# Evaluation of Reinforcement Learning Techniques for Trading on a Diverse, Portfolio

**Ishan Khare**
Stanford University
iskhare@stanford.edu

**Tarun Martheswaran**
Stanford University
tarunkm@stanford.edu

**Jonah Ezekiel**
Stanford University
jezekiel@stanford.edu

**Akshana Dassanaike-Perera**
Stanford University
akshana@stanford.edu

## Introduction

This project aims to develop an optimal policy of trading signals for algorithmic trading by comparing on- and off-policy reinforcement learning techniques. Specifically, we are implementing Value Iteration (VI), State–action–reward–state–action (SARSA), and Q-learning algorithms to determine when to buy/sell our shares of a set of stocks. Using VI and SARSA, we aim to learn the most optimal on-policy trading strategy given a policy designed by hand. Using Q-Learning, we aim to learn the most optimal policy for algorithmic trading. We also have trained our models on our set portfolio over multiple years of data, and subsequently tested it in an attempt to determine a policy ROI estimate. After obtaining this evaluation metric, we can compare with the S&P 500, our selected baseline oracle. Overall, we aim to answer the following questions this updated report:

- Which reinforcement learning technique, on or off policy, is able to generate a higher ROI? Why may this be?
- Should we be using reinforcement learning rather than holding S&P 500? In which situations is this true/false?
- What further hyperparameters changes should we make in order to further increase this ROI?

## Literature Review

In the realm of on-policy approaches, such as Value Iteration, Sumit Rawat applied a on-policy based technique using the Bellman optimality equation. As he knew that evaluation at every step may not be necessary, he elected to stop after just one sweep (Rawat, 2005). Additionally, Lee utilized Generalized Policy Iteration (GPI) to find an evaluation-based optimal policy (Lee, 2001).

Within off-policy approaches, which have been more popular recently, we see several studies starting to implement Q-learning. Sun and colleagues found that a combination of Mean Absolute Deviation (MAD) and Q-learning performs better than other common reinforcement learning techniques. Specifically, he used epsilon-greedy after pre-processing with MAD. (Sun et al., 2023). Also, Chakole and colleagues used Q-learning to demonstrate its outperformance of the Buy/Hold strategy and Decision-Tree strategies (Chakole et al., 2021).

However, there is research of techniques that combine both on and off-policy approaches. Alimoradi adapts a league championship algorithm (LCA), hybridized with both Sarsa and Q-learning (Alimoradi and Kashan, 2018). Using a combination of these techniques, the researchers found that they were able to outperform evolutionary programming and Buy & Hold strategies when trading almost all companies.

Since our last literature review, we have officially decided that our project will be focused on trading ETFs, specifically SPY. So, we start to search for articles that traded SPY with reinforcement learning. Yang and his coauthors explored a reward-based inverse reinforcement learning technique to extract trading signals. Similarly to us, they used the buy and hold SPY ETF as a benchmark. They were able to outperform this based on signals from new sentiment (Yang et al., 2018). Based on research like Yang's, we would like to see how our tests of both off and on policy reinforcement learning techniques compare.

## Dataset

For the purpose of our project, we decided to connect to the Yahoo Finance API through Python, in order to get a time series of data for a certain ticker. Luckily, through the "yfinance" library the data is collected in an easy-to-use fashion, so pre-processing was not a significant consideration. The information that we chose to use for each day was simply Open & Closing Price, as well as High and Lows.

After obtaining dataframes of information on the price of SPY from 1980 to 2023, we moved forward to apply our techniques of Value Iteration, SARSA, and Q-learning to develop actionable policies of trading signals.

## Baseline

Many are familiar with the S&P 500 index. This is a popular index that represents the general trend of the stock market at any given point. In past research works, people have tried to develop algorithmic techniques that output signals to game the S&P 500. While the common economic theory is that a stochastic set of choice is likely to do better, we hope to see if a carefully designed artificial intelligence algorithm can refute this claim. Can we in fact define and evaluate a policy that creates actionable insights on the market? Is it better than the most simplified method to track the market?

As an update to our previous project proposal, we have done extensive research to conclude that we will also use the "Buy & Hold Strategy" as a baseline. This strategy is based off the saying that "time in the market beats timing the market", which comes from the belief that the market will generally increase over longer periods of time, so holding a stock is advantageous.

Additionally, the models described in the Literature Review provide another set of baselines. These, alongside several other models that researchers have developed have similar evaluation metrics to the ones we have elected to use. Our project aims to learn an optimal policy for only trading SPY, so comparison against the results of multi-stock models should be interpreted in consideration of that context.

## Main Approach

As mentioned in previous sections, we plan to implement and test these three common reinforcement learning approaches on trading company stock. We aim to use VI, SARSA, and Q-Learning to develop an optimal trading policy for trading SPY, an ETF which tracks the S&P 500. We are only trading one ETF to provide clear analysis across the three methods that we are testing by eliminating confounding variables that arise when trading multiple stocks.

As VI and SARSA is on-policy, we need to have a policy before entering the market. Following the motto "buy low, sell high", we follow the straightforward policy:

- If the price we purchased is lower than the current price, choose a random number of shares to sell
- If the price we purchased is higher than the current price, choose a random number of shares to buy

Now, for our off-policy approach we chose to implement Q-learning to our financial data. Here, we do not need a specific policy to test, instead the algorithm learns as it goes through the data, finding the $Q_{opt}$ value we discussed in lecture.

In terms of data, we will do two sets of training/testing as follows

- 2000-2021 for the "SPY" ticker as our training set, and subsequently test on 2021-2023.
- 2000-2016 for the "SPY" ticker as our training set, and subsequently test on 2016-2023.

This is to see if including the COVID period in our training data has an impact on our model. We hypothesize that it will, as the events that occurred were unpredictable and the on/off policy approaches may not work as expected.

## Evaluation Metric

To evaluate the performance of our trading strategies in both the on/off policy approaches, we elected to use traditional profit and loss (PNL) as our evaluation metric. We focus on the total profit and loss over the timespan that our model runs, with the amount of starting capital constant. We also plan to track PnL over each day and subperiods, to see if we should focus on specific time spans.

## Results and Analysis

As demonstrated across the two graphs, including market data from COVID in the training dataset leads to superior results compared to our baseline. This is likely due to the fact that the unprecedented conditions of COVID led to unpredictable market conditions and behavior. During the beginning COVID, there was a rapid market crash, followed by an even greater tech-stock-led boom that was not seen in earlier training data. As a result, as demonstrated by Figure 1, holding the S&P 500 beat out all of our reinforcement learning strategies when COVID was excluded from our training data (and included in our testing data). However, when we included COVID in our training data (and excluded it from testing), all models beat the baseline. This may be attributed to the increased robustness that the data from COVID provided and also the stagnation of the S&P 500 during that time.

When comparing all three reinforcement learning strategies, we can see multiple takeaways. First, although based on simple principles, the “buy low, sell high” policy behind VI and SARSA proves to be fairly robust. In both experimental runs, regardless of whether COVID was included or not, Q-Learning leads to the greatest portfolio value during testing. Because Q-Learning is an off-policy algorithm, it is able to learn a more optimal policy than the

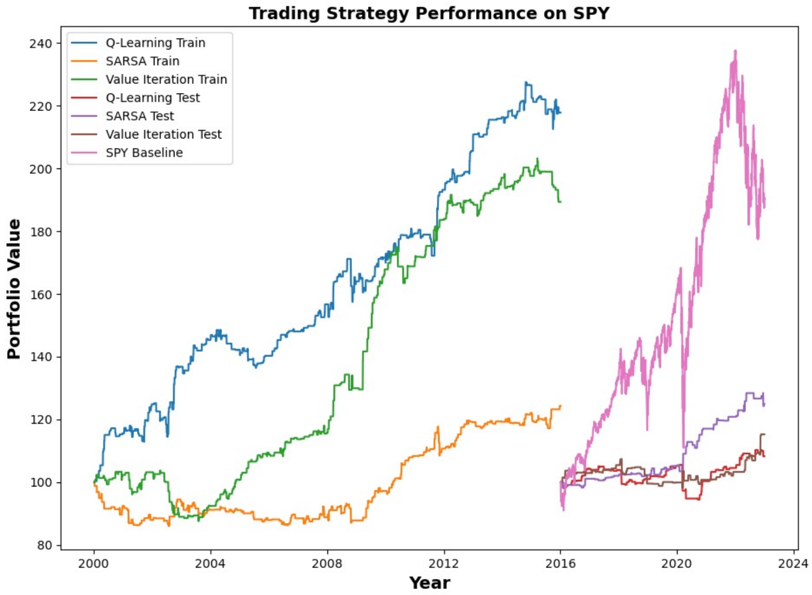

Figure 1: This figure displays our results when training our models on the years 2000 - 2015 and testing from 2016-present day. **Note:** we include COVID years in our testing data and exclude COVID years in our training data.

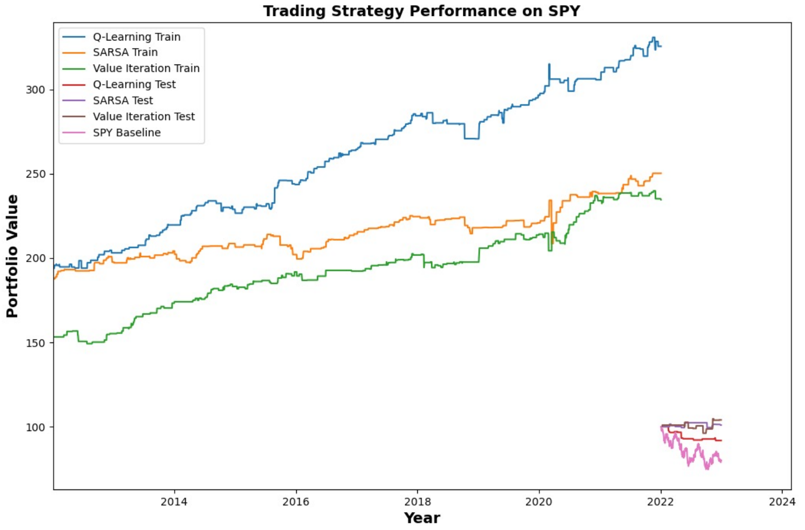

Figure 2: This figure displays our results when training our models on the years 2000 - 2022 and testing from 2022-present day. **Note:** we exclude COVID years in our testing data.

provided policy for VI and SARSA that was hand-designed. However, during our testing phase, Q-Learning performs the worst out of the three models.

We attribute this to the nature of the task at hand. During testing, we do not update $Q_\pi$, which means that Q-Learning does not "learn" as time progresses. As such, Q-Learning is unable to adapt the specific policy it learned under previous market conditions to the new conditions it experiences during testing. On the other hand, because VI and SARSA are on-policy and utilize a simple, fixed model, they are able to perform better than Q-Learning as they generalize better to unseen training data. This dynamic between learning algorithms is likely due to the bias-variance tradeoff, where the simplistic policies of VI and SARSA generalize far better when the models are unable to "learn".

That being said, its important to recognize that both experimental runs test on modern market conditions, which are quite unique to that of past conditions. Although most COVID restrictions have been eliminated, the economy has not fully recovered since COVID. Many communities are still feeling the impact. Even for large companies, COVID-era trends still have not settled to a post-pandemic world, such as the dynamic between online streaming and in-person movies, online shopping and in-person shopping, etc.

As such, our results may not be fully generalize to future market conditions that are more stable, but, for conditions that are similar to that of our training data, we hypothesize that our Q-Learning will perform quite competitively.

Revisiting the three questions we originally sought out to answer:

- **Which reinforcement learning technique, on or off policy, is able to generate a higher ROI? Why may this be?**

  During training time, Q-Learning performs far better than VI and SARSA. However, during testing, SARSA performs the best and VI outperforms Q-Learning. We attribute the performance drop of Q-Learning to the bias-variance trade off. We hypothesize that SARSA performs better than VI because it also optimizes on next-action, which essentially adds another layer of depth in its optimization.

- **Should we be using reinforcement learning rather than holding S&P 500? In which situations is this true/false?**

  Based on our results, we conclude that holding the S&P500 is preferable during times of large uncertainty or new market conditions, such as during COVID, whereas reinforcement learning is preferable during market stagnation.

- **What further hyperparameters changes should we make in order to further increase this ROI?**

  To further increase this ROI, we should focus on weighing near-term rewards, as market predictability drastically decreases over larger time horizons.

## Future Work

Moving forward, there's multiple experiments we would like to perform. First, for Q-Learning, we would like to experiment with updating our Q-Learning policy during testing as well and compare it to the S&P 500 baseline. Although this wouldn't following testing as defined in a supervised learning context, it may make more sense in a reinforcement-learning context to have our model constantly updating itself as it encounters new data.

Additionally, an interesting experiment would be to trade a diverse set of individual stocks. As of now, we have only tested on the SPY, and have received promising results. How may our policies be able to exploit the market conditions of individual stocks to outperform a large index?

Finally, another possible avenue of research would be to train our models on an alternative measure of price. Currently, our models are being trained on the percentage change of the closing price. Instead, we could train our models on other economic indicators such as the 40-day Simple Moving Average or the Relative Strength Index.

## Video Link

Our video is linked here: `youtu.be/xv5GBr_iHDo`.

## Code

Our GitHub is linked here: github.com/jonah-b-ezekiel/CS221.

## Non-Private Data

We downloaded market data from Yahoo! Finance's API linked here which is open-source and free to use: pypi.org/project/yfinance.

## References

M. R. Alimoradi and A. H. Kashan. A league championship algorithm equipped with network structure and backward q-learning for extracting stock trading rules. *Applied soft computing*, 68:478–493, 2018.

J. B. Chakole, M. S. Kolhe, G. D. Mahapurush, A. Yadav, and M. P. Kurhekar. A q-learning agent for automated trading in equity stock markets. *Expert Systems with Applications*, 163:113761, 2021.

J. W. Lee. Stock price prediction using reinforcement learning. In *ISIE 2001. 2001 IEEE International Symposium on Industrial Electronics Proceedings (Cat. No. 01TH8570)*, volume 1, pages 690–695. IEEE, 2001.

S. Rawat. *Stock market prediction using reinforcement learning*. Utah State University, 2005.

Y. Sun, M. Gao, C. Yang, D. Yuan, P. Zhu, H. Dong, and N. Zhou. Research on stock trading prediction based on MAD and Q-learning. In Y. Zhong, editor, *Fifth International Conference on Computer Information Science and Artificial Intelligence (CISAI 2022)*, volume 12566, page 1256649. International Society for Optics and Photonics, SPIE, 2023. doi: 10.1117/12.2668175. URL `https://doi.org/10.1117/12.2668175`.

S. Y. Yang, Y. Yu, and S. Almahdi. An investor sentiment reward-based trading system using gaussian inverse reinforcement learning algorithm. *Expert Systems with Applications*, 114: 388–401, 2018. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2018.07.056. URL `https://www.sciencedirect.com/science/article/pii/S0957417418304810`.