Trading for Gold

Developer - Erik Johnson

Goal

Through reinforcement learning, create and train an agent that will buy and sell in-game items to create maximum profits.

Project Summary

Why?

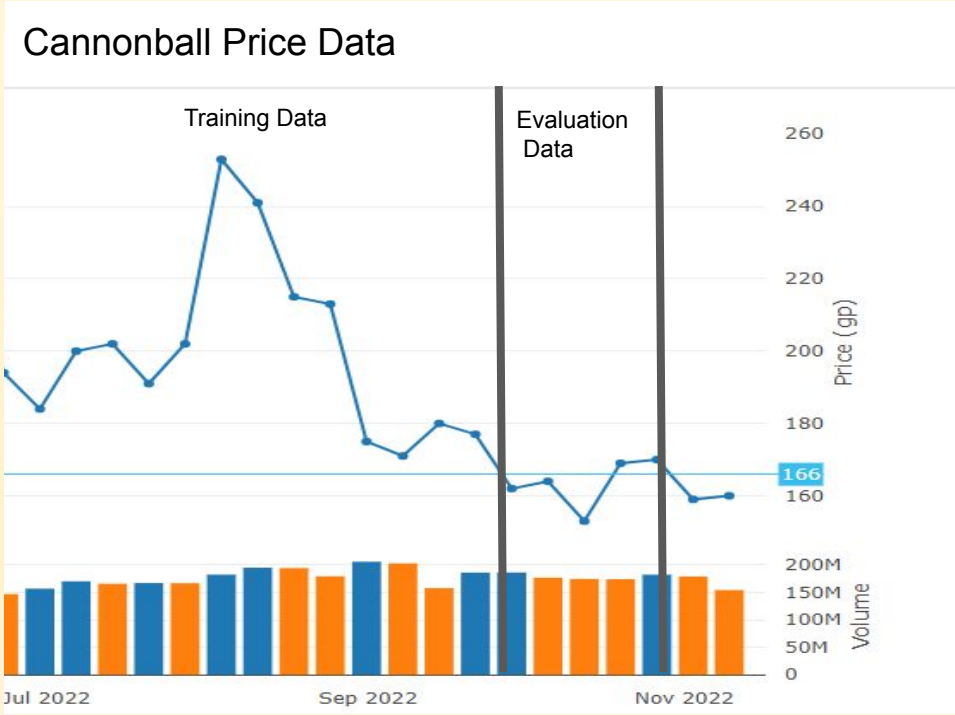
- Learn short term trading strategies
- Monetary Gain

Steps Taken

- Adapt Open-Gym environment - (started from stock-trading-env)
- Create TD agent
- Create DQN agent - Modified code - (https://github.com/viuts/q-trading-pytorch)
- Tuning agents

Cannonballs

- Thanks to Brownsey for the data
- Data contains 37,000 price points
- taken at 5 minute intervals
- July to end October
- (27,000 for training, 10,000 for evaluation)
- Training Data is approx 3 months of the item's price/volume



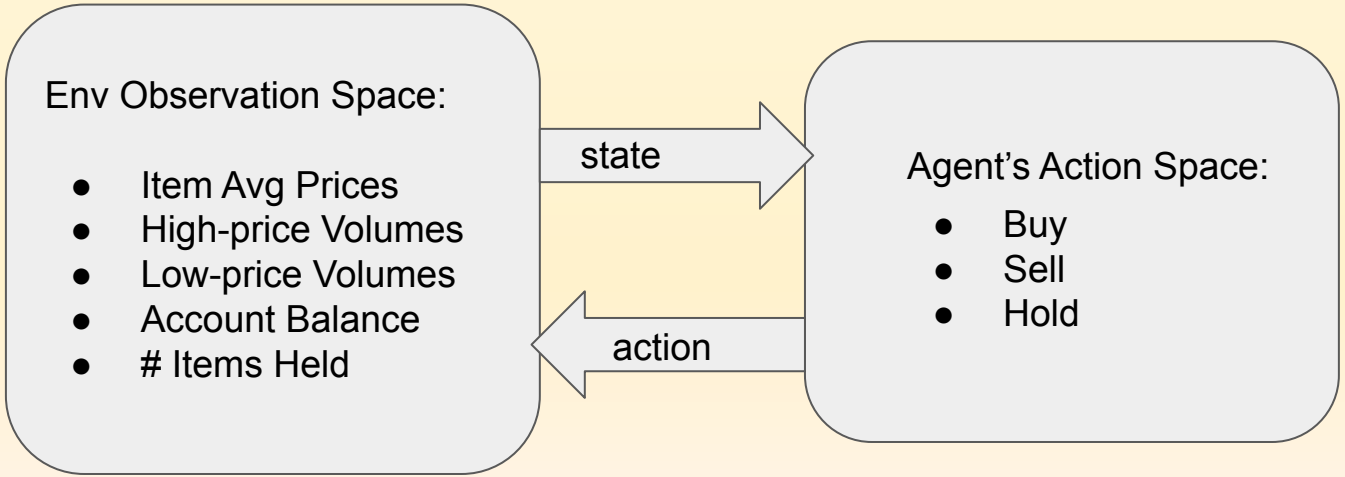
How Does the Training Work?

Gym-Env

- Starting cash : 1 million gold pieces
- Picked random starting states in the first 10,000 data points
- State includes window of last 10 price points for price/volume

Agent

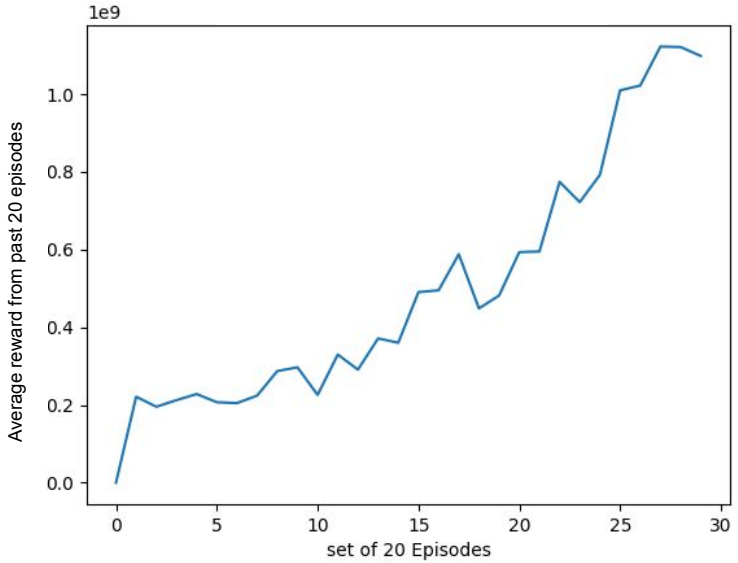
- Neural Network (TD or DQN)
- E-greedy
- Buy - 1000 or using remaining balance
- Sell - 1000 or remaining items
- Hold - do nothing



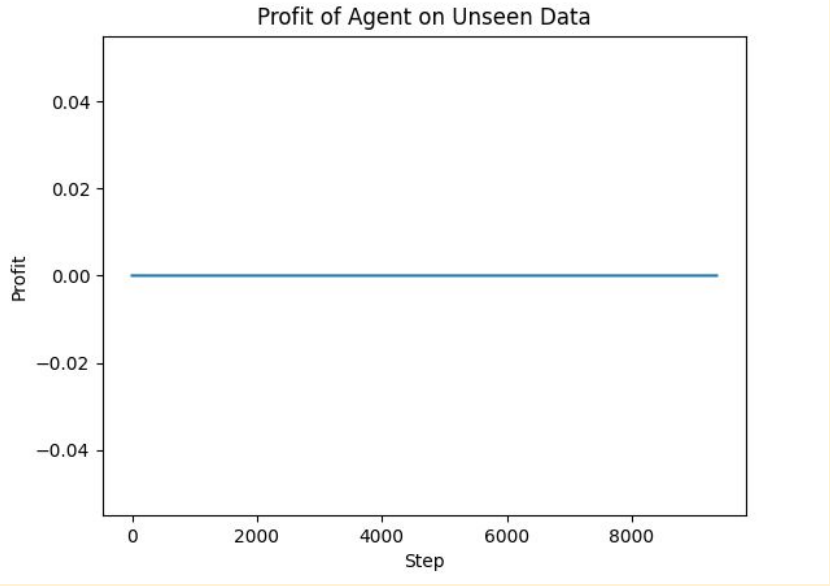
Temporal Difference Agent (TD)

Training

TD Agent learning curve over 600 episodes



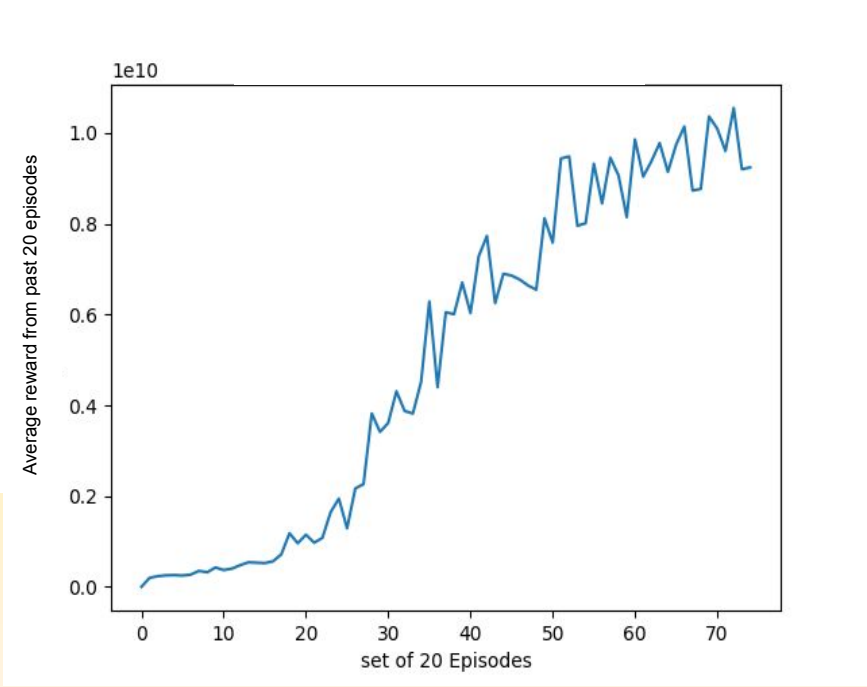
Evaluating



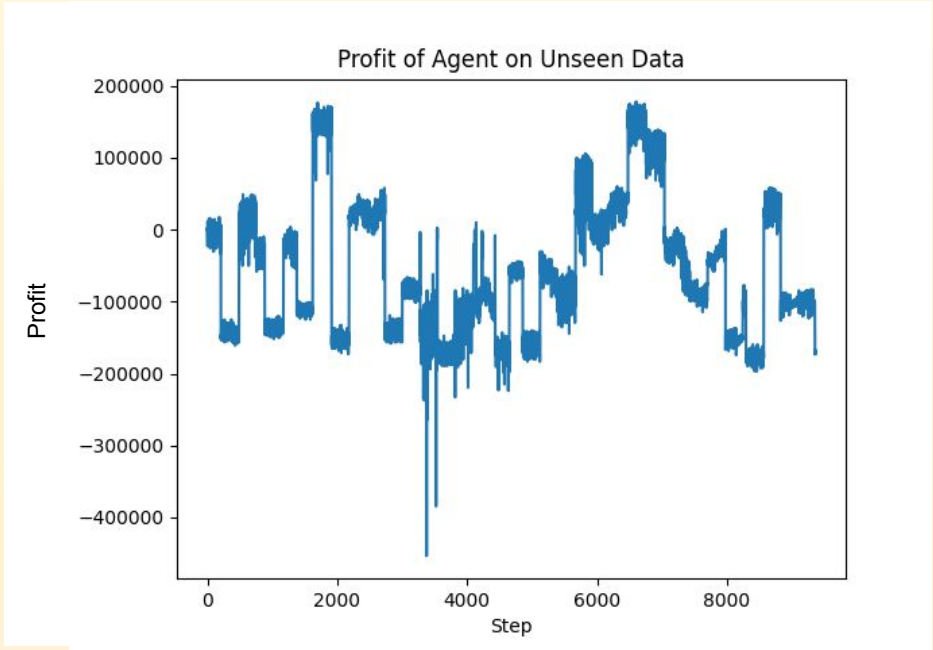
Deep Q-Learning Agent (DQN)

Training

DQN Agent learning curve over 1400 episodes



Evaluating



Resulting DQN Training Episodes

Balance: 113.52
Items held: 5334 (Total sold: 51717)
Net worth: 146923.52 (Max net worth: 1471032.0)
Profit: 146923.52

\+ 15%

Balance: 11747.5
Items held: 8695 (Total sold: 59000)
Net worth: 1846392.5 (Max net worth: 2083380.0)
Profit: 846392.5

\+ 85%

Summary and Conclusions

- Able to get TD and DQN agents learning
- Not obtaining consistent profits, especially on unseen data
- These approaches may not be optimal for this problem space
- Difficult problem to solve