Shopify Summer 2022 Data Science Intern Challenge

January 10, 2022

1 Question 1

[126]:
```
# Library imports
import numpy as np
import pandas as pd
```

[127]:
```
# Data import
df = pd.read_excel(r"2019 Winter Data Science Intern Challenge Data Set_.xlsx")
```

[128]:
```
# Data View or EDA
df.head()
```

[128]:

	order_id	shop_id	user_id	order_amount	total_items	payment_method	created_at
0	1	53	746	224	2	cash	2017-03-13 12:36:56.190
1	2	92	925	90	1	cash	2017-03-03 17:38:51.999
2	3	44	861	144	1	cash	2017-03-14 04:23:55.595
3	4	18	935	156	1	credit_card	2017-03-26 12:43:36.649
4	5	18	883	156	1	credit_card	2017-03-01 04:35:10.773

a. Think about what could be going wrong with our calculation. Think about a better way to evaluate this data By just using the formula (Total Revenue/Total number of oders) we are not considering the presence of outliers in the data. Better way would be to remove the outliers as average or mean is such a factor which is highly influenced by the outliers because average or mean depends on the sum.

[129]:
```
# Data pre-processing

# Replacing invalid values with nan so that could be dropped later.
numberical_cols = df.select_dtypes(include=np.number).columns.to_list()
for col in numberical_cols:
```

```
    mask = (df[col] <= 0)
    df.loc[mask, col] = np.nan
    print(df[column].isna().sum()) # Printing total nans

# Replacing Outliers with nan
column = 'order_amount'
Q1 = df[column].quantile(.25)
Q3 = df[column].quantile(.75)
IQR = Q3 - Q1
lowerBound = Q1 - 1.5 * IQR # Upperbound
upperBound = Q3 + 1.5 * IQR # Lowerbound
mask = (df[column] < lowerBound) | (df[column] > upperBound)
df.loc[mask, column] = np.nan
print(df[column].isna().sum())

# Dropping nans
df = df.dropna()
print(df[column].isna().sum())
```

```
0
0
0
0
0
141
0
```

2 AOV = (Total Revenue)/(Total Orders)

[130]:
```
# Total Revenue
orderSum = np.sum(df[column])
```

[131]:
```
# Total Number of orders
numberOfOrders = df.shape[0]
```

[132]:
```
# Average order value
AOV = orderSum/numberOfOrders
```

[133]:
```
AOV
```

[133]: 293.7153735336489

2.0.1 AOV : 293.71$

b. What metric would you report for this dataset?

Ans: Revenue Per Visitor(RPV)

c. What is its value? RPV = Total Revenue / Total Visitor

[134]:
```
# Total Visitor
totalVisitor = len(df.user_id.unique())
```

[135]:
```
totalVisitor
```

[135]: 300

[136]:
```
RPV = orderSum/totalVisitor
```

[137]:
```
RPV
```

[137]: 4757.21

2.0.2 RPV: 4757.21$

[ ]:

[ ]:

3 Question 2

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

[ ]:

Answer: 68

Query: SELECT COUNT(OrderID) FROM Orders WHERE ShipperID is 3;

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

[ ]:

Answer: Peacock

Query: SELECT LastName FROM (SELECT * FROM (SELECT COUNT(OrderID) as Orders, EmployeeID FROM Orders GROUP BY EmployeeID) ORDER BY Orders DESC LIMIT 1) AS a INNER JOIN Employees ON a.EmployeeID = Employees.EmployeeID;

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

[ ]:

Answer: Gorgonzola Telino

Query: SELECT ProductName FROM (SELECT COUNT(b.OrderID) as SUM, ProductID FROM (SELECT Orders.OrderID, a.CustomerID, a.Country FROM (SELECT CustomerID, Country FROM Customers WHERE Country IS "Germany") as a INNER JOIN Orders ON Orders.CustomerID = a.CustomerID) AS b INNER JOIN OrderDetails ON b.OrderID = OrderDetails.OrderID group by ProductID order by SUM DESC LIMIT 1) AS c INNER JOIN Products ON c.ProductID = Products.ProductID;

[ ]: