or - you send an order to Joe and Matt as you see the opportunity that they price is low

Successful if you send order quickly
Unsuccessful if people beat you and their apples run out

Optiver Execution Research Home Assignment

orders are sent
They are successful if they end up being traded
If the opportunity passes then it is unsuccessful
e.g with Apples => you buy apples at 0.1 to sell for 2 => order succeeds if you manage to sell them for the 2

The Problem

At Optiver we send thousands of orders each day, these orders all represent some opportunity we see in the market. You can think of an order as the intention to trade an opportunity presented to a trader. When an order trades it is deemed successful. On the other hand if the opportunity is withdrawn before our order arrives, or if another competitor's order arrives first, the order will not trade and is unsuccessful. The varying nature of these opportunities and how we try to execute these orders means they have a different probability to trade. For example, a less competitive order would have a higher chance to trade if all other variables were held constant. Unfortunately, due to our real-time trading requirements we cannot easily isolate these variables in our production trading system. This means most data we collect will have inherent biases or quirks in it. The task below requires you to understand some basic trading concepts and common terminology used at Optiver specifically: Edge, Latency, Gateways and what we call 'Success Rate'.

Perhaps try and find these biases and quirks
E.g. You can group Low edge opportunities that have extremely high competition and can see what is common

Perhaps a bias in these orders

each order has varying factors, You cannot isolate a model for each order, rather you give a general model to all orders

What is Edge?

Edge is the difference between what we think a product is worth (theoretical price), and for what price we could trade that product at this instant in time (actual price). This is shown as EDGE in the data set provided. This can be converted into euros (EDGE_EUR) by multiplying by the volume of the order and a fixed currency offset. VOLUME in this context is the number of contracts or products traded in that transaction. EDGE is a representation of how much we expect the trade to be worth per unit, and EDGE_EUR is how much the trade is worth in total.

It's worth noting that this number will not be perfectly related to PNL (profit and loss) as a low-edge trade is typically harder to convert into real PNL. You could think of it as low-edge trades are generally riskier than high-edge trades, or that higher edge trades are more obvious and therefore competitive. To give a real world example. Imagine a market where you could sell Apples at €2 each in a roadside stall. Now imagine you have the option of buying apples from two farmers:

1. Joe will sell you apples for 10c, however, he will only sell you 10 at this price.
    1. You can make a total of (€2 - 10c) * 10 = €19 (EDGE_EUR)
    2. Your EDGE is €1.90 per apple.
2. Matt will sell you apples for €1.90 but can sell you 190 Apples.
    1. You can make a total of (€2 - €1.90) * 190 = €19 (EDGE_EUR)
    2. Your EDGE is 10c per apple.

we think price is 2
High edge trade => actual price is 0.1
Low edge trade => actual price is 1.9

In both cases you can make the same amount of money, however, they do not have the same risk profile as you must handle (and then resell) a larger number of apples if you bought from Matt. We would prefer the trade with Joe because we can achieve the same outcome (EUR€) with a smaller risk profile. However, at the end of the day we would ideally do both trades as this has the maximum return of €38.

To summarize in a table:

	Low EDGE	High EDGE
Low EDGE_EUR	Lower Value, Higher Risk (Less Competitive)	Lower Value, Lower Risk (More Competitive)
High EDGE_EUR	Higher Value, Higher Risk (Less Competitive)	Higher Value, Lower Risk (More Competitive)

low profit per unit

Edge = profit you can get per unit

Edge_EUR = total profit you can get = Edge x volume x (currency adjustment)

What is Latency?

In a highly competitive trading environment many people will be competing (sending orders) for the same opportunities. However, only the first person to the exchange is able to trade the opportunity. This means the speed which we can see the event, calculate the opportunity and send an order is generally a determining factor in our success.

Time it takes to see an opportunty, calculate profit, and send order

We measure this as the time (in nanoseconds) between the timestamp of the market event that caused us to send an order and timestamp of the order we sent in response to that event. In this case a lower latency is better as it means we are able to make that decision and send the order faster. In the data-set below this is the column LATENCY. In the apples example above this would represent how quickly we could make the decision to buy either Joe or Matts' apples upon receiving their prices and considering our existing apples inventory. Note therefore that LATENCY represents our internal latency (the time between us seeing the event and sending an order), not the exchange latency.

Time from (when Matt and Joe release their prices, AND their price is lower than market price) TO when we send order to them

So time it takes to decide our order AND consider our inventory) => does not include Gateway

What is a Gateway?

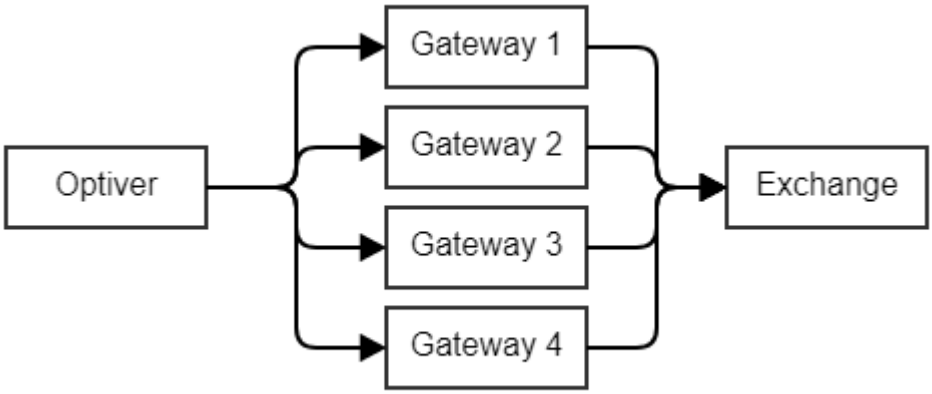
Ultimately, the first order to reach the exchange is the one that trades. In the apples example above this is similar to the first person who can tell Joe they'd like to buy his apples gets to buy them at 10c each. Unfortunately, when we decide to send an order there is not a single pathway to the exchange. This is like the road you take to the apple farmers. The highway might be faster during off-peak but the back roads are faster during peak hour.

Time it takes our order to reach the exchange

We call these different routes 'gateways' and they typically represent different servers or network paths on the exchange side. Since each order can only be sent down one gateway we must decide which gateway to use for any given opportunity and this can have an impact on if an order trades. Figure 1 shows a conceptual diagram of the gateway paths to the exchange. Gateway is denoted by the GW column in the data provided.

Pathway to the Exchange - each with varying time
Time = gateway

Figure 1: gateway definition



Find a metric for risk!

Success rate:
General = % of total orders that get traded (reach exchange first => have lowest latency + gateway)
To try and impoorve system

Success rate for High edge = % of high edge trades that succeed => tells us how we do for competitive trades

weighted success rate = split into certain category (e.g number of high egdge or high edge_euro) * weighted value * % that succeed

or Value based success = Total Edge_Euro earned / Total Edge_euro of orders

Success on orders on Particulary gateways in general and Over time
Success by risk

What is Success Rate?

The term we use to encompass what percentage of our orders trade is our 'success rate'. This can be trivially calculated by looking at the total percentage of orders that traded out of our complete set of orders. However, such an all encompassing statistic is of limited use when trying to improve our systems. In this case success rates can have a wide range of definitions depending on the context it is used in. For example, we might look at our success rate for high edge trades when considering how we're doing in competitive opportunities. We might also weight trades by edge as not all trades are equally valuable. These numbers could look quite different to our overall statistics but tell us different things. As part of an Execution Research role you will often be asked to come up with a suitable metric to answer questions from trading.

How well do we do on these types of trades
Can look at it from PnL
or PnL compared to market ... etc.

Task

You are a researcher at Optiver and you have been given a data set with the columns as shown in Table 1: Dataset Columns. This data represents 20 days of trading data from our automated trading system. Note that this dataset, while not real, is realistic – i.e. it is noisy, and values might be missing or invalid. Each row is an individual order with the columns containing the attributes of the orders. We're interested in how our success is changing over this time period and if there's anything the data can tell us. Using the tool of your choice (e.g. Python, R, Matlab):

1. Investigate this data and decide on some useful metrics for success rates.
2. With bullet points and the aid of a few charts present your view on what is happening with these success rates over the 20 day period.
3. Bullet points to cover any recommendations you have to improve success rates.

Do not spend time writing paragraphs. Bullet point conclusions and some charts are all that is needed. You can provide a PDF or a Jupyter notebook.
Please provide any code/scripts used to generate the statistics/plots used to draw your conclusions.

Table 1: Dataset Columns

Field	Type	Meaning
DAY	int	The date / day
EDGE	float	Edge in points (trade price - theoretical price)
EDGE_EUR	float	Edge converted into EUR (EDGE * VOLUME * EUR)
VOLUME	int	Volume in lots of contract traded.
GW	int	Numeric exchange gateway/path order was sent down
LATENCY	int	Total system latency for this order (in nanoseconds)
SUCCESS	int	1 for Successfully Traded, 0 for Failure

Noisy Dataset with the following columns
each row is an individual order on a given date

How does Our "Success" change over the time period?
What does the data tell us?

Investigate the data and find what useful metrics for success

Find things that can tell us info about how well we trade etc.

find how these success metrics change over time
present them in a graph

Compare them to each other

Remember, some gateways are faster at certain times

You can then make a model that shows:
Based on time and type of order
what is the best gateway to go to for success => Full data is only succesful orders

or can think of full orders

find probability of given orders trading

Find things that affect the probability of trading

Make a model to show what correct gateway to use for given order

Redo on current data set and compute efficiency

Can have Y variable = Given Success Probability
X = A given metric

Give recommendations on how to improve success rate
- Increase speed

Extra:
- speed of network / comms channels
- speed of models
- specific systems for orders of certain types
  - group assets, by edge and then have spe