Summer 2022 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!

- I have included my written answers for Question 1 and SQL query answers for Question 2 at the bottom of this document, please see the Jupyter notebook in this repository for quick Python code I used to answer Question 1.

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.
b. What metric would you report for this dataset?
c. What is its value?

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?
b. What is the last name of the employee with the most orders?
c. What product was ordered the most by customers in Germany?

My Answers

Question 1

a) Upon quick visual inspection of the data set, I notice right away that there are a few outliers which are transactions lying largely outside of the typical number of units sold and order total values. For example, a few times a transaction with 2000 units (order total of approximately $704 000) occurs which could skew the AOV which seems to be typically on the order of 1000 times less. Given the fact that the mean was used as the metric to report on this dataset and that the mean of a dataset will in general be sensitive to outliers, the outlier transactions are very likely the cause of the unrealistically high AOV. In order to rule out human error in recording the outlier transaction (i.e. not a real transaction at all/does not belong in data set), I calculated the unit price of sneakers in these transactions to be $352 per sneaker pair, which is consistent with other transactions from the corresponding shop_id (42). Therefore, these transactions were not due to human/recording error and indeed belong in this data set. Perhaps they are associated with a customer who purchases in bulk to then sell in their own business.

A better way to evaluate this data set may be to disregard these transactions as they are unrepresentative of typical transactions. I used a small code snippet (my code for this question can be found in the same repository as this file) in order to find that the outlier transaction occurs 17 times in the data set which has a total of 5000 transactions, meaning that these outliers account for about 0.34% of all transactions (not very many at all). I think it would be reasonable therefore to remove the outliers from the dataset and perform the AOV calculation without them.

b) I would report the median order amount (on the data set with outlier transactions removed) as the median is less sensitive to outliers.

c) The median order amount on the data set with outlier transactions removed is $284.0.

Question 2

a) SELECT Count(*) FROM Orders WHERE ShipperID=1;

answer=54

-the above answer was found using the knowledge that SpeedyExpress has a ShipperID=1

b) SELECT count(OrderID), EmployeeID
FROM Orders
GROUP BY EmployeeID
ORDER BY count(OrderID) desc;

SELECT EmployeeID, LastName FROM Employees;

answer=Peacock

c) SELECT ProductName, SUM(Quantity) AS TotalOrders
FROM Products p, Orders o, OrderDetails od, customers c
WHERE c.CustomerID = o.CustomerID AND c.Country = 'Germany' AND o.OrderID = od.OrderID AND od.ProductID = p.ProductID
GROUP BY ProductName
ORDER BY TotalOrders DESC
LIMIT 1;

answer=Boston Crab Meat