Cornell Data Science: Algorithmic Trading Team

Market Making:

In order to have an efficient market, an individual has to be able to buy and sell their security at any given time (when the market is open). Ordinarily it is hard to be able to ensure that there is enough liquidity in the marketplace such that buying and selling can happen seamlessly and nearly instantaneously, but when there are market makers present this is possible. Market Makers are participants in the marketplace that provide liquidity by offering both a buy and sell quote (bid-ask) on a given security on a continuous basis. The goal of the market maker is to profit as much money as possible (this is done by recovering the money made off the bid-ask spread by buying and selling the security) with as little risk as possible (holding minimal inventory at all times). My goal is to automate this process and use a machine learning approach to formulate bid-ask quotes with the objectives of an ideal market maker in mind. Below we will discuss the methods we used to create an automated market maker.

A Limit Order Book Market Making Strategy :

Imagine that you want to buy Facebook stock right now. You open up your Robinhood application and place an order to buy one share. Soon after, you receive confirmation of the transaction and find a new addition to your portfolio. Could the whole process be so simple? Have you ever wondered what’s really going on under the hood?

Behind the scenes, Robinhood’s execution algorithm interacts with a limit order book of FB stock on an electronic stock exchange to acquire your share. The limit order book, or LOB, is a structure that contains all of the price levels at which market participants intend to buy or sell FB stock on an exchange. It contains both the quantity and the side (buy or sell) of orders at every price level. To ensure that LOBs are densely populated, or “liquid,” an entity called a market maker exists. The market maker is legally obligated to both buy and sell shares of FB regardless of the stock’s daily trajectory. In order to fulfill this obligation he will submit prices to a LOB. Throughout a day of trading, the market maker accumulates short or long positions in FB stock and faces the risk of losing money on these holdings. The keen market maker, however, mitigates this inventory risk while profiting from buying stock and selling it at higher levels. With these crucial tradeoffs in mind, I will now discuss a market making strategy along with its simulated performance.

I will approach the problem of market making with the goal of maximizing an exponential utility function listed below (1).

(1) https://lh3.googleusercontent.com/LRd7ESbtGCQqxXVI21ubn9aZ2cMjkTSINCEt7WUXxmtKJVfyEgttEE-SExwVWYqeL6-wBl2M7NRahqozLZE_T28huDCv-Y8gYiBuxsCm6Ej3ty5LMYSFhxMePHVV2apzfsTaMzz_vuE

Given the current value of a stock (*s*), our current inventory of it (*x*), and a quantity representing our “desire to take risks” (), this function outputs a measure of our expected utility.

Next we must use this value function to determine optimal prices at which we will offer to buy and sell stock. For this we will calculate our reservation bid and ask prices, and respectively. These values denote the prices at which we are indifferent to buying a share and doing nothing, or selling a share and doing nothing. Moreover, we would never want to buy for a price above or sell for below .

is given by the value that solves the following equation (2):

https://lh4.googleusercontent.com/Dw4i3ijauw-1M6Aaw8c4blCyMYO8pOPGW0gjuMyXoQTuHbS-3dzhWnfvVoR-OH4CpTQr67sySgIow9qkLb57A_E4aJ-XVOTKzyMRDHGnmTZtnLYcvivSJTBhFENMPiOmH_JncNkMfXQ

(2)

A dynamic programming approach to approximate (1) and find our reservation values is outlined in *Avellaneda and Stoikov.* We will use their results:

(4) https://lh3.googleusercontent.com/0ZZu-unDRMqH8Iqyf1KP9rqrxZA1JaYpElCeMzKWpE9AXMYx7T-ZBwElEzyMsU0rzU4diCKDAGlg7ycuHKY0nemz_FQLSrWh8_zZmIN0ORQq5NDvcBnt5ogeTr6N80xoVrPY04_PmP8

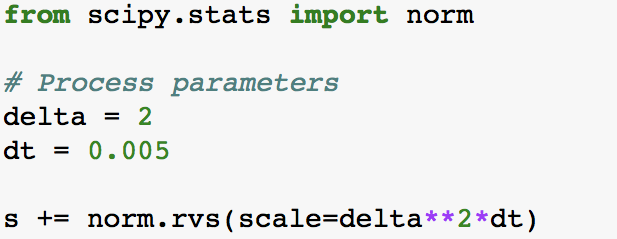
(5) https://lh6.googleusercontent.com/w0aCGQj0LbS3gb6dV9TxdFetHI71646_2nGZWt7OhFj8zcLncOpjahJyIiEF1wSGvFrCGkQqq7yRjuOII92r5Jiv2wQkpjV0D8rwkT6xlDKmawM7oiV1Mp4G69gt8kOuPb8GQn0hSdA

with and

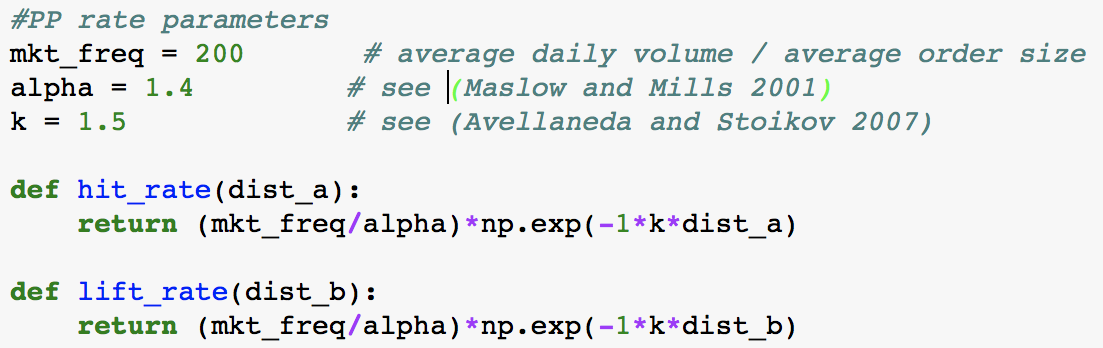
We will quote our bid and ask prices symmetrically around the arithmetic average of and which is approximated by (4). Then, we will submit orders to buy and sell shares at levels equidistant from and determined by (5).

After Implementing our inventory (*q*) based pricing policy given by (4) and (5), we will

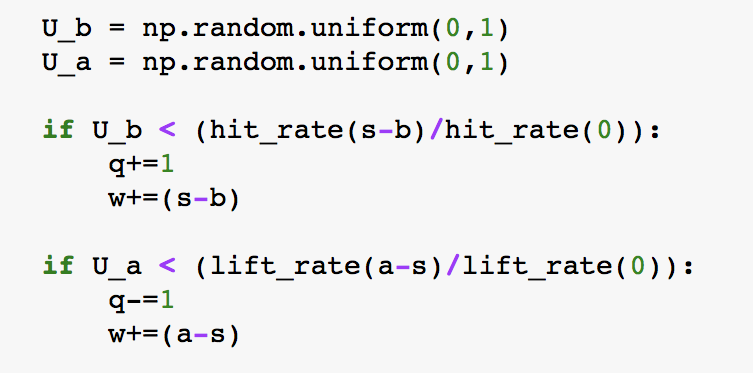
examine the strategy’s simulated performance over a period [0,T]. The simulation will procced in discrete increments *dt* and will terminate at time *T*. First, we will model the stock price as standard Brownian motion with initial value s = 100. The following code is used to update price *s* at each time increment *dt* while t<T:



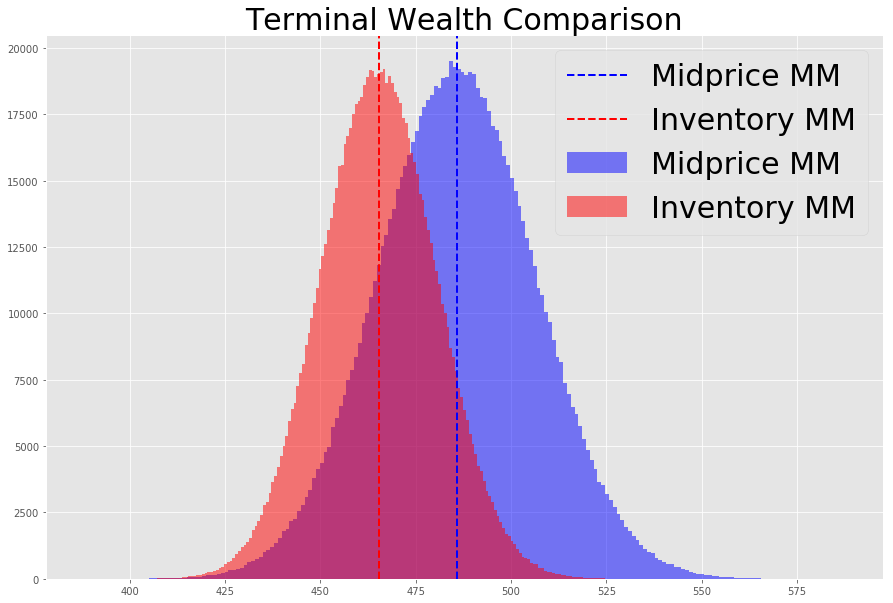
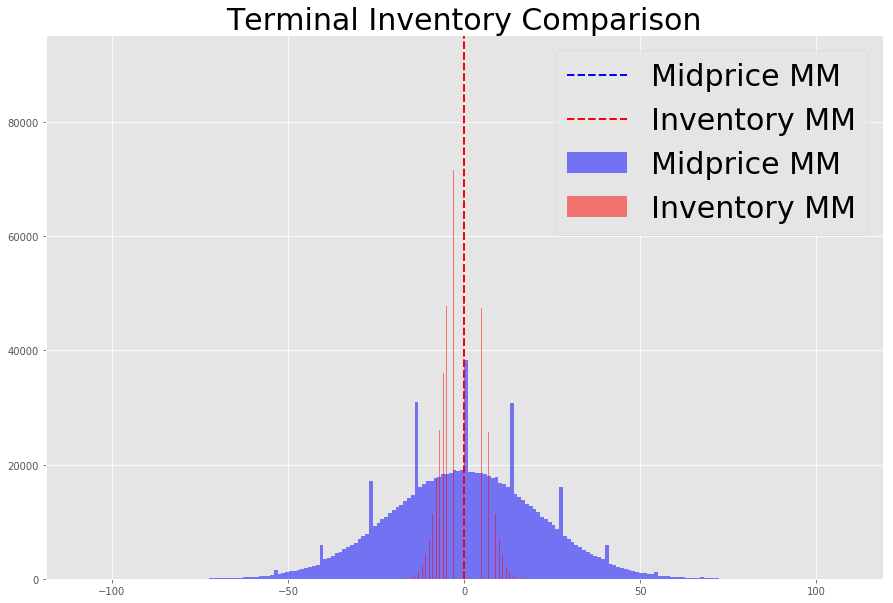
Next we will simulate the frequencies at which our quotes are “hit” or “lifted.” This is the rate at which market participants will sell to or buy from us respectively. We will assume that these hit and lift rates follow Poisson Processes. The rate parameter of each process will be directly proportional to how good our price is. For example, if we offer to buy a stock right below it’s true value *s* at time *t*, we are likely to have more interest than if we offered to buy for much less than *s*. Code for the exact functional form of these rates is given below:



At each time increment *dt*, we will sell a share with true value *s* for a price with probability . Independently, we will buy a share for with probability . Given that these rate functions are both non-increasing with the size of our spread, = and = . In the event of a buy order arriving, we decrease our inventory *q* by one and increase our profit *w* by . If a sell order arrives, we increase inventory by one and increase profit by . With these assumptions, we are able to simulate the arrivals of buy and sell orders as follows:



After iterating thousands of T-period simulations we are able to view the performance of our strategy that quotes prices symmetrically around our reservation value. For comparison we also simulate the performance of a simpler strategy that always quotes prices symmetrically around the true value of the stock. This strategy does not seek to mitigate inventory risk in any way. The following figures display the results of final wealth and inventory levels accumulated by each strategy over every simulated period.

In figure (9) on top we see that our inventory strategy is on average less profitable than the benchmark, mid-price strategy. Our inventory strategy however consistently manages inventory risk by ensuring that we never buy or sell to much stock (10). Thus, although it seems to be slightly less profitable then a benchmark market making approach, our inventory-based strategy proves to yield consistent profit with low positional risk.

One area we are looking to experiment with more in the future is by taking a Bayesian approach to the market making problem. One way of thinking about this problem is by separately considering the true underlying price of the security (which is what the bid-ask quote should be centered around) in addition to the value of the spread (which is what our inventory is based off of). Since we don’t know the true value of either the underlying price or the true optimal value for the spread, we can use a probability distribution on both of these to model our uncertainty on both of these values. Ideally, when we place quotes to the marketplace we would then use the expected value of our price distribution in tandem with the expected value from our spread distribution. And then when new information comes to us, we can use Bayes Rule to conditionally update our distributions. For instance, more inventory would cause us to update in a way such that our spread value would decrease.

The key to making the Bayesian Approach would be to find a probability distribution that can approximate the true underlying price and spread to high degree. Additionally, it would help if there would be a way to give slightly more importance to the more recent information as market data is more relevant if it is newer.