Summary

Papers

High-frequency trading in a limit order book

True price is given by the market mid price.

Agent fills a bid and ask order.

The reservation bid price is the price that would make the agent indifferent between his current portfolio and the portfolio he would have if he filled the bid order.

The bid price p^b is this bid price and $\delta^b = s - p^b$ is the bid spread and determines the priority of the execution.

Mbt-gym: Reinforcement learning for model-based limit order book trading

Three broadly approach,

- market replay approach (historical data is replayed and the learning agent interacts with it)
- model-based approach (this paper)
- agent based market simulator (learning agent interacts together)

Training

PPO (Proximal Policy Optimization) is used to train the agent. It is a model-free algorithm.

Deep Reinforcement Learning for Market Making Under a Hawkes Process-Based Limit Order Book Model

However, due to the naive assumptions they are predicated upon, the LOB models underlying most con-temporary MM approaches remain inconsistent with respect to direction, timing, and volume, leading to phantom gains under backtesting and preposterous events [16], such as price decreases after a large buy market order. For example, in the original AS model [1], price movements are assumed to be completely independent of the arrivals of market orders and the LOB dynamics, while the subsequent approaches only partly address such inconsistencies. To ameliorate this, a novel weakly-consistent pure-jump market model that ensures that the price dynamics are consistent with the LOB dynamics with respect to direction and timing is proposed in [16]. Nevertheless, it still assumes constant order arrival intensities, meaning that any (empirically found) effects of self- or mutual-excitation and inhibition between various types of LOB order arrivals remain unaccounted for

Theory

Glosten-Milgrom model

We trade for liquidity reasons, we do not have informations. Exclusively exegenous trading decisions.

The key insight of the Glosten-Milgrom model is that the market maker protects themselves against potential losses to informed traders by setting a spread between the bid and ask prices. This spread compensates the market maker for the risk of trading with someone who has better information. As a result, the spread can be seen as a measure of the information asymmetry in the market.

In this instance we are uninformed traders.

https://github.com/jkillingsworth/mm-glosten-milgrom Demonstrates the difference between the true price and the market price.

Bandit Problem

A bandit problem consists of a game played over T rounds. In each round t, the player must choose an action a_t (or arm) from a predefined set of alternatives $\mathcal{A}$. After each choice, the player is given a reward $r_t(a_t)$ and play continues to the next round. The player's aim is to accrue the largest total reward at the end of the T rounds. Crucially, the rewards for the unplayed actions are not revealed each round, so the player is forced to balance exploring unplayed actions (in case they have larger rewards) and repeatedly playing those actions which have given high rewards in the past.

When analysing algorithms that play bandit games, a central quantity is the regret of a particular sequence of actions $a_{1:T} := a_1, \ldots, a_T$. This is the difference between the total reward $R(a_{1:T})$ the player received and the best total reward that would have been received if a single arm were played repeatedly. That is, the regret of the sequence $a_{1:T}$ is

$$\texttt{Regret}(a_{1:T}) = \sum_{t=1}^{T} r_t(a_t) - \max_{a \in \mathcal{A}} \sum_{t=1}^{T} r_t(a).$$

Typical analyses of bandit problems derive sublinear (i.e., $o(T)$) bounds on the worst-case regret for particular algorithms and under various assumptions about the power of the player's adversary when choosing the sequence of reward functions $r_1, \ldots, r_T$. The structure of the action space $\mathcal{A}$ (e.g., finite, convex, etc.) and the set of reward functions the adversary can choose from also play an important role in the type of results that are obtained.

Implementation

We will only model one player as the stock price already reflects other participants.

The market making problem can be seen as reinforcement learning problem. The agent has to learn the optimal policy to quote bid and ask prices in order to maximize its reward. As gradient descent, looking for the local maximum will not yield the best result. The agents actions falls between holding the stock without proving liquidity and provoding liquidity at the risk of his profit, which means running after the local maxima, i.e. buying and selling indendently of the stock path.

Nevertheless we assume the trader can only make exegenous trading decisions. To strengthen the model we could include the whole book and make the agent aware of the order flow. For now it mostly relies on the midprice which we assume to be the true price [Marco Sasha].

$$\texttt{Mid-price} = \frac{\texttt{Best bid} + \texttt{Best ask}}{2}$$

Following the Glosten-Milgrom model, the agent will quote a bid price p^b and ask price p^a. The bid price is the price at which the agent is willing to buy the stock and the ask price is the price at which the agent is willing to sell the stock.

We also define the reservation bid, resp. ask price as the price that would make the agent indifferent between his current portfolio and the portfolio he would have if he filled the bid resp. ask order. The reservation bid price is defined by, $$

$$

For now, the state space is composed of the inventory, the cash and the midprice. The action space is the displacement from the midprice at which the agent quotes their ask and bid prices. The reward is the change in the value of the position adjusted by the risk.

The agent will learn the optimal policy to quote bid and ask prices in order to maximize its reward. It can be represented as holding a stock providing dividends.

To some extent everybody gains by being sometimes a market maker. If I'm convinced the stock will go up but it is right now falling down, I can fill all the order to gain money and then hold it until it goes up. This is again a complex problem as it is a balance between receiving these rewards and suffering future losses and buying the stock at the lowest price to maximize the future gains without collecting rewards. Under cash constraint the agent has to learn something not trivial.

Interest

Performance with ulta volatile stocks and its performance with financial crash. (Adversial training)

We are training agents on a model which is a strong limitation of the economy. It will be interesting to see if the agents can transcend this framework where the model assumptions fails, i.e. model the stock with GBM and see if the agent trained on this approximative model perform where the stock is not a GBM.

Questions

What represent a step in our model ? We we will need to backtest our agents, how should we model this? Since the terminal time is always one and the markets are open from 9:30 to 16:00. It means there is 6:30 hours = 390 minutes hence we should take 390 steps so that every minutes is a step ?

What is the end value of his portfolio? (cash + end_inventory * value of the stock + cumsum of the reward)

The agent will surely outperform with OU process. I feel like he kinda learns how to leverage the volatility of the stock.

To Do

Read paper:

- Deep Reinforcement Learning for Market Making Under a Hawkes Process-Based Limit Order Book Model

See how they implement hawkes process in their depp RL model. (Question 1)

- Quantifying endogeneity of cryptocurrency markets

Understand the strength of Hawkes process. (Question 4 -> POMDP or semi-Markov decision processes)

- Adversial training

See how to implement adversial training in the model (Question 2)

- (Still have not find the paper)

(If there is time or if we fail to implement the other models we can look at offline RL for (question 3))

Implementation:

- Could be great to increase the state to the whole book so it has full endogeneity.
- To back test we should have some real data to compare the different agents. (Stable data or in period of crisis…)
- Metric to evaluate

AGENTS:

- 0 actions
- Random actions
- Avellaneda and Stoikok
- Cartea, Jaimungal, and Ricci

4/5 agents we will build

- PPO
- PPO on hawkes…

METRICS:

- Mean spread
- Mean & Sdev PNL (cumulative reward at the end of the day)
- Mean & Sdev Inventory
- Mean & Sdev

? Sharpe ratio ? (does it make sense)