RL Policy Ensemble for Stock Trading

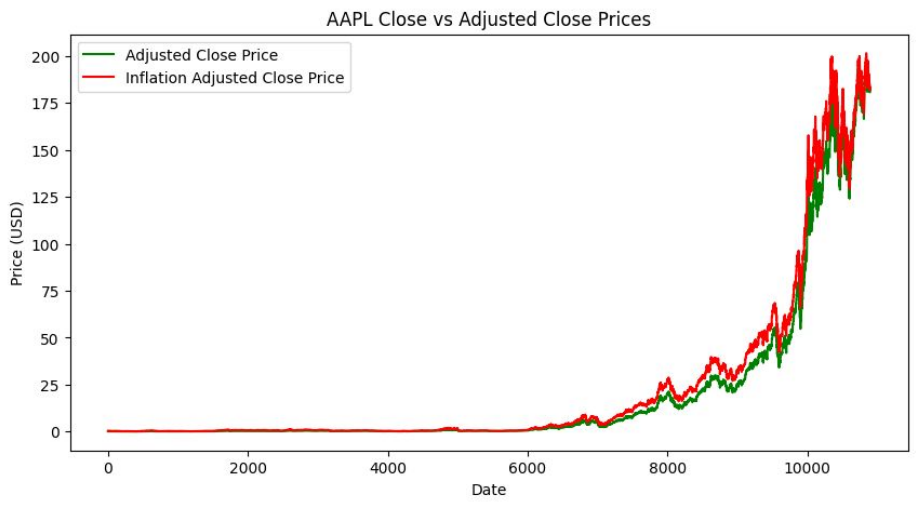
By: Ananya Kulshrestha and David Chaudhari

Background/Motivation

- Predicting behavior in the stock market or other financial markets is generally a very difficult task
  - The stock market is generally unpredictable, stemming largely from the large number of factors that affect the stock market in a daily basis
- Provides a prime opportunity to use more advanced techniques to uncover subtle patterns for optimal trading performance
- Due to the unstable nature of stock markets, using a methodology that is robust to these changes is crucial
  - Using an ensemble method allows us to ensure stability in our predictions.

Data Preprocessing

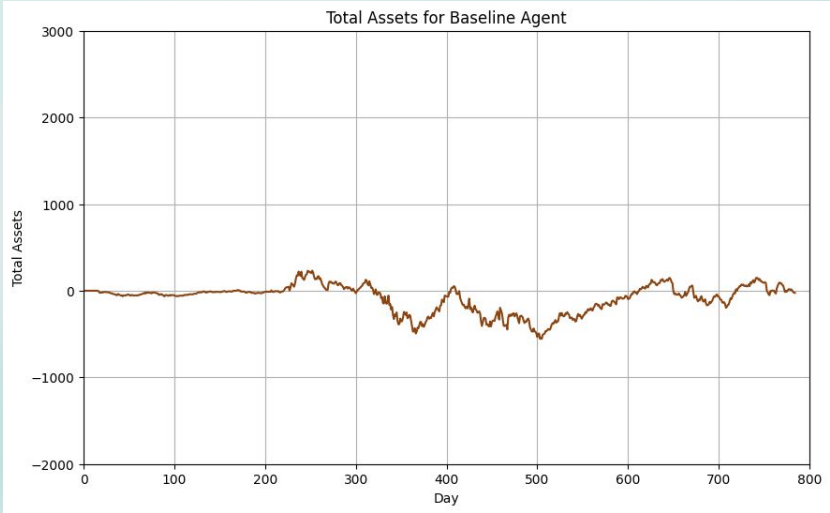
- Used data from Yahoo Finance library
  - Adjusted Closing Prices
    - Accounts for stock splits
- Utilized data from the Federal Reserve of St. Louis in our inflation calculations.
  - CPI Multiplier in terms of 2024 dollars
  - Incorporated the inflation adjusted prices by multiplying the CPI multiplier by the adjusted closing price
  - For project, used newly calculated inflation adjusted price



Sources: https://fred.stlouisfed.org/series/CPIAUCNS, https://towardsdatascience.com/adjusting-prices-for-inflation-in-pandas-daaaa782cd89

Baseline Method

- For this project, we defined a baseline that uses a simpler strategy compared to the other models
  - The baseline checks the closing price of the past 5 days including today
  - If the past 5 days had negative gains, then it buys the stock
  - If the past 5 days were positive gains, then it sells the stock
- Our results are that it is net neutral in gains after 800 episodes

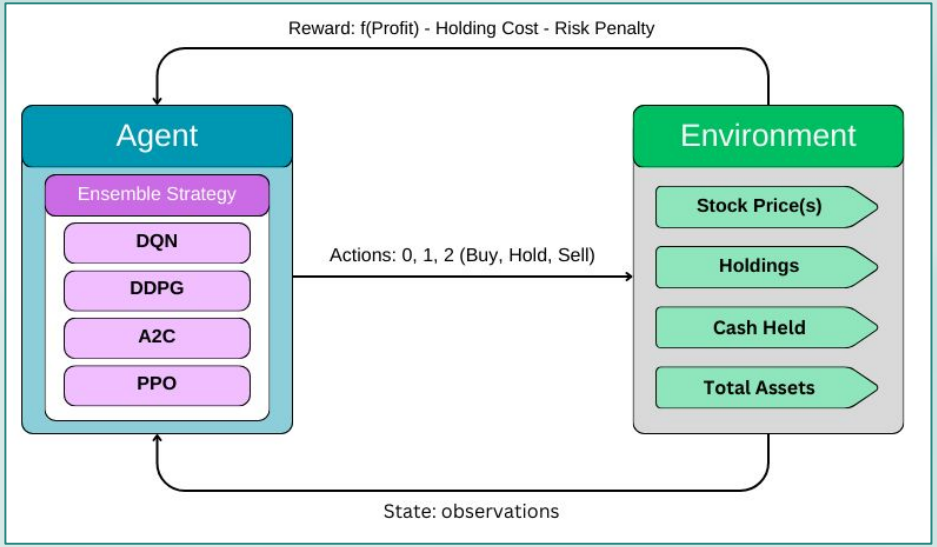


Methodology

- We learned about many agents and their individual strengths / weaknesses
  - Wanted to combine the models
- Method: Ensemble couple agents together
  - Deep Q-Networks (DQN)
  - Deep Deterministic Policy Gradient (DDPG)
  - Advantage Actor-Critic (A2C)
  - Proximal Policy Optimization (PPO)
- Ensemble Techniques:
  - **Max Voting:** Each model votes for an action, and the action with the majority is selected (preference to 0)
  - **Converted Majority Thresholding:** Each agent converts its action into a standardized form (-1, 0, +1 for sell, hold, buy)
    - Ensemble decides based on the sum of actions belonging to certain ranges
- We train each model individually and ensemble statically

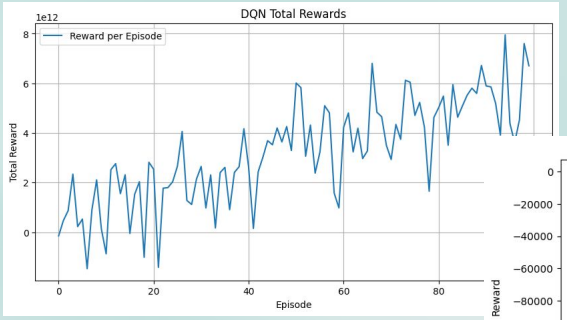
The Setup (AAPL Stock)

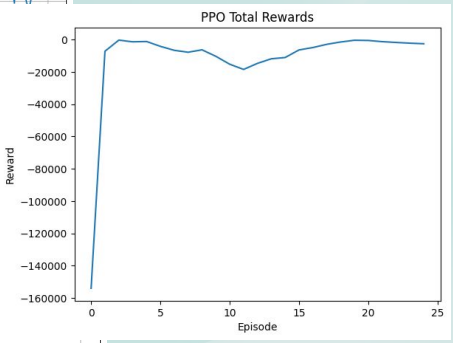
- Observation Space:
  - current price and next n-1 closing prices (window size = n)
  - holdings
- Action Space:
  - Buy (0)
  - Hold (1)
  - Sell (2)
- Reward Design:
  - f(profit)
  - holding cost = holding pct * holdings
  - risk penalty (based on total asset vs initial total asset)
- There can be negative cash and holdings

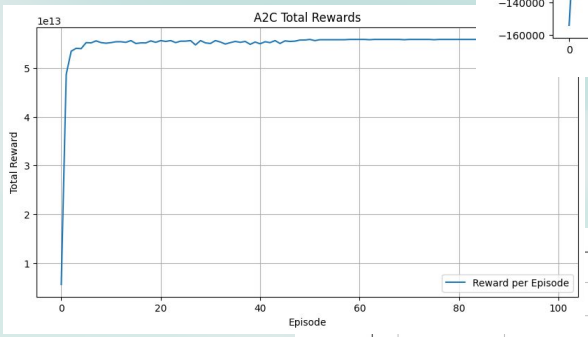


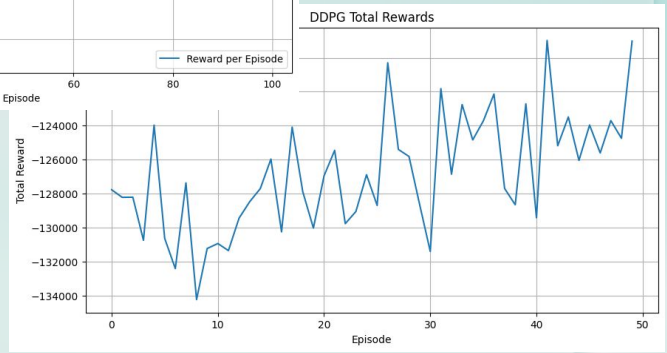
Experiments

- We ran experiments with different reward designs and ensemble strategies
  - Most hyperparameters were slightly adjusted for better performance and were not the main focus of the project
- We trained each agent (DQN, DDPG, A2C, and PPO) on data from January 1, 2010 to January 1, 2020
  - Models were generally converging
- We tested each model on data from January 1, 2021 to March 1, 2024
  - Some models performed well, while others did NOT



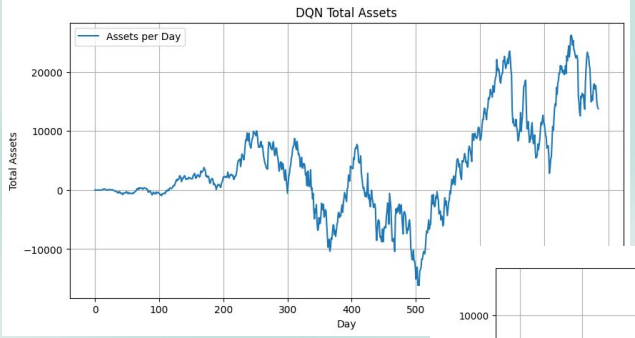


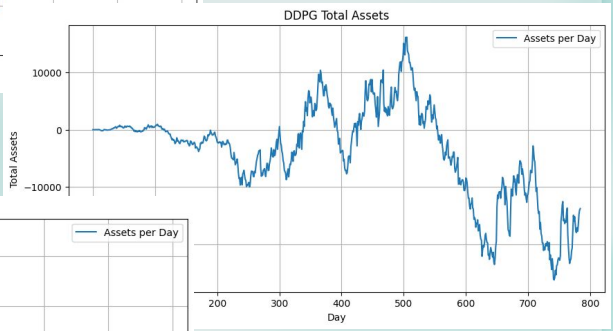


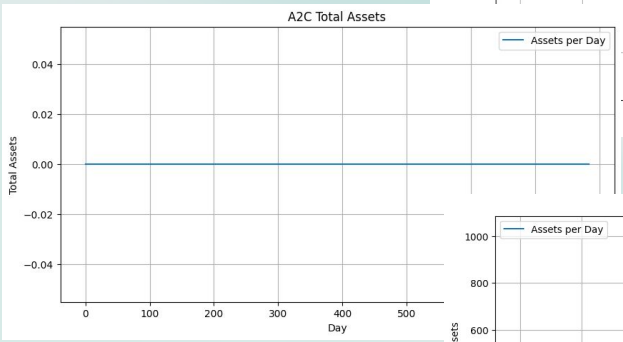


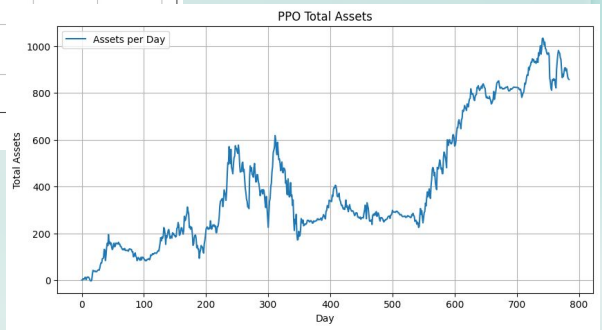
Initial Results

- Our reward consisted of couple parts:
  - [+] PNL: For sell it was the current price - immediate future price and other way for buy
  - [-] Holding Cost: 0.0001 * initial price of stock * |# of holdings|
  - [-] Risk Penalty: Total Asset / 1e-6
- Reward was scaled too high
  - Agents not learning properly even though assets are good
  - Too extreme



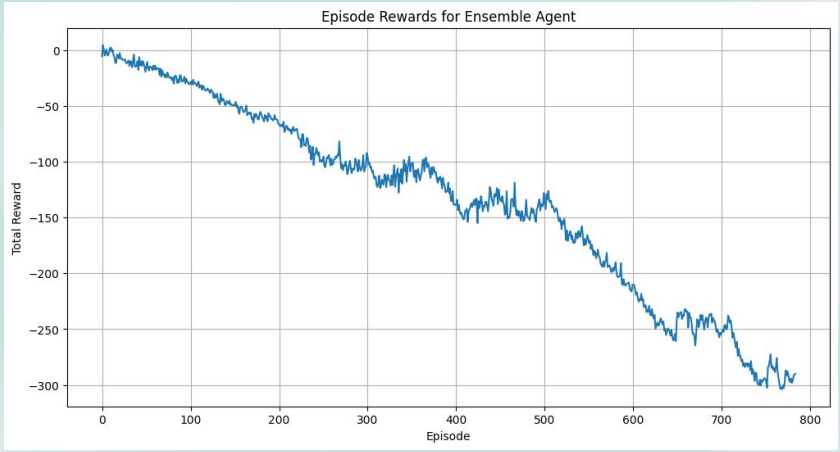


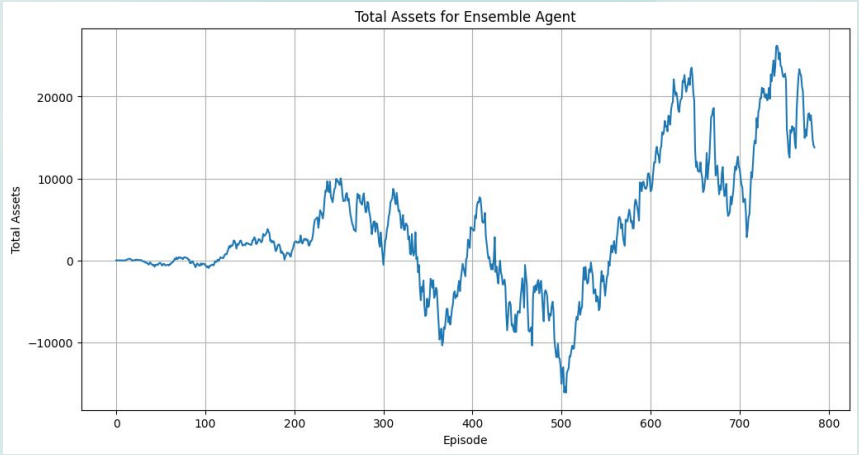




More Results

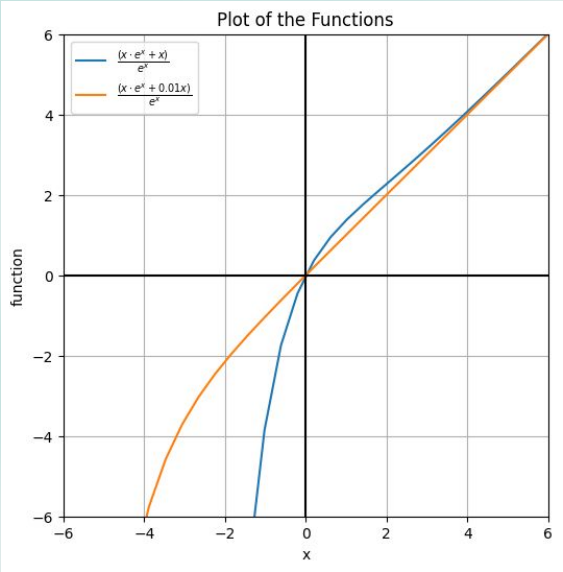
- This time accounted for inflation
  - [+] PNL: For sell it was the current price - immediate future price and other way for buy
  - [-] Holding Cost: 0.002 * initial price of stock * |# of holdings|
- Each model performed weirdly
  - There was no accounting for the volatility
- Ensemble strategy (Max Voting) is too much into buying and very large losses at some points like -$10000
- Even though net profit is good, not good result





Change in Reward Design

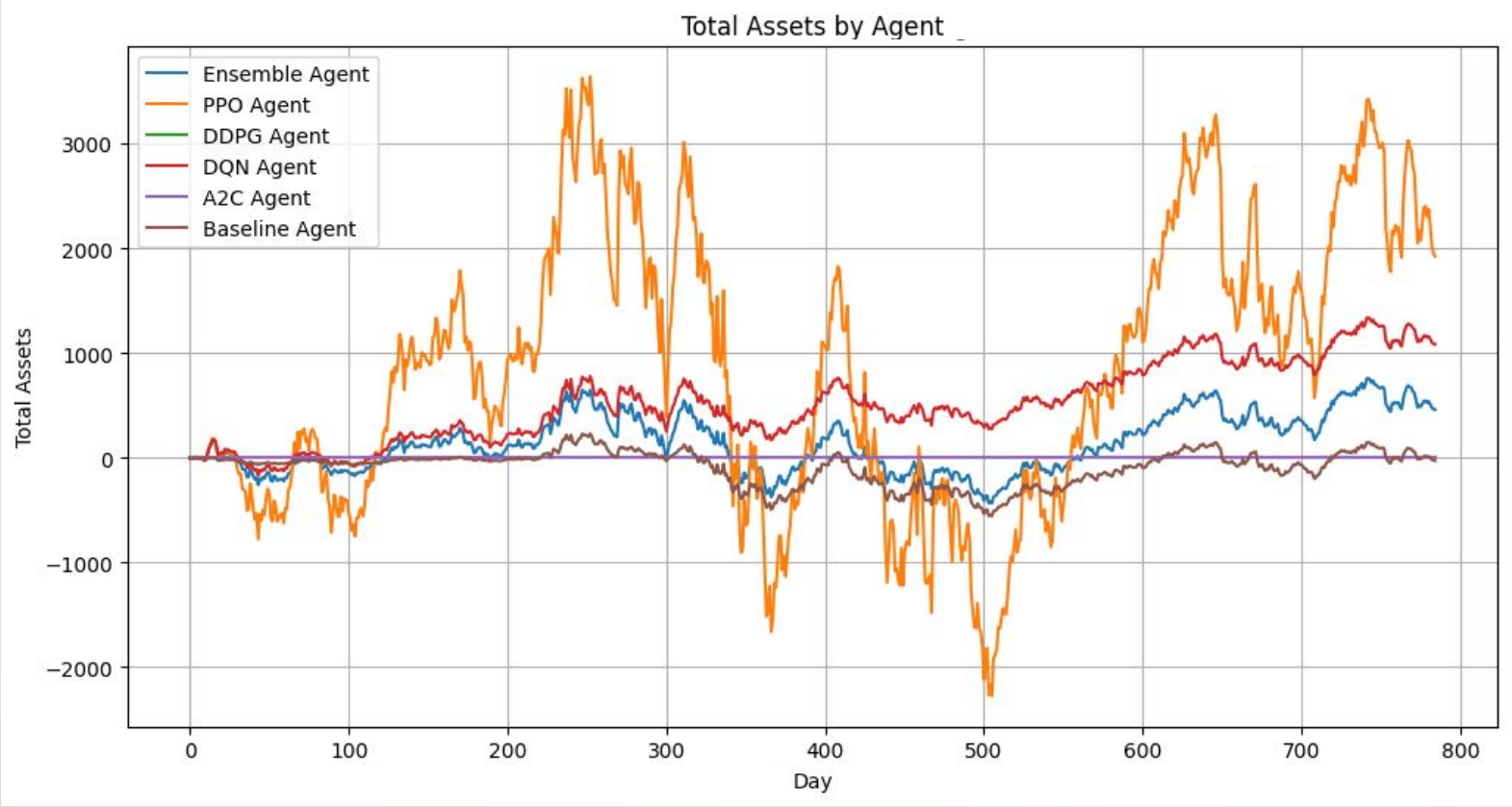
- Want to penalize losses
  - $f(x) = \frac{(x \cdot e^x + 0.01x)}{e^x}$
  - Used f(profit) for positive term
- Holding cost
  - holding pct * current price * |# of holdings|
  - Changed around holding pct
- Risk Penalty
  - Took out concept of initial price (only current price)
  - Drawdown concept
    - Maintain a peak total asset
    - drawdown = (peak total asset-total asset)/peak asset
      - if peak asset < 0, drawdown = 0
  - risk penalty = drawdown * risk pct
  - Played little with risk pct



Final Results

- Ensemble Strategy
  - Max Voting was too biased towards buying when ties occurred, leading to significant losses
  - Changed to thresholding
- Converted Majority Thresholding:
  - Step 1: Get action from each agent
  - Step 2: Convert actions to -1, 0, 1 (Sell, Hold, Buy)
  - Step 3: Sum the converted action
  - Step 4: Threshold
    - Sum > 1 : Action = 0 (Buy)
    - Sum < -1 : Action = 2 (Sell)
    - Else : Action = 1 (Hold)
- Strategy to encourage little more holding

Final Results



Conclusions

- Stock markets are extremely volatile
  - In our results there were some episodes where the baseline does better than some of our models.
- A model is only as good as it's environment + Reward design
  - A lot of time went into creating the right reward function and fixing the environment for buying and selling a stock
  - Changing the observation space with more information can lead to better results
- Expanded on our knowledge of models such as PPO, A2C, and DQN
  - We applied DDPG, which is one of the models that we didn't get to implement in the homeworks
- In our paper we will discuss in more detail the hyperparameter adjustments that we made that gave us the results we got in our experiments