Financial Simulation Project Description

The goal of this project is to create a simulation of a financial market in Python. In this simulation, non-human controlled AI agents represent stock traders, trading various securities/stocks. This simulation of a financial market is meant to mimic real-life circumstances as close as possible. Therefore, there will be two broad types of AI stock traders:

- *Institutional Traders* – Represent hedge funds that has access to large amounts of capital that invest in stock market
- *Retail Traders* – Represent the "everyday" people that trade in the stock market with limited capital

Within institutional traders & retail traders, we can further differentiate them by differing the strategies they can trade with. For example, one group of retail traders may utilize momentum trading, in which they trade with the trends, and another group may employ moving average-based strategies. Additionally, retail traders could have certain behaviors such as herd behavior or overconfidence when trading.

On top of the trading strategy they would use, machine learning algorithm(s) would be integrated with these agents to ensure a more realistic simulation. One or both of the following machine learning algorithms can be used for this project:

- *Reinforcement Learning* – Training an AI agent via trial and error using feedback in the form of rewards and punishments
- *Neuroevolution* – Using evolutionary algorithms to develop artificial neural networks to evolve trading strategies over time, like natural selection in biological evolution

If the latter approach is chosen, where both algorithms are being used, the following could be the design:

1. First use *neuroevolutionary* to find the optimal neural network architecture that can process market data

2. Once we have a promising neural network architecture, use *reinforcement learning* to optimize the trading strategy

The technical architecture of this project will be in Python and use Object Oriented Programming (OOP). The classes and methods will be most likely to be organized as the following:

1. Trader Class – Represents AI agent. Contain methods for creating limit and market orders

2. AI Agent Class – Inherited from trader class. Contain methods based on AI algorithms

3. Security Class – Represents the stock. Contains method for order book and executing trades

4. Market Class – Responsible for opening and closing market

This project also may contain an optional UI for viewing graphs of all the securities and inspecting traders of their portfolio value and % change.

Guiding Questions

The goal of this project is to answer the following research questions:

1. How do the interactions of different traders and strategies contribute to broader market trends and phenomena such as liquidity, volatility, and uptrends/downtrends?

2. How do behavioral factors (e.g., herd behavior, overconfidence) incorporated into retail trading influence overall market stability and volatility?

3. How effective is the combination of neuroevolutionary and reinforcement learning in developing and optimizing trading strategies?

4. How do different trading strategies manage risk and affect trading performance when being in competition with other strategies?

Other AI Agent Simulation Examples

Similar AI agent simulations has been used in other domains, which can be seen below:

- [AI Agents Playing Monopoly Against Each Other](#)

- [AI Agents Simulating Civilization in Minecraft](#)