import numpy as np

import pandas as pd

import plotly.express as px

import matplotlib.pyplot as plt

import seaborn as sns

----------------------------------------------------------------------------------------------

from google.colab import drive

drive.mount('/gdrive')

-----------------------------------------------------------------------------------------------

#the main data is called "data"

root = '/gdrive/MyDrive/csv/'

pd.set\_option('display.max\_columns' , None)

data = pd.read\_csv(root + 'csv\_1.csv')

data.head()

----------------------------------------------------------------------------------------------

data.info()

----------------------------------------------------------------------------------------------

#here we delete columns with less than 50% null data

data.columns[(data.isna().sum())/len(data)>0.50]

----------------------------------------------------------------------------------------------

#Now we need to shrink the range of data changes that are large

for cols in data.columns :

if data[cols].dtype =='int64' or data[cols].dtype == 'float64 :

data[cols] =(data[cols] - data[cols].mean())/data[cols].std()

----------------------------------------------------------------------------------------------

data.head()

----------------------------------------------------------------------------------------------

#Now we need to delete the int or float null data

data.fillna(data.\_get\_numeric\_data().mean(),inplace = True)

----------------------------------------------------------------------------------------------

data.isna().sum()

----------------------------------------------------------------------------------------------

#Now we need to fill in the data that is typed str or obj and is null

for cols in data.columns :

if data[cols].dtype == 'bool' or data[cols].dtype == 'object' :

data[cols].fillna(data[cols].value\_counts().head(1).index[0],inplace=True)

----------------------------------------------------------------------------------------------

data.isna().sum()

----------------------------------------------------------------------------------------------

#Now we need to find and delete outlier data

for cols in data.columns :

if data[cols].dtype == 'int64' or data[cols].dtype == 'float64':

upperRange = data[cols].mean()+3\*data[cols].std()

lowerRange = data[cols].mean() - 3 \*data[cols].std()

indexs = data[(data[cols]>upperRange) | (data[cols] < lowerRange)] .index

data2 =data.drop(indexs)

data2.shape

----------------------------------------------------------------------------------------------

#Number of ads in each geographical area

fig = px.pie(data, names='neighbourhood\_group', title='share in neighborhood')

fig.show()

----------------------------------------------------------------------------------------------

#Check general price indica tors

data[['neighbourhood\_group','price']].groupby(['neighbourhood\_group'] , as\_index = False).median().sort\_values(by='price', ascending= False)

data['host\_name'].value\_counts()

----------------------------------------------------------------------------------------------

#The connection between visits and the most popular places

review = data.sort\_values('number\_of\_reviews',ascending=False)

top\_reviewed = review.loc[:,['neighbourhood','number\_of\_reviews']][:20]

top\_reviewed = top\_reviewed.groupby('neighbourhood').mean().sort\_values('number\_of\_reviews',ascending=False).reset\_index()

fig4,ax3 = plt.subplots(figsize=(12,8))

sns.barplot(x=top\_reviewed['neighbourhood'],y=top\_reviewed['number\_of\_reviews'].values,color='yellowgreen',ax=ax3)

plt.plot(top\_reviewed['number\_of\_reviews'], marker='o', color='red',linestyle='--')

plt.ylabel('Reviews', fontsize='15')

plt.xlabel('Location',fontsize='15')

plt.ylim((400,580))

plt.title('Most-Reviewed Rentals by location',fontsize='15')

plt.show()

sns.set()

import numpy as np

upper\_east = data[data['neighbourhood'] == 'Upper East Side']

ninetieth\_percentile = np.quantile(upper\_east['number\_of\_reviews'], 0.85)

upper\_east = upper\_east[upper\_east['number\_of\_reviews'] >= ninetieth\_percentile]

upper\_east = upper\_east.sort\_values('price',ascending=True)

private\_room = upper\_east[upper\_east['room\_type'] == 'Private room'].reset\_index()

entire\_home = upper\_east[upper\_east['room\_type'] == 'Entire home/apt'].reset\_index()

shared\_room = upper\_east[upper\_east['room\_type'] == 'Shared room'].reset\_index()

private\_cheapest = private\_room.loc[0,:].reset\_index()

private\_cheapest.rename(columns={'index':'data','0':'values'},inplace=True)

entire\_cheapest = entire\_home.loc[0,:].reset\_index()

entire\_cheapest.rename(columns={'index':'data','0':'values'},inplace=True)