

Automated Market Making

Overview: You'll build an adaptive quoting strategy that mimics a market-maker in a volatile, real-time environment. Every timestamp, your code posts:

- bid_price
- ask_price

based on the L2 order book, recent public trades, and your current inventory. Maximize total PnL while keeping inventory within ± 20 units.

Data Provided:

1. orderbook_train.csv (training) - [Link](#)

- timestamped L2 snapshots: bid/ask prices & sizes at multiple levels.
- Columns: timestamp, bid_price_1, bid_size_1, ..., ask_price_2, ask_size_2, etc.

2. public_trades.csv (training) - [Link](#)

- timestamp, price, size, side (buy/sell)

3. orderbook_test.csv

4. public_trades_test.csv

Tick Size: All quotes must be multiples of 0.1. Any quote not on this grid is invalid and will incur hidden penalties.

PLEASE NOTE - PICK ONLY TOP **3000 rows** from all datasets for submission

Submission Files:

1. submission.csv
 - Rows only at timestamps when you post or update a quote.
 - Columns: timestamp, bid_price, ask_price
 - Example: 0, 100.0, 100.5
2. Update strategy function in class AutomatedMarketMaking

Detailed Flow: At each timestamp t :

1. Inventory is known from all fills up to t (inclusive).
2. You call `strategy(...)`, passing current inventory, order book & trades up to t , and `TICK_SIZE`.
3. You output a new quote for t (must satisfy $\text{bid} \leq \text{ask}$, and multiples of tick size).
 - If you don't post a quote at t , your previous quote remains active.
 - Any time your inventory changes or a fill occurs at t , you must send a new quote on that side at t , else that side will deactivate.
4. The simulator applies any fills at $t+1$ (based on the quote you posted at t) to update inventory.

5. Fills are simulated based on public trades

Inventory Risk & Rules: – Keep $|inventory| \leq 20$

- $bid_price < ask_price$
- Quotes active from $t+1$ to next update or end of test
- If your inventory changes at t , update that side in your quote at t , or that side deactivates
- Please note that at time T , all the quotes should be based on time from 0 to $T-1$, any attempts at using future data (coming up for a quote at T using data from $T+1$ to N) will result in submission failures with error "future-peek detected"

Evaluation Criteria:

1. PnL Score: realized PnL, unrealized PnL (mark-to-market), and penalties.
2. $PnL = realized\ PnL + unrealized\ PnL - penalties$, where unrealized PnL is calculated using the last mid from order book
3. Risk Management:
 - Quotes crossing the market ($bid \geq ask$) incur penalties.
 - Inventory beyond the limit of ± 20 units incurs penalties.

Input Format

Orderbook Snapshot (L2)

Timestamp	Bid Price	1Bid Size	1Bid Price	2Bid Size	2Ask Price	1Ask Size	1Ask Price	2Ask Size
1738368005000000	99.8	10	99.7	20	100.2	15	100.3	25
1738368006000000	99.9	12	99.8	30	100.3	18	100.4	20
1738368007000000	100.0	20	99.9	30	100.4	20	100.5	25

Public Trades

Timestamp	Price	Size	Side
1738368005000000	100.35	buy	
1738368006000000	99.9	10	sell
1738368007000000	100.46	buy	

Constraints

As a market maker you can hold/short a maximum of 20 units.

Output Format

Timestamp	Bid Price	Ask Price
1738368005000000	99.9	100.4
1738368006000000	100.0	100.5

Sample Input 0

```
orderbook_train.csv
public_trades_train.csv
```