QUANTITATIVE TRADING WITH DEEP REINFORCEMENT LEARNING

TEAM: FARHANUR RAHIM ANSARI AND SHUBHI SAXENA

GOAL

Implement and use a deep reinforcement learning technique to perform portfolio optimization and stock trading to maximize returns and profit.

PROBLEM STATEMENT

- Complex and huge : The financial market is a complex system influenced by several factors like news, economy etc.
- Partial Observable Environment : Difficult to comprehend the full state at any time
- End Goal : Maximize returns based on estimates of potential return and risk.

This makes portfolio optimization and stock trading an effective application of Reinforcement Learning.

TECHNICAL INDICATORS

- Sharpe Ratio
  - Measures the performance of an investment compared to a risk-free asset
  - Represents the additional amount of return that an investor receives per unit of increase in risk
  - Sharpe Ratio = (Return on Investment – Risk-free return) / (standard deviation of investment)
- Dow Jones Industrial Average (DJIA)
  - Is a stock market index gauging the performance of the industrial sector
  - Include 30 companies and averaged their values by following a specific formula

WHY DEEP REINFORCEMENT LEARNING (DRL)?

- Modern Portfolio Theory (MPT) not enough : calculated only based on stock returns and highly sensitive to outliers.
- Markov Decision Process : stock trading is a continuous process, can be modeled as MDP.
- Optimization Problem : DRL solves maximizing the expected total reward from future actions.
- Multidimensional & large dataset : DRL approximates the Q value with a neural network

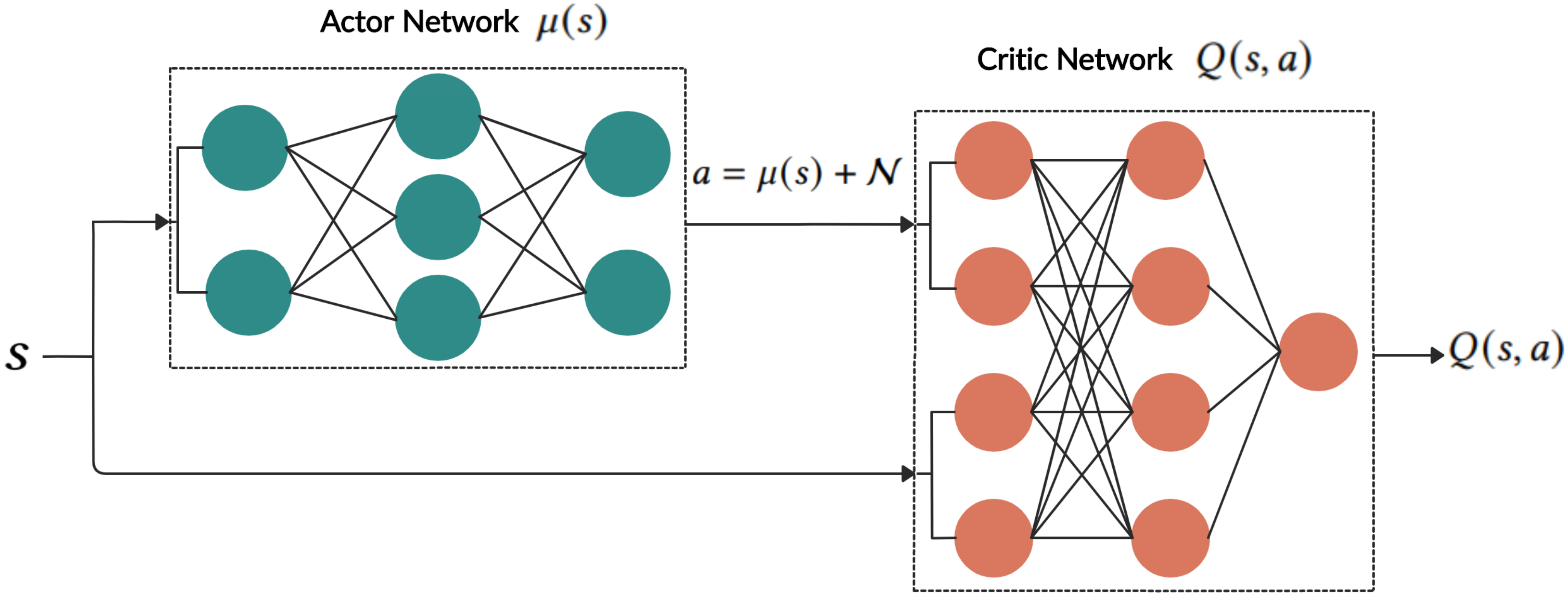
TRADING ENVIRONMENT

- States
  - account balance – money in the trading account
  - Date, Open, High, Low, Close, Adjusted Close, Volume
  - relevant stock technical data
- Actions
  - Buy - Perform and record buy transactions
  - Sell - Perform and record sell transactions
- Rewards
  - Total asset gain/loss by the end of each day
- Episode
  - End of episode is defined when timestamp reaches last day in feature data
  - Reset environment after each episode

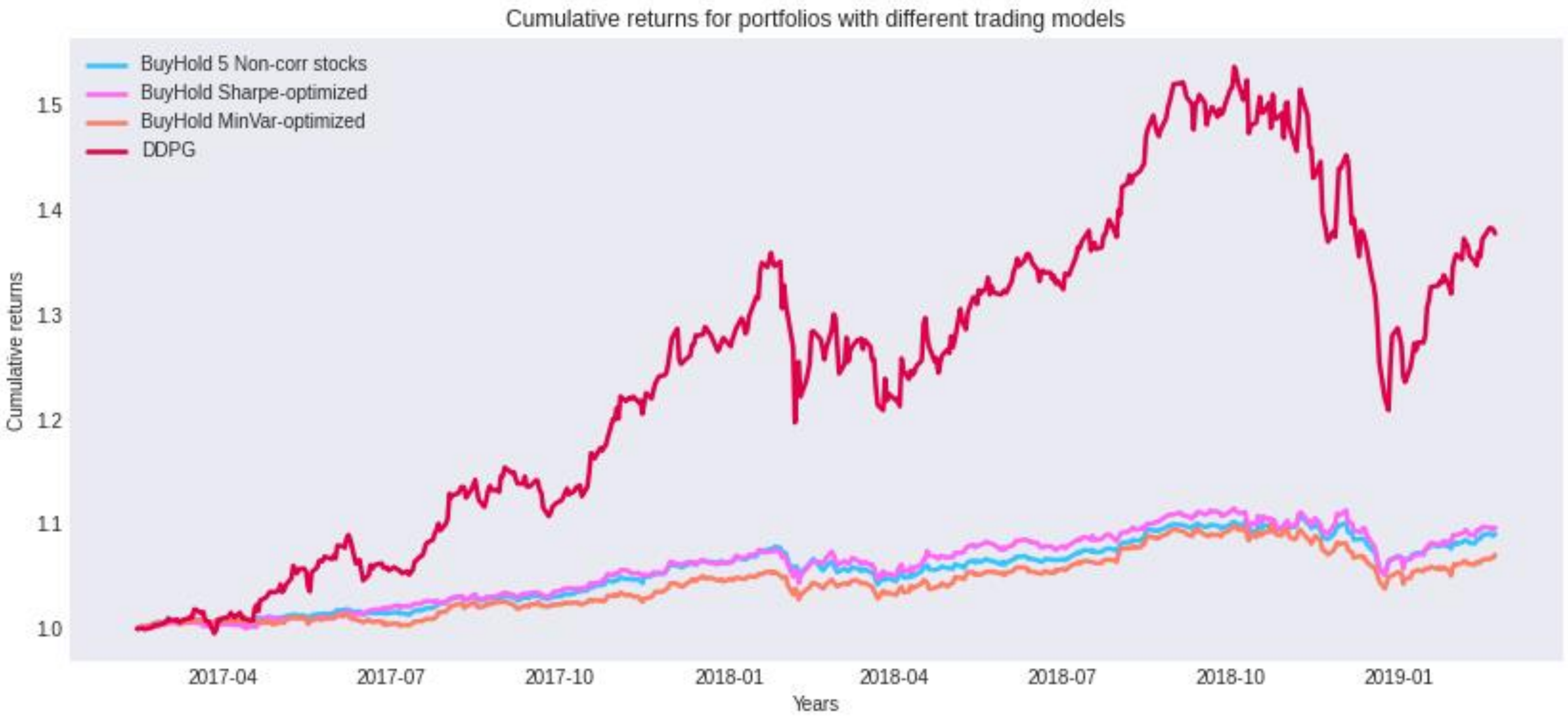
DEEP DETERMINISTIC POLICY GRADIENT (DDPG)

- Actor-critic based algorithm
- Actor – Proposes an action given a state, Critic – predicts if action is good/bad given a state and an action
- Combines both **Q-learning and policy gradien**t frameworks
- Uses neural networks as function approximators
- Learns directly from the observations through policy gradient

DDPG ARCHITECTURE



RESULT – DDPG VS MPT



FUTURE WORK

- Comparing DDPG performance against the benchmark DJIA model
- Implementing a supervised deep learning approach using Recurrent Nural Network (RNN) and comparing it with DDPG result
- Trying our approach on different portfolios

THANK YOU!