Data Scientist Technical Test

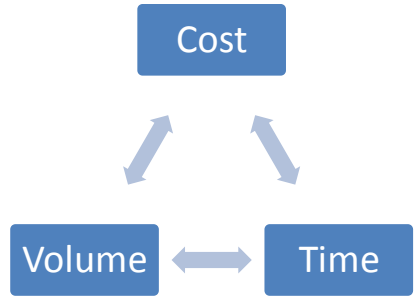
Background

The test is based on a real data science problem in Asset Management. The data which you will use for this test is artificial Tick data for a fictitious equity instrument.

Liquidity Risk

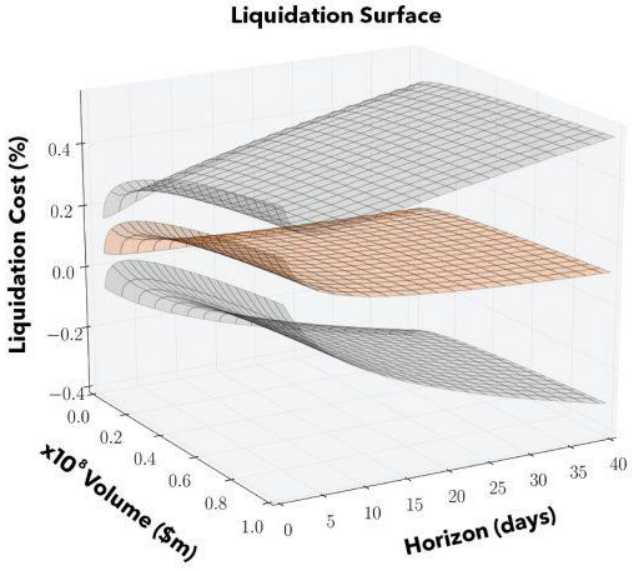
The task centers around predicting the cost of selling a security. In Asset Management, we always need to ensure that the portfolio has enough liquidity such that, should we have to sell a large amount of assets, not too large a cost would be incurred. Not accounting for this risk is what led to the downfall of Neil Woodford.

The cost of selling a security depends heavily on the volume and the time available to execute the trades (shown below).



$$cost = \alpha + \beta_1 volume + \beta_2 time + \varepsilon$$

Therefore, a plane can be drawn in a three dimensional space, and the cost of a trade can be calculated given a volume and time horizon.



Source: https://www.bloomberg.com/professional/blog/practically-speaking-determine-liquidity/

*The task therefore asks you to determine the cost of selling a given volume of security **ABC** within a specific time horizon. In doing so, we can give an indication of how liquid the security is and how costly it would be to sell a certain volume of assets.*

Data

The data consists of level 1 and level 2 tick data – aggregated into hour and minute windows respectively.

- Level 1 data provides the open, high, low and close price of the security, as well as the volume and time weighted price averages over the hour, and the average bid ask spread.
- Level 2 data provides the 10 levels of bid price and size, and of ask price and size. We have also calculated the normalized order book imbalance.

A data dictionary has been provided on page 3 of this document.

*Note: You **do not** need to use all of this data – your solution may just use a subset of the data available or just one of the data sources.*

Task

1. Download the dataset and work locally on your computer. Both Level 1 data and Level 2 tick data is provided for security **ABC**, an equity instrument, in two CSV files.
2. Build a model using Python to predict the cost of selling a given volume and time horizon of the security. You can use any model you like (e.g. Support Vector Machine, Random Forest or Neural Networks). Try and get your model to achieve a mean squared error below 5%.
3. Upload your code and results to GitHub (https://github.com/). Ensure your code is able to be executed once downloaded and give brief instructions on how it can be run.

Questions

1. Write a short outline of your approach to the task and any assumptions you made. Explain why you chose your approach with justification.
2. Explain how you might expand your approach to have a single model to predict the liquidity costs for any security in the market, rather than one model per security.
3. Explain how you will present your model to the liquidity business team? Would you create any charts of the model results? Would you utilise any visualisation tools in your presentation?
4. Explain how you would run your model using any cloud provider (e.g. Google Cloud) and which cloud components you would use. Give reasons for your choices.
5. Briefly describe how you would structure and scale the code of your model if a larger dataset was provided? *(optional)*
6. What changes would you make to your code to make it suitable to run on a production environment? *(optional)*

Submission

We require you to upload your source code to GitHub and send us an email with the link to the repository. In the email include the write-up of your answers to the above questions.

Data Dictionary

File: ABC_Level_One_Tick_Data.csv

- Time_Hour - Datetime hour window
- Instrument_Code - Unique identification of instrument
- Open - Instrument opening price over the hour window
- High - Instrument high price over the hour window
- Low - Instrument low price over the hour window
- Close - Instrument closing price over the hour window
- VWAP - Volume Weighted Average Price (VWAP) is a 'benchmark' price of the instrument for specified time
- TWAP - Time-weighted average price (TWAP) is the average price of an instrument for specified time
- NumberOfTrades - number of trades in the window
- Volume – Total volume of all the trades over the hour window
- Turnover - Turnover over the hour window, a measure of stock liquidity calculated by dividing the total number of shares traded over a period by the average number of shares outstanding for the period
- MinTimeHour - Minimum trade time in the window
- MaxTimeHour - Maximum trade time in the window
- Avg_Bid_Ask_Spread - Average spread between the level 1 bid and ask price for the hour window

File: ABC_Level_Two_Tick_Data.csv

- Instrument_Code - Unique identification of instrument
- Time_Minute - Datetime minute window
- L1_BidSize to L10_BidSize - Sum of the bid sizes for that level over the minute window
- L1_AskSize to L10_AskSize - Sum of the ask sizes for that level over the minute window
- L1_BidPrice to L10_BidPrice - Average of the bid prices for that level over the minute window
- L1_AskPrice to L10_AskPrice - Average of the ask prices for that level over the minute window
- Normalised_Order_Book_Imbalance - Normalised measure of whether there is excess buy orders or sell orders in the market, calculated as:

$$\left(\sum_{n=1}^{10} L_n BidSize - \sum_{n=1}^{10} L_n AskSize\right) \Big/ \left(\sum_{n=1}^{10} L_n BidSize + \sum_{n=1}^{10} L_n AskSize\right)$$