**Title:** Can Reinforcement Learning Algorithms Be Used to Create a Better Stock trader?

**Abstract**

Reinforcement learning methods in stock training environments could help offer a way to see practical applications for reinforcement learning firsthand, and help reinforce the potential benefits of algorithms. Many can find themselves puzzled by stocks, and worried about investment. Could a robot be trained on stock market to improve gains, and beat the stock market?

For this, I studied and worked on performing a methodological study on Reinforcement learning algorithm’s performance on the Stock market. Both to strengthen and learn in avenues where I had struggled before, but also to gain more understanding about Tensorflow and the various ways potential stock reinforcement methods could be applied, (as well as learning to set up package dependencies such as past Python 3.7 packages and specific packages to fulfill dependencies). The studies itself helped me learn how to set up packages where I had previously struggled.

The methods that were studied were DQN, DDQN,  as well as Proximal Policy Optimization (PPO), Advantage Actor Critic (A2C), and Deep Deterministic Policy Gradient (DDPG) methods and ensemble Strategies using stable baselines.

Various states could then be compared, such as the training time comparison between multiple baseline models, and the gains (or losses!) each program might be tested on or compared with vs the baseline offerings or starting investment. These studies began with the intent to compare the effectiveness of different reinforcement learning algorithms, and see which approaches found success in their studies, as well as noting training times, and size of datasets.

**Intro**

**What and Why:** I wanted to gain a better understanding of Reinforcement learning algorithms, improve where I had struggled, and learn how to run the packages and fix mistakes I had earlier. I wanted to apply

Problem statement: How can we use Reinforcement learning to maximize the gains of Stocks?

Given: find: such that: (solving a problem):  
Given a starting sum of cash, I wanted to study a series of different algorithms, as well as modify the training data and learning rates to study how it could effect or improve the algorithm to optimize the maximum gains and try to improve for the most return over a baseline.

question if a methodological study: Which algorithms produced the most return, as well as what were the differences between them and the training required per episode? ( The data will support gain from the start. Although no baseline for a buy and hold

**Methods**

**RL method applied:** DQN, DDQN, Proximal Policy Optimization (PPO), Advantage Actor Critic (A2C), and Deep Deterministic Policy Gradient (DDPG)

**Results**

I was able to backtest the results of the first run back against the baseline system, where the program recorded it’s high return of the ensemble strategy, and was about to achieve a return of about 1.64x (1.64 Million dollars), vs the about 1.38x (1.38 Million dollars) of it’s run’s baseline.   
Chart, line chart, histogram

Description automatically generatedChart, histogram

Description automatically generated

The program can compare it’s cumulative returns volatility next to the benchmark.

![Table

Description automatically generated with medium confidence]()

**The average return per year averaged 12.087%, and the cumulative return over all the years summed up to 64.332% for the final model baseline. Volatility was rated at 7.672% and The sharpe ratio (analysis of investing risk/volatility) averaged 1.53.**

**Conclusions:**![Table

Description automatically generated]()

Reinforcement learning stock trading methods can be applied to increase the consistency of gains through the use of algorithms, but there is still the volatility of the stock market at risk.

The various algorithms used were able to increase the return over the 38% baseline interest by rates of +10-40% above the return of the baseline metric.

![Chart

Description automatically generated]()

Sources:

1st Baseline: <https://github.com/Albert-Z-Guo/Deep-Reinforcement-Stock-Trading>  
2nd Baseline: <https://github.com/AI4Finance-Foundation/Deep-Reinforcement-Learning-for-Automated-Stock-Trading-Ensemble-Strategy-ICAIF-2020>