CS 7646 - Project 8 Report:

Abhijeet Gaurav
agaurav@gatech.edu

Indicator overview:
I used below indicators to devise my Manual Strategy and Strategy learner:

1. Momentum: Momentum of a stock over n days is the return of a certain day compared to N days before.
   Momentum (n day) = (today price/price n days ago) – 1

2. Price/Simple Moving Average ratio: Price/SMA shows the prices of the stock relative to its mean value (simple moving average).

3. Bollinger Band percentage: Bollinger Band is calculated by first calculating SMA and then adding two standard deviations above and below the SMA. In our logic we used a 10-day window (lookback) for SMA. The standard deviation was calculated over the last 10 days.
   Upper Bollinger band = SMA + 2 x std
   Lower Bollinger band = SMA – 2 x std
   Bollinger Band Percentage is calculated as: (price–lower band)/(upper band–lower band)

   In both the strategies we optimized for parameters: momentum, Price/SMA ratio and bollinger band percentage. More details to be discussed below.

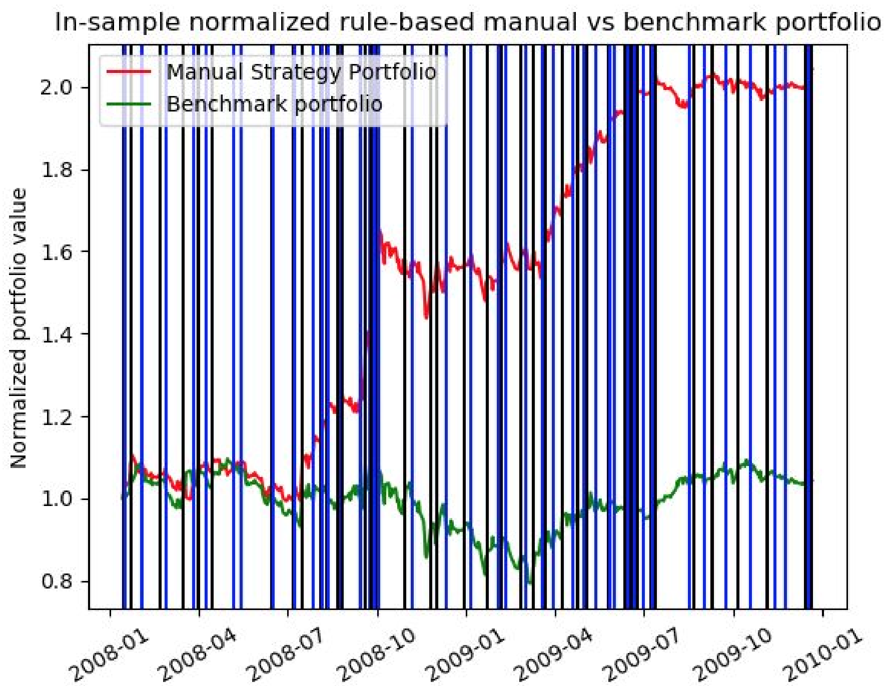
Manual strategy:
My manual strategy utilizes all the indicators mentioned above. They take into account the past price of the stock to determine whether the stock is being overbought or oversold. Below is the exact strategy I used:

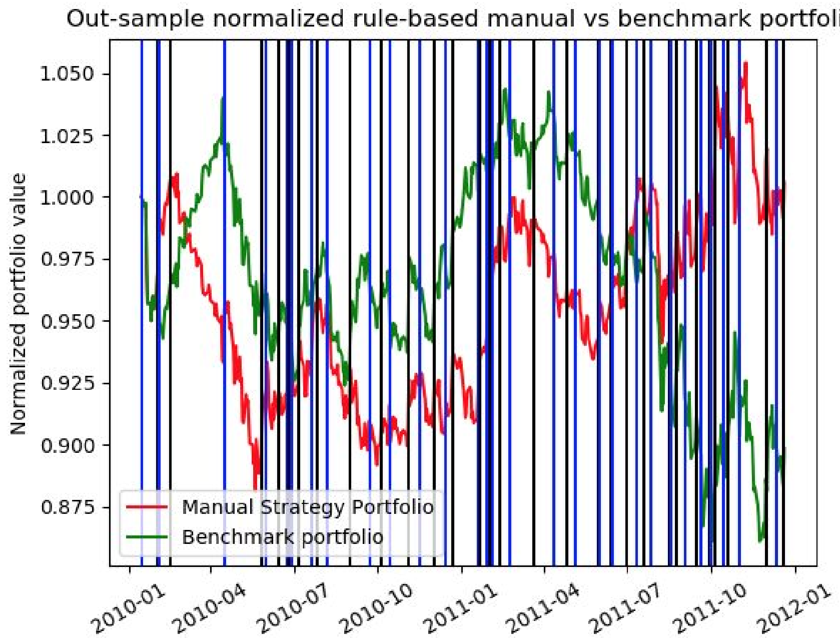
1. Oversold opportunity when any two of the below conditions is true:
   a. Bollinger Band Percentage <= 0.6
   b. SMA ratio < 1
   c. Momentum > 0.1
2. Overbought opportunity, when any two of the below conditions is true:
   a. Bollinger Band Percentage >= 0.82
   b. SMA ratio > 1 3
   c. Momentum < 0
3. If none of the 1. Or 2. is true, we maintain the current position and do not trade at all.

The threshold used above for each of these indicators are determined by trial and error on the in-sample dataset (JPM from 1st January 2008 to 31st December 2009) to maximize return.
The experiment and assumptions Trading details:

Date range: JPM, January 1, 2008 to December 31 2009 for in-sample.
Date range: JPM, January 1, 2010 to December 31 2011 for ou-sample.
Allowed position: 1000 shares long, 1000 shares short, 0 shares.
Impact: 0.005 ; Commission: 9.95
Starting value: $100,000

I found that the strategy was very effective for in-sample and it was okayish (positive return) forout-sample as can be shown below:





In-sample	Benchmark	Manual Strategy
Cumulative return	0.04190000000000005	1.0423
Std dev	0.016734748266475915	0.011648099318147826
Mean	0.00022342557133216622	0.0015315310859777702

Out-sample	Benchmark	Manual Strategy
Cumulative return	-0.10160000000000002	0.005800000000000027
Std dev	0.008611353892198395	0.008467164360565941
Mean	-0.00018295421064133355	4.765565042731636e-05

As can be seen above, for in-sample data, the strategy performs much better than the benchmark and does so consistently , but this success is not replicated to such an extent for out-sample data. The out-sample performance of the strategy is overall positive if we look at the cumulative return (while the benchmark is negative). I think the performance of in-sample was not replicated for the out-sample because the thresholds we're using to determine long/short position in the strategy was fine-tuned on the in-sample data, resulting in somewhat overfitting which can also be observed by looking at the density of the sell/buy vertical lines which is quite high as compared to out-sample.

Strategy learner:

For my strategy learner I have used Q-learner. The integral elements required by the Q-learner strategy are the states, actions and rewards. Basically, to determine those elements, first, it is necessary to frame the trading problem in terms of a learner trying to maximize the rewards for its actions. Therefore, using Q-learner for strategy learner is very intuitive because real-time trading involves taking risks in form of actions such as buy, sell and hold and rewards are received in terms of monetary profit/loss.

In my strategy, I define:

Q-learning actions as long, a maximum of 1000 shares, short, a maximum of 1000 shares shorted), and hold (keep the status quo for the current holding).

Q-learning rewards as: The daily return (next trading day compared to current day) as the result of an action taken today.

Q-learning states are given by the indicators of the stock on any given day. The indicators used to describe the states of the stock on any given day are the same as the one used in Manual Strategy which is Momentum, Price/SMA and Bollinger Band percentage. To turn the indicators into a **discretized state**, we first calculate each indicator for a given day.

Let's consider our three indicators Momentum with M, Bollinger Band percentage with B and Price/SMA as P. We separate the values of each indicator across all trading days being used into 10 bins.

Now, each indicator for a date is given by an index value from 0 to 10, corresponding to the bin value. Thus, now the discretized state of a trading day is then given by: 50*M + 10*P + 2*B. Since, each of the indicator can take 10 possible values (bin size = 10), we not have a total of 1000 states in the Q-learner

Experiment 1:

In the first experiment, we compare the performance of the Strategy Learner (based on Q-learner) and the manual strategy.

The experiment and assumptions Trading details:

Date range: JPM, 1st Jan 2009 to 31 Dec 2009.

Allowed position: 1000 shares long, 1000 shares short, 0 shares.

Impact: 0.005

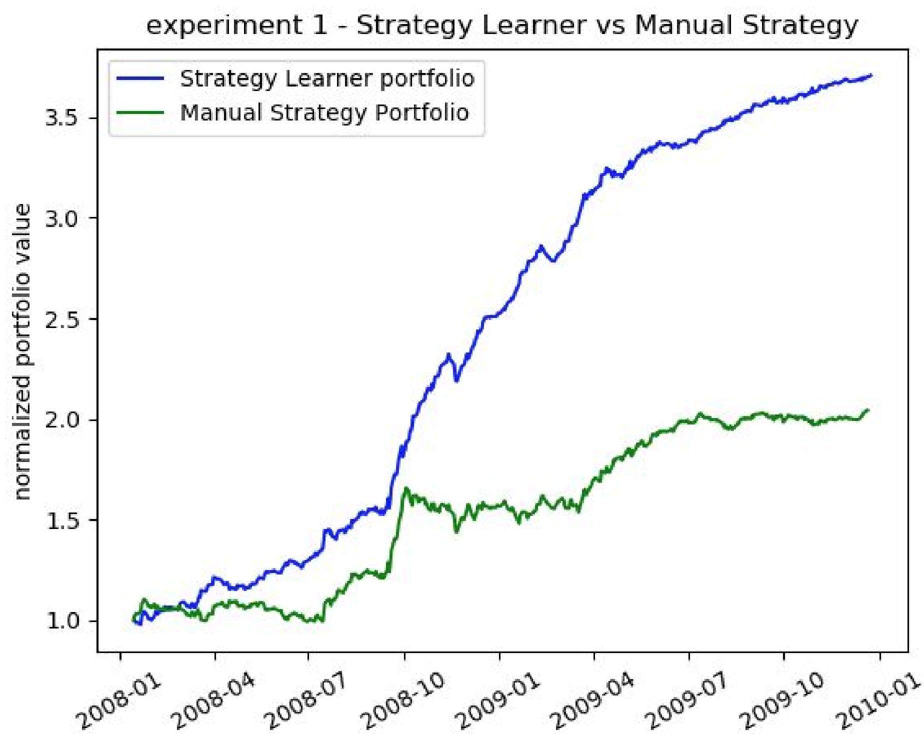
Commission: $0

Starting value: $100,000

Q-learner parameters: learning rate alpha 0.2, discount rate gamma 0.9, random action rate rar 0.5, random action decay rate 0.99, dyna 0.

Outcome:
The outcome of the experiment was that Strategy Learner greatly out-performs the Manual Strategy in terms of cumulative return. This can be expected for the in-sample period. This is because my Manual Strategy was manually fine tuned using trial and error to maximize returns. On the other hand, the Strategy Learner was trained with 10 discrete thresholds (bin size) for each indicator.

I expect this relative result every time with in-sample data because the Strategy Learner can overall detect more granular changes in the stock movement because it has 1000 states. Thus, for in-sample it'll show better performance and can even overfit.



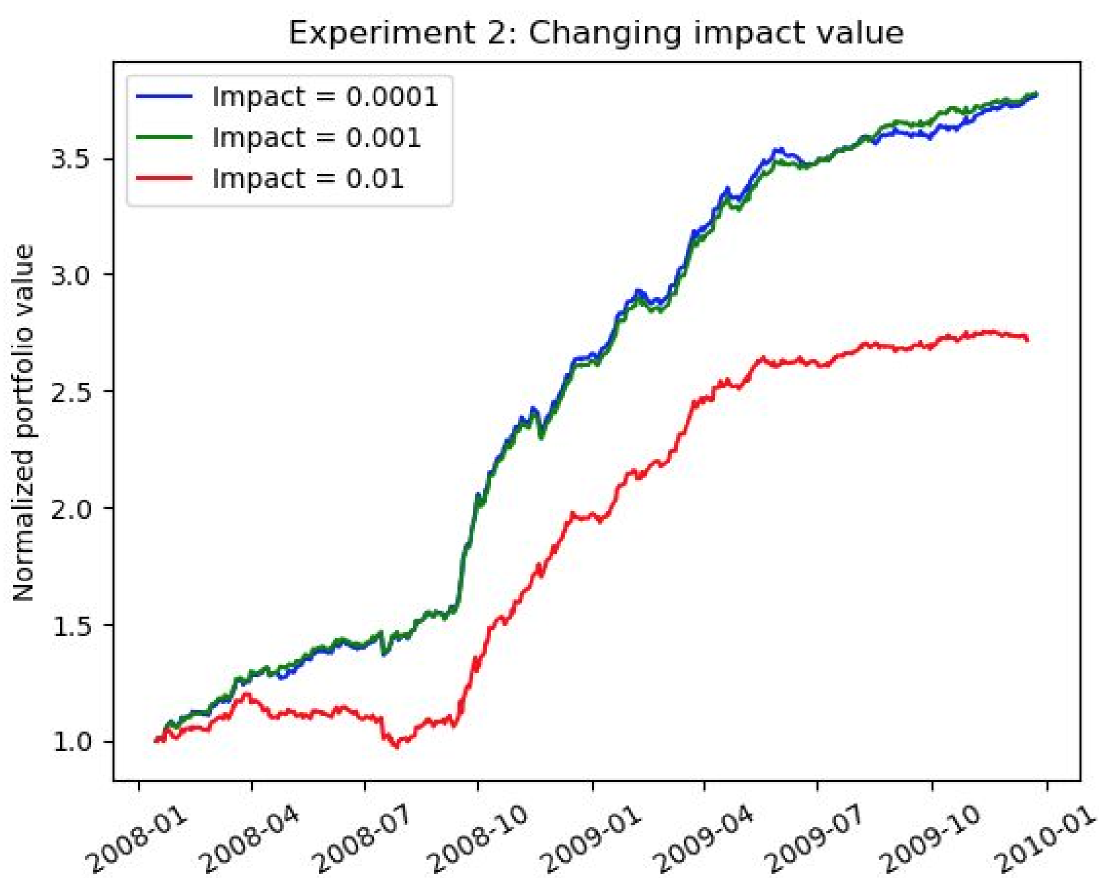
In-sample	Strategy Learner	Manual Strategy
Cumulative return	2.7058	1.0423
Std dev	0.008582901347324326	0.011648099318147826
Mean	0.0027133089167516175	0.0015315310859777702

Experiment 2:

Hypothesis: As the impact increases, the trade frequency and return decreases.
More: As the impact increases but remains below 10^{-2}, there is a very negligible (almost no) impact on the trading behavior. However, as the impact increases and comes close to 10^{-1}, the trade behavior starts getting a huge impact.
As the impact increases, the performance decreases as it is costlier to trade and make profit. Thus, the learner trades less frequently and is less likely to take advantage of small change in prices.

Experiment details:
Trading dates: JPM, 1st Jan 2009 to 31 Dec 2009.
Allowed positions: 1000 shares long, 1000 shares short, 0 shares.
Impact: Three different impact values are tested 0.0001, 0.001, and 0.01
Starting value: $100,000 Q-learner parameters: learning rate alpha 0.2, discount rate gamma 0.9, random action rate rar 0.5, random action decay rate 0.99, dyna 0.

Comparison:



The above figure shows the learner performance with respect to three different impact values: 0.0001, 0.001 and 0.01.
Higher impact value implies poorer performance. Also, as the impact increases, the learner trades slightly less frequently. This is because there will be less occasions where trading is less profitable due to the cost associated with trading (impact value), and therefore the learner would continue to hold the current position most of the time.

Impact	**0.0001**	**0.001**	**0.01**
Number of trades	197	195	83
Cumulative return	2.7661780138671492	2.7767111191319267	1.7186086219824706
Std dev	0.008070725186300845	0.008024124006067048	0.010969414571456946
Mean	0.0027422022392996166	0.002747549764211264	0.002124057929759635