Renan Peneluppi

renanpeneluppi@gmail.com | 416 830 7444 | https://github.com/RenanPeneluppi

Data Scientist applicant.

Winter 2021 Data Science Intern Challenge

Please complete the following questions, and provide your thought process/work. You can attach your work in a text file, link, etc. on the application page. Please ensure answers are easily visible for reviewers!

Question 1: Given some sample data, write a program to answer the following: click here to access the required data set

> On Shopify, we have exactly 100 sneaker shops, and each of these shops sells only one model of shoe. We want to do some analysis of the average order value (AOV). When we look at orders data over a 30 day window, we naively calculate an AOV of $3145.13. Given that we know these shops are selling sneakers, a relatively affordable item, something seems wrong with our analysis.

a. Think about what could be going wrong with our calculation. Think about a better way to evaluate this data.

A: The main issue with the first assumption for AOV was that it was done with plain mean calculation, without first cleaning the data and looking for information with values that are too high, too low, or just out of context. After exploring the data I found that user_id 607 has transactions on shop_id 42 for the amount 70400 and the transaction timestamp always at 4:00:00 time. This led me into assuming these are some sort of rebate or adjustment, so I chose to exclude these high amounts of transactions from the data set. Another curious finding was shop_id 78 had pretty expensive shoes (given each shop only sells one model), The amount for these shows are so high that one could even assume this was done by some sort of adjustment. I'm assuming it's in a category of its own, an outlier, so I chose to also exclude these transactions. After this cleaning, the AOV comes to 302.58, with a maximum order amount of 1760

and a minimum amount of 90. That makes more sense compared to the original 3145.13. But that doesn't really tell much about the data.

Anyway, there is more to this data, so I did some feature engineering to get the days the transactions are made (given they are all for the same month), We can also explore the time and weekday of these transactions.

b. What metric would you report for this dataset?

A: I first looked at the payment type proportion, looking for any methods that might stand-out, but that is pretty balanced overall.

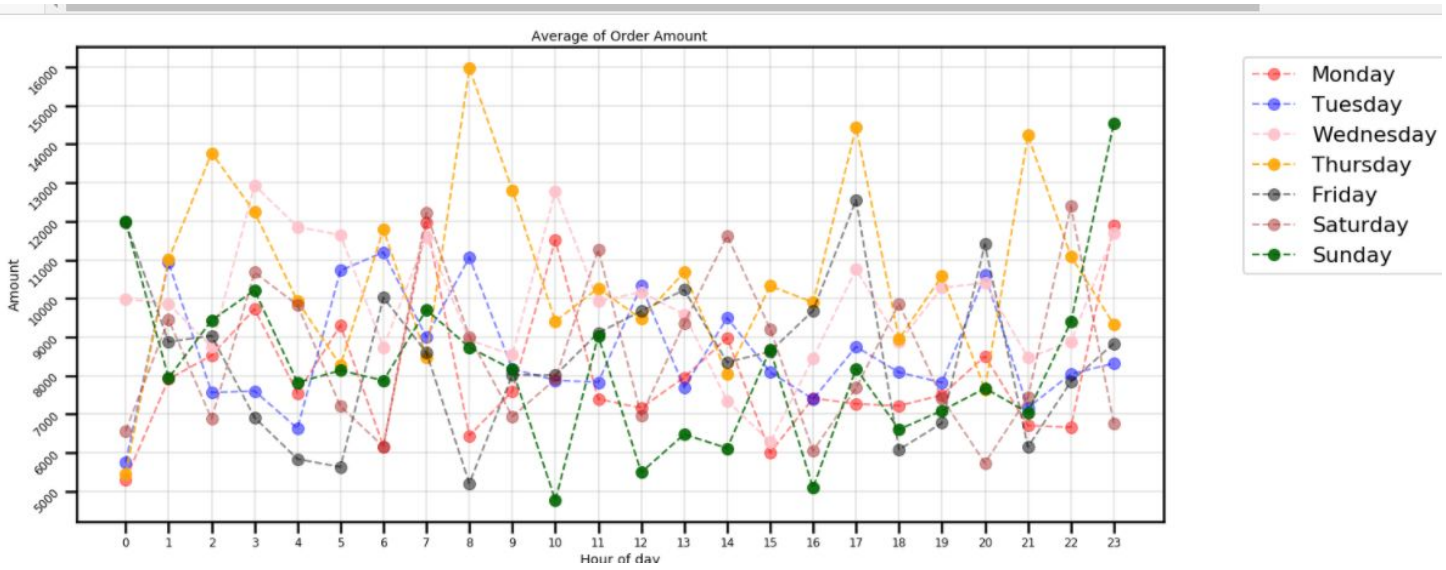
I then explored the time stamp to find patterns in time, day of the week and period of the month with the best performance. This was pretty interesting, and even could be further explored. Focusing on the days of the week and hour performance I could already get some interesting insights, with some more time I could further explore it to breakdown my findings for each shop, price bin group even customers.

As you can check on the bottom end of this file the time of the transactions have great value in this data set. After some feature engineering, I was able to check, for example, what time of the day and day of the week has the best average order amounts, and found out that Thursdays are the best day for sales, but also that lunchtime is a slow time of the day. In addition, the first and last weeks of the month had the best performance on order amounts.

c. What is its value?

This is a valuable file to help understand customer behaviour and plan for promotions, customer support, even server requirements.

The outliers and data that were dropped from this table could also be used to find suspicious transactions or understand operational procedures that can be improved, for example, all adjustments seem to be done using the same account but don't match the values for the shop's sales.



Question 2: For this question you'll need to use SQL. Follow this link to access the data set required for the challenge. Please use queries to answer the following questions. Paste your queries along with your final numerical answers below.

a. How many orders were shipped by Speedy Express in total?

After consulting the Shippers Table I know that Speedy Express ID is 1.

A: 54 orders were shipped by Speedy Express in a total of 3575 items.

SQL Statement:

```
SELECT count(*)
FROM [Orders]
Where [ShipperID] is 1
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 1

count(*)
54

SQL Statement:

```
SELECT sum(OD.Quantity) as Speedy_Express_QTY
FROM OrderDetails as OD JOIN(SELECT *
                             FROM Orders
                              WHERE ShipperID==1) as a ON OD.OrderID=a.OrderID
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 1

Speedy_Express_QTY
3575

b. What is the last name of the employee with the most orders?

A: The Employee with the most orders is Peacock, with a total of 40 orders.

SQL Statement:

```
SELECT Ord.OrderID, Ord.EmployeeID,EP.LastName as Employee_Last_Name, count(Ord.EmployeeID) as Total_Emp_Order
FROM Orders as Ord
JOIN Employees as Ep ON Ord.EmployeeID=Ep.EmployeeID
GROUP BY Ord.EmployeeID
ORDER BY Total_Emp_Order DESC
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 9

OrderID	EmployeeID	Employee_Last_Name	Tota
10250	4	Peacock	40
10251	3	Leverling	31
10258	1	Davolio	29
10262	8	Callahan	27
10265	2	Fuller	20
10249	6	Suyama	18
10289	7	King	14
10248	5	Buchanan	11
10255	9	Dodsworth	6

Peacock is also the employee with the highest QTY in the Orders table.

SQL Statement:

```
SELECT Emp.LastName,Emp.FirstName, sum(tb.Quantity) as TotalQuantity
FROM Employees as Emp
JOIN(SELECT Ord.EmployeeID, Ordet.ProductID, Ordet.Quantity
     FROM Orders as Ord
     JOIN OrderDetails as Ordet ON Ord.OrderID=Ordet.OrderID) as tb ON Emp.EmployeeID = tb.Er
GROUP BY Emp.LastName
```

Result:

Number of Records: 9

LastName	FirstName	TotalQuantit
Peacock	Margaret	3232
Davolio	Nancy	1924
Leverling	Janet	1725
Fuller	Andrew	1315
Callahan	Laura	1293
Suyama	Michael	1094
Buchanan	Steven	778
King	Robert	733
Dodsworth	Anne	649

c. What product was ordered the most by customers in Germany?

A: Boston Crab Meat is the most ordered product by customers in Germany.

```
Select  p.ProductName, MAX(t.QTY) as BestSellingProductInGermany
FROM Products as p
JOIN (SELECT Ger.Country, OD.ProductID, SUM(OD.Quantity) as QTY
      FROM (SELECT Ord.OrderID,Ord.CustomerID,Cus.Country
              FROM Orders as Ord
              JOIN Customers as Cus ON Ord.CustomerID=Cus.CustomerID

              WHERE Cus.Country = 'Germany') as Ger
      JOIN OrderDetails as OD ON Ger.OrderID=OD.OrderID
      GROUP BY OD.ProductID
      ORDER BY QTY DESC) as t on p.ProductID=t.ProductID
```

Result:

Number of Records: 1

ProductName	BestSellingProductInGe
Boston Crab Meat	160