Assignment 1 Report

Sri Vyshnavi 23b1264

# 1. Normalization and Stationarity Methods

In this assignment, I collected 5 years of daily OHLCV data for two assets: Apple (AAPL) and Microsoft (MSFT). I used the yfinance API to fetch the data.

As of now, I have not applied normalization or differencing. My reason for this is to first understand the raw structure of the data and the impact of the technical indicators I applied. Raw data also helps visualize the real market trends more clearly at this early stage.

However, I am aware that time-series models usually benefit from normalization. If I were to normalize, I would use:

* + **Min-Max Normalization:** This scales all values to a fixed range (usually 0 to 1). It is simple and works well when there are no extreme outliers.
  + **Z-score Normalization:** This method transforms the data to have mean 0 and standard deviation 1. It is useful when the data has different scales or units.

For making the time series stationary, I plan to try differencing (subtracting previous values) in the next phase. This is useful because many models assume constant statistical properties over time.

# 2. Technical Indicator Used: RSI

For this part of the assignment, I implemented the Relative Strength Index (RSI) for Apple stock. RSI is a momentum indicator that helps in identifying overbought and oversold conditions in the market.

I used a 14-day RSI, which is a standard value. It is calculated by comparing average gains and losses over the past 14 days. When RSI goes above 70, the stock is said to be overbought, and when it falls below 30, it is considered oversold.

**Why I chose RSI:**

* + It is easy to calculate and interpret.
  + It helps identify potential reversal points in the market.
  + It can be a good signal input for a trading agent to learn profitable actions. In future steps, I plan to add more indicators like:
  + **MACD:** It shows momentum and trend direction using EMAs.
  + **SMA/EMA:** Moving averages help in smoothing the price data.
  + **Bollinger Bands:** They show volatility and can act as dynamic support/resistance.

# 3. Data Splitting and Visualization

After calculating the RSI, I split the data into:

* + 70% for training
  + 15% for validation
  + 15% for testing

This is a time-series-aware split, meaning I did not shuffle the data. Shuffling breaks temporal patterns, so it’s avoided in financial data.

For visualization, I plotted the Close Price of Apple along with its RSI on the same plot. I noticed that RSI tends to drop after sharp price rises and vice versa, which confirms its usefulness.

# 4. Challenges and Learnings

Some of the challenges I faced included:

* + Understanding how RSI is calculated manually using pandas.
  + Making sure the rolling averages worked properly even with NaNs at the start.
  + Scaling RSI and price for better visualization (they were on different value ranges).

This task helped me understand the importance of preprocessing and basic technical analysis. I also learned the value of choosing indicators wisely for a reinforcement learning model.

# 5. Next Steps

In the next assignment or further steps of this project, I will:

* + Add more technical indicators.
  + Try out normalization and differencing methods.
  + Create more descriptive plots like raw vs normalized data.
  + Prepare the dataset for use in DRL models.

# Conclusion

This assignment was useful in helping me build a clean data pipeline and begin understanding market signals. I am excited to continue working on this project and apply what I learned here to reinforcement learning-based trading systems.