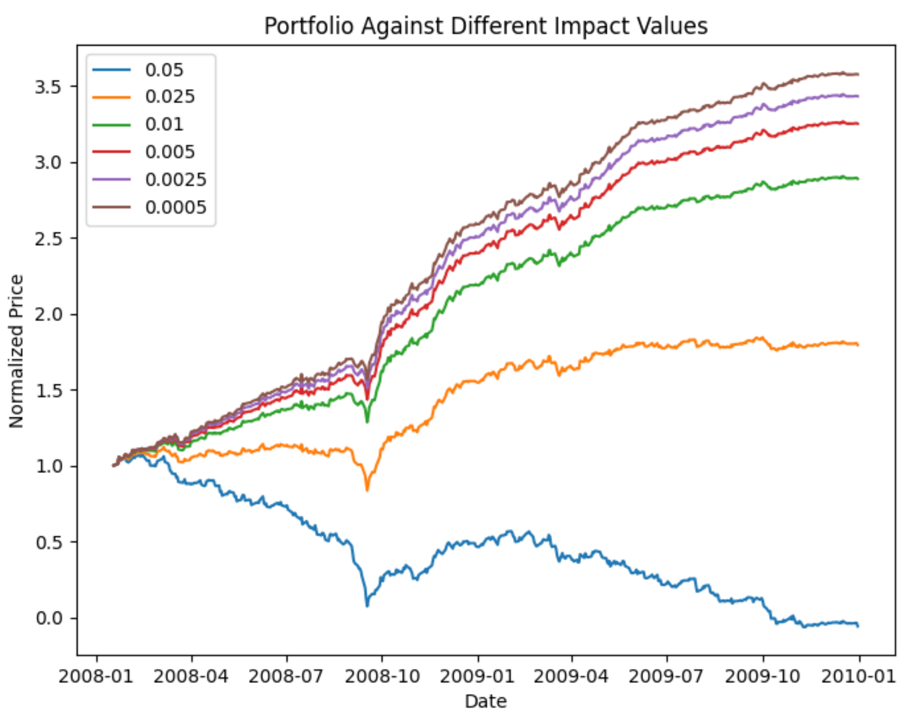


Figure 4- Comparison graphs for impact values in Strategy Leaner on in sample period.