import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

get\_ipython().run\_line\_magic('matplotlib', 'inline')

import seaborn as sns

data=pd.read\_csv(" Data Science Intern Challenge .csv")

data.head()

data.describe()

# Any missing entries

data.isnull().values.any()

# It has been reported that the'Average Order Value('AOV')is $3145.13. Lets Confirm

# In[16]:

data['order\_amount'].describe()

# By observing the statistics of order\_amount column,its clear that mean was considered of all the orders.There may be a reason like existence of outliers in dataset.Further , looking at the standarad deviation ,we can deduce that values on average are much farther away from the mean,making it poor estimate of AOV.

# The describe() function has also given us some tstas on the percentiles of the dataset,the values

# we can see that 75% of the data is belwo 390 ,and yet the AOV is well into thousands.Looking at min order of 90 vs max order of 704000 is so much farther from any of the other values,it has the potential to say the data is heavily upward in thiscase. Since the mean is susceptible to large or small outliers which can drastically affect its value,a much better estimate would reside in the median of the column.

ax=sns.scatterplot(x=data['order\_id'],y=data['order\_amount'])

# By the plot we can say that some orders have multiple sneakers in them.

# Some sneakers are valued very highly.

round(data.order\_amount.median(),2)

# The median value is just $284 while the average far exceed(11X)the median.Indiactive of outliers in the data and a skewed distribution.

# Instead of 'AOV' to evaluate sneaker price."AOV/item"should be the average metric to be used.To get this metric,order amount must be divided with the number of items in each other.

# Calculate AOV per sneaker.

data['order\_value\_per\_item']=data['order\_amount']/data['total\_items']

data.head()

round(data.order\_value\_per\_item.mean(),2)

# Thus,the average sneaker value is $387.74

data.order\_value\_per\_item.describe()

# By the observation it is clear that Sneakers are priced high($25,725),which makes the distribution skewed.

#

# Metric suitable for this dataset :In skewed distribution,instead of average ,Medains are better metric to report .

# In this case ,we will need to evaluate the frequency weighted median of sample (i.e.50% weighted percentile)is calculated

data.sort\_values('order\_value\_per\_item',inplace=True)

cummilative\_sum=data.total\_items.cumsum()

cutoff=data.total\_items.sum()/2.0

weighted\_median=data.order\_value\_per\_item[cummilative\_sum>=cutoff].iloc[0]

print(weighted\_median)

# Hence,the order size weighted median sneaker value is $352.0

Question2:

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

**Query**: SELECT ShipperName, COUNT (\*)

FROM (Orders

INNER JOIN Shippers ON Shippers.ShipperID=Orders.ShipperID)

WHERE ShipperName='Speedy Express’.

**Result:**

|  |  |
| --- | --- |
| Shipper Name | COUNT (\*) |
| Speedy Express | 54 |

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

**Query:** SELECT LastName, MAX(OrderID)

FROM (SELECT LastName,COUNT(\*) AS ORDERID

FROM (Orders

INNER JOIN Employees ON Orders.EmployeeID=Employees.EmployeeID)

GROUP By Employees.EmployeeID);

**Result:**

|  |  |
| --- | --- |
| LastName | MAX(OrderID) |
| Peacock | 40 |

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

**Query:** SELECT ProductID, ProductName,MAX(GermanyOrders)

FROM (SELECT Products.ProductID,ProductName,SUM(Quantity)AS GermanyOrders

FROM (((Products

INNER JOIN OrderDetails ON Products.ProductID=OrderDetails.ProductID)

INNERJOIN Orders ON OrderDetails.OrderID=Orders.OrderID)

INNER JOIN Customers ON Orders.CustomerID=Customers.CustomerID)

WHERE Country='Germany'

GROUP BY Products.ProductID);

**Result:**

|  |  |  |
| --- | --- | --- |
| ProductID | ProductName | MAX(GermanyOrders) |
| 40 | Boston Crab Meat | 160 |