Strategy Evaluation

Kangni Dang
kdang49@gatech.edu

Abstract—Automated Trading is an interested topic of many people. In this report, JPM stock data is chosen as the test data and three technical indicators are selected as my indicators and one reinforcement learning algorithm, Q-Learning, is used to explore my next trading actions.

1 INDICATOR INTRODUCTION

1.1 Technical Indicators Selection

In total there are 3 indicators used in my manual strategy and strategy learner.

First indicator: price/SMA. Simple Moving Average is the rolling average over a window of time of a stock and price/SMA is an indicator that measures how much the price has diverged from the rolling mean, n is the lookback period.

This indicator is defined as: ratio[t] = (price[t]/price[t-n:t]. mean()) -1

Second indicator: Bollinger Bands. The BBands values is composed of three values, SMA, SMA+2*sigma, and SMA-2*sigma, where sigma is the standard deviation of sma data over a lookback period.

The BBands is defined as: bbands[t] = (prices[t]-sma[t])/(2*sma[t]. std())

Third Indicator: Momentum. This describes how much the price has changed over a period and whether it is trending up or down. Momentum is defined as: momentum[t] = (price[t]/price[t-n])-1, where n is the lookback period.

2 MANUAL STRATEGY

By observing the indicator values, I manually discretize each indicator into different bins.

2.1 Discretize indicators

By observing the data, it is found that the price/SMA ratio normally fall into -0.5 to 0.5, momentum typically ranges from -0.5 to 0.5 and Bollinger Bands typically ranges from -1 to 1. So I first discretize the indicators and then generate signals based on discretized indicators.

When price/SMA ratio is above 0.5, it is regarded as a short signal, when price/SMA ratio is less than – 0.5, it is regarded as a long signal.

Similarly, when momentum value is above 0.5, it is regarded as a short signal and when it is less than -0.5, it is regarded as a long signal.

When bbands values is above 1, It is regarded as a short signal and when it is less than -1, it is regard as long signal.

Table 1 — Technical indicators

Technical Indicator	**Long/Short/Hold**
price/SMA	*_sma_ls*=1/-1/0
momentum	*_mtm_ls*=1/-1/0
bbands	*_bbands*_ls=1/-1/0

2.2 Overall Signal

For each technical indicator, since we use 1, -1 and 0 to mark long, short and hold, when sum of the three indicators larger or equal than 2, then the signal for the underlying stock is long. Thus we can generate our manual signals. When sum of the three indicators smaller or equal than -2.5, then the overall signal for the underlying stock is short.

$$sum_{ls} = \_sma_{ls} + \_mtm_{ls} + \_bbands_{ls}$$

Table 2 — Overall signal

sum_{ls}	Signal
>=2	*long*
<-2.5	*short*
else	*hold*

2.3 Compare manual strategy and benchmark

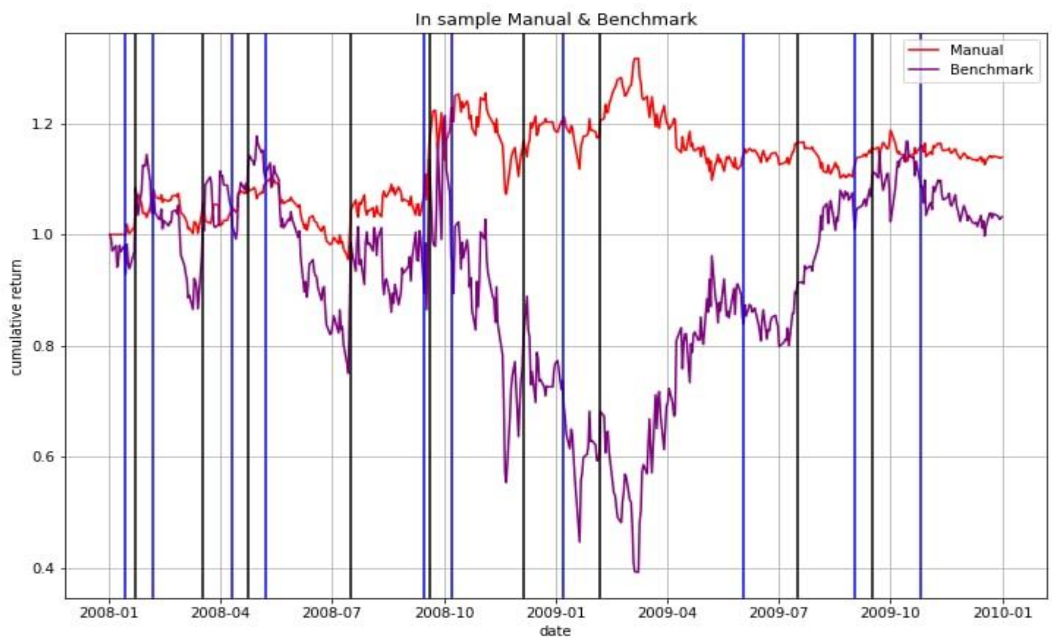

Figure1 - Compare in sample Manual and Benchmark performance

Here the benchmark is holding 1000 JPM stock data from beginning to the end. The benchmark portfolio value and manual strategy portfolio value are both normalized with starting value equal to 1. It can be observed that the during the in sample and out of sample period, the manual strategy performs better and is less volatile than the benchmark values although manual's performance during in sample period looks better.

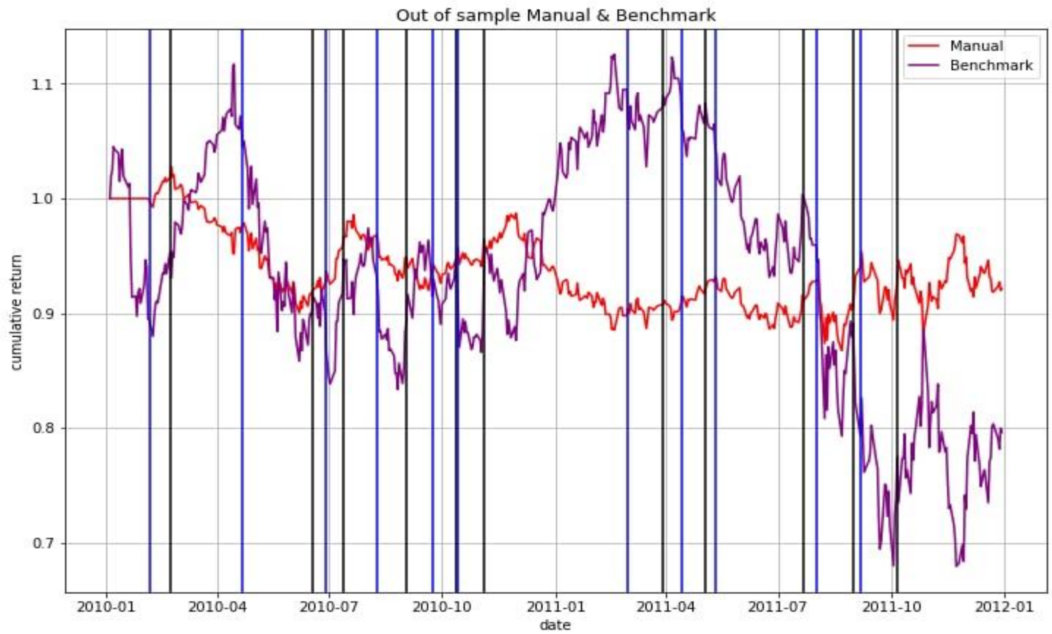

Figure2 - Compare out of sample Manual and Benchmark performance

As for the difference between in sample and out of sample result, the cumulative return of in-sample and out-of-sample period are quite different. Since the benchmark always follows the stock market when market down the benchmark will get down. While for manual strategy when market goes down, we will short the stock therefore we can see that the manual return is more stable and better than benchmark.

Table 3 – Performance of Manual Strategy and Benchmark

	Manual Strategy	**Benchmark**
Cumulative return	0.13913	*0.03197*
Std of daily return	0.01400	*0.05214*
Mean of daily return	0.00035	0.00140

3 STRATEGY LEARNER

Q-Learning is an algorithm used to help generate signals/decide next trade action. The add_evidence function is used to train this learner while the testPolicy function is used to test it against new data.

Based on the discretized indicators, we have 125 possible states (since each indicator has 5 states). The Q-Learning will study the return in each state by taking different actions (actions are controlled by parameters so initially the actions are random but will tend to choose actions bring more return as experience increase). There are 3 actions in total and reward is computed as daily return of the current position. If the agent take long action and price goes up, a positive reward will be received, while if the price goes down, a negative reward will be received and will reinforce the fact that long action is not the best action.

3.1 Discretize indicators

By exploring the indicator distributions, it is found that the price/SMA ratio normally fall into -0.5 to 0.5, momentum typically ranges from -0.5 to 0.5 and Bollinger Bands typically ranges from -1 to 1.

For each technical indicator, to use Q-Learner explore next actions, for each indicator, five different bins were used for discretization. For price/SMA and momentum indicator, the five bins are:

0: x < -0.5 ; 1: -0.5 <= x <=0 ; 2: 0< x<= 0.5; 3: x>0.5 4: x==NaN

For Bollinger Bands, the five bins are similar:

0: x < - 1 ; 1: - 1 <= x <=0 ; 2: 0< x<= 1; 3: x>1 4: x==NaN

3.2 Hyperparameters

Table 4 - parameters

Parameter	**Value**	**Description**
Number of States	*125*	*Total states the learner has*

Parameter	Value	Description
Number of actions	*3*	*Long/Short/Hold*
Alpha	0.2	*The learning rate*
Gamma	0.9	*The discount factor*
Epsilon	0.5	*The random action rate*
Epsilon Decay	0.99	*The decay factor for the random action rate*

Here alpha is our learning rate and is how much we weigh more recent experience when updating our Q-value; Gamma is the discount factor and quantifies how much importance we give for future rewards. Epsilon introduce randomness into the algorithm and force us to try different actions and epsilon will decay when experience increase. Overall, we update our Q-table like this:

$$q[s,a] = (1 - alpha) * q[s,a] + \ alpha \ * \ (r \ + \ gamma \ * \ max(q[s\_prime]))$$

4 EXPERIMENT 1

It can be seen from the in-sample comparison that the manual strategy performs best among the three values. The Q-Learner performs a bit better than the benchmark and more stable than benchmark, which is expected to happen because for manual strategy we already have a criterion for trading while for Q-Learner since it starts to learn from zero so it is normal that it will perform less better than manual strategy.

Since experience of learner increase, it can be seen from the figure below that the Q-Learning agent is able to produce a strategy that can yield better results during the out of sample period than in in-sample period.

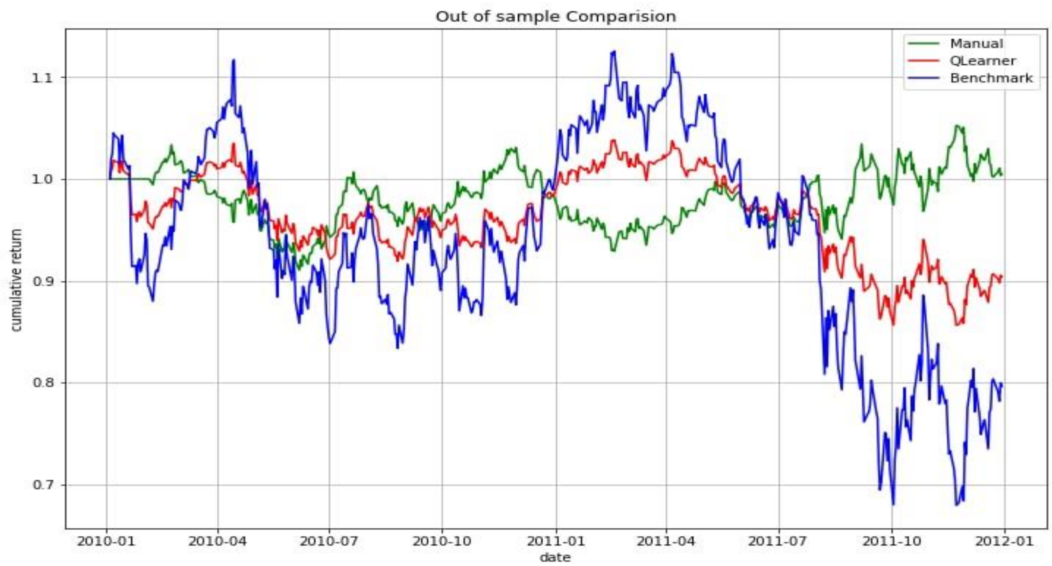

Figure3 - Compare out of sample Manual , Q-Learner , Benchmark performance

It can be seen from the out-of-sample result that the benchmark is the most unstable one while Q-Learner and manual are more stable and with smaller volatility.

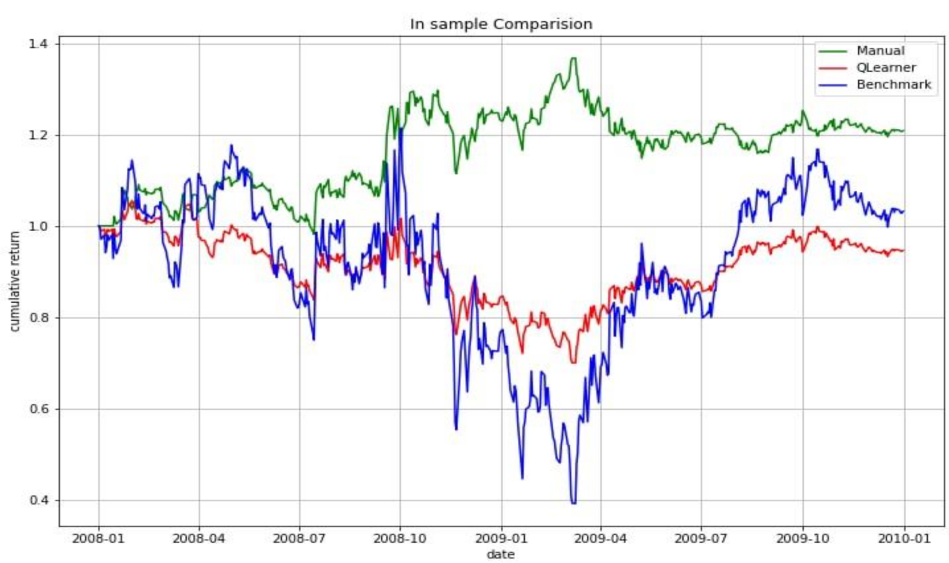

Figure4 - Compare in sample Manual, Q-Learner, Benchmark performance

5 EXPERIMENT 2

Figure5 – different impact values' influence on learner

Market impact is an important factor when simulating transactions. In this experiment the effects of the market impact were studied to validate our hypothesis: the impact encapsulates the overall fluctuation of the market; movements will affect one's portfolio.

Here the impact value was changed for our built strategy learner. We can see from the chart that when impact value increase, the portfolio value will decrease accordingly. We can see that the impact will directly affect the learned policy.