Sergiy Palguyev

spalguyev3

903272028

Machine Learning for Trading – Strategy Learner Project

Description:

The base problem to be solved is whether to BUY or SELL certain amount of stock in order to gain value in a portfolio. This problem is then described as a Machine Learning (ML) problem and is formulated in such a way as to use QLearning strategy to determine the solution. The QLearning strategy uses available, past, data to train the QLearner in order to build a QTable which may be used on future data.

Technical indicators are used to determine and discretize available data into training samples. Discretization is necessary for the QLearner to understand discrete states that it is learning on. The technical indicators used are Short SMA, Long SMA, Upper and Lower Bollinger Bands, and Momentum values. These indicators are discretized into a finite set between 0 and 9 (10 steps). The states are calculated as the addition of each technical indicator, each multiplied by an increasing multiple of 10. Since multiple indicators are used, the entire value set is divided by 1000 and rounded down in order to get a valule set between 0 and 100. This is appropriate in order to further discretize the final State values. The difference in States is preserved even as the value is decreased with division. The formula is as follows:

$$floor\left(\frac{sma_{long} * 10000 + sma_{short} * 1000 + bb_{up} * 100 + bb_{down} * 10 + momentum}{1e3}\right)$$

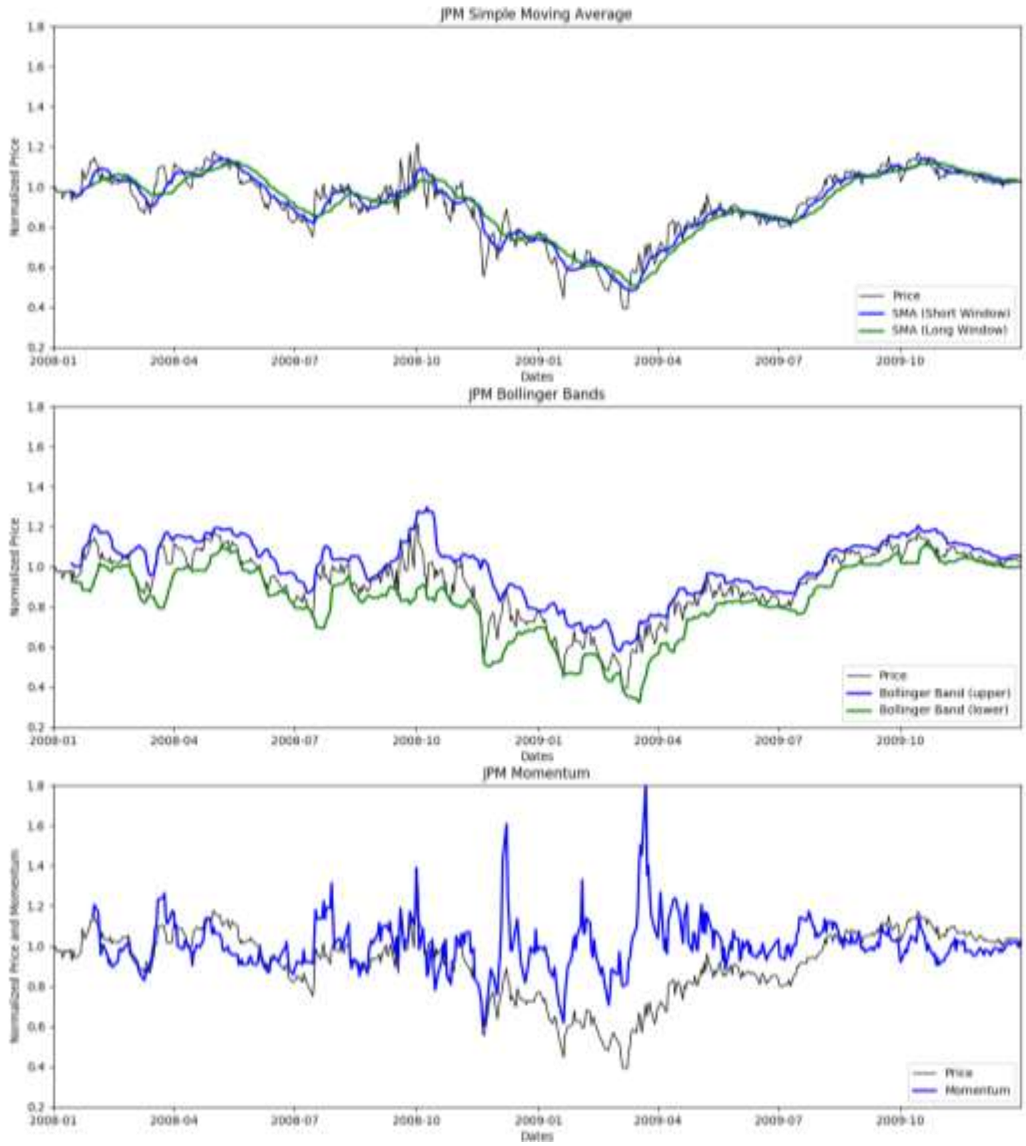
Considering the values, the number of states set for the QLearner is set to 100 and the actions are set to 3. We are interested in only 3 actions for our trading problem as only 3 possible action may be taken per stock trade. The actions may be BUY(1), SELL(2), or HOLD(0). As per the Manual Strategy, the maximum number of shares allowed to be traded is up to 1000 shares long or -1000 short. The training logic for the learner then ends up as:

If holding = -1000 and action is HOLD, do nothing.
If holding = -1000 and action is LONG, do nothing.
If holding = -1000 and action is SHORT, earn 2x rewards and hold = 1000.
If holding = 0 and action is HOLD, earn reward and hold = 1000.
If holding = 0 and action is LONG, earn reward and hold = 1000.
If holding = 0 and action is SHORT, earn reward and hold= -1000.
If holding = 1000 and action is HOLD, do nothing.
If holding = 1000 and action is LONG, earn 2x rewards and hold = 1000
If holding = 1000 and action is SHORT, do nothing.

Iterating the learner through 100 loops guarantees that the QTable is updated to the optimal values.

Experiment 1:

For this experiment, none of the indicators were changed from the Manual Strategy project. Namely, the indicators of long and short SMA, Bollinger Bands, and Momentum were used for conducting both the Manual Strategy and comprising the state for the QLearner.



The result of comparing the QLearner against the Manual Strategy yields a striking difference in the success of the two strategies. While the Manual Strategy fairs well against the Benchmark, the QLearner is able to leverage the training in order to largely outperform both the Benchmark and the Manual Strategy.

Cumulative Return of QLearner: 0.7107
Cumulative Return of ManualStrategy: 0.2366
Cumulative Return of Benchmark: 0.0123
Mean Daily Return of QLearner: 0.00114764151157
Mean Daily Return of ManualStrategy: 0.000504055458862
Mean Daily Return of Benchmark: 0.000168086978191
Standard Deviation of daily return of QLearner: 0.0128117177035
Standard Deviation of daily return of ManualStrategy: 0.0121399938874
Standard Deviation of daily return of Benchmark: 0.0170043662712

This result would be expected every time. Since the QLearner is learning and trading over the same set of data, it should learn the optimal actions that must be taken for a given state in order to gain a positive return. This should always fare better than the Manual Strategy, since the QLearner has the benefit of multiple iterations of learning, whereas the manual Strategy does not.

Experiment 2:

Impact is the effect that acting upon market conditions effects the trader. In theory, the higher the impact, the more the trader is affected by the trades he/she makes. I predict that the QLearner should still do better than the Manual Strategy, since the trades ordered by the QLearned are optimized for best returns, while the Manual Strategy has no training and will make as many trades as set up by the technical indicator logic. However, as the impact increases, the returns by the QLearner should decrease.

The impact values under test are arbitrarily chosen as 0.002 and 0.007 (based around the default of 0.005).

As observed by the graphs, the greater impact has a negative effect on the QLearning strategy. This is expected per the theory outlined above.

Conclusion:

In conclusion, it is evident that the QLearner strategy is better than the Manual Strategy, which itself is better than the Benchmark. With the caveat of impact having a negative effect on the returns from the QLearner Strategy, it is still a much better performing strategy to utilize machine learning (i.e. Q-Learning) rather than Manual Strategy techniques. Additionally, both Manual and QLearning strategies demonstrate that the weak version of the Efficient Market Hypothesis is not true, since both strategies are capable of beating the market.