|  |  |  |
| --- | --- | --- |
| Zomoto data analysis using python | August 24  2024 | |
|  | | Name:M.Danusri |

**Project Overview:**Unveiling valuable insights from Zomato, a popular restaurant platform, requires the power of Python. Libraries like Pandas and Matplotlib become your allies in this task. Pandas helps you wrangle the Zomato data into a structured format, while Matplotlib brings it to life with informative visualizations. Through data exploration and analysis, you can uncover hidden trends. Perhaps you’ll identify popular cuisines by location or explore how pricing influences ratings. Python empowers you to ask questions of the data and uncover knowledge that can benefit both restaurants and diners.

**Objectives**

* Collect and preprocess Zomato data.
* Perform exploratory data analysis (EDA) to identify trends and patterns.
* Visualize data using Matplotlib or Seaborn to uncover insights.

**Skills Demonstrated**

* Data wrangling and preprocessing using Pandas.
* Exploratory data analysis (EDA).

Python and its following libraries are used to analyze Zomato data.

1. [Numpy](https://www.geeksforgeeks.org/python-numpy/)– With Numpy arrays, complex computations are executed quickly, and large calculations are handled efficiently.
2. [Matplotlib](https://www.geeksforgeeks.org/python-introduction-matplotlib/)– It has a wide range of features for creating high-quality plots, charts, histograms, scatter plots, and more.
3. [Pandas](https://www.geeksforgeeks.org/pandas-tutorial/)– The library simplifies the loading of data frames into 2D arrays and provides functions for performing multiple analysis tasks in a single operation.
4. [Seaborn](https://www.geeksforgeeks.org/introduction-to-seaborn-python/)– It offers a high-level interface for creating visually appealing and informative statistical graphics.

To address our analysis, we need to respond to the subsequent inquiries:

1. Do a greater number of restaurants provide online delivery as opposed to offline services?
2. Which types of restaurants are the most favored by the general public?
3. What price range is preferred by couples for their dinner at restaurants?

Before commencing the data analysis, the following steps are followed.

Following steps are followed before starting to analyze the data.

**Step 1: Import necessary Python libraries.**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

#### Step 2: Create the data frame.

Download the file containing the data using the [link](https://drive.google.com/file/d/1lamgErENUmuzvENgl4nMtbaJohwEP8KP/view?usp=sharing).

dataframe = pd.read\_csv("Zomato data .csv")

print(dataframe.head())

output:

name online\_order book\_table rate votes \

0 Jalsa Yes Yes 4.1/5 775

1 Spice Elephant Yes No 4.1/5 787

2 San Churro Cafe Yes No 3.8/5 918

3 Addhuri Udupi Bhojana No No 3.7/5 88

4 Grand Village No No 3.8/5 166

approx\_cost(for two people) listed\_in(type)

0 800 Buffet

1 800 Buffet

2 800 Buffet

3 300 Buffet

4 600 Buffet

def handleRate(value):

value=str(value).split('/')

value=value[0];

return float(value)

dataframe['rate']=dataframe['rate'].apply(handleRate)

print(dataframe.head())

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

def handleRate(value):

value=str(value).split('/')

value=value[0];

return float(value)

dataframe['rate']=dataframe['rate'].apply(handleRate)

print(dataframe.head())

output:

name online\_order book\_table rate votes \

0 Jalsa Yes Yes 4.1 775

1 Spice Elephant Yes No 4.1 787

2 San Churro Cafe Yes No 3.8 918

3 Addhuri Udupi Bhojana No No 3.7 88

4 Grand Village No No 3.8 166

approx\_cost(for two people) listed\_in(type)

0 800 Buffet

1 800 Buffet

2 800 Buffet

3 300 Buffet

4 600 Buffet

dataframe.info()

output:

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 148 entries, 0 to 147

Data columns (total 7 columns):

# Column Non-Null Count Dtype

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

0 name 148 non-null object

1 online\_order 148 non-null object

2 book\_table 148 non-null object

3 rate 148 non-null float64

4 votes 148 non-null int64

5 approx\_cost(for two people) 148 non-null int64

6 listed\_in(type) 148 non-null object

dtypes: float64(1), int64(2), object(4)

memory usage: 8.2+ KB

We will now examine the data frame for the presence of any null values. This stage scans each column to see whether there are any missing values or empty cells. This allows us to detect any potential data gaps that must be addressed.

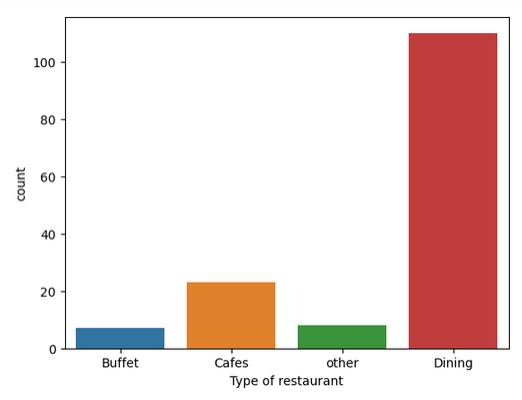
*There is no NULL value in dataframe.*

Let’s explore the listed\_in (type) column.

sns.countplot(x=dataframe['listed\_in(type)'])

plt.xlabel("Type of restaurant")

output:



***Conclusion:****The majority of the restaurants fall into the dining category.*

grouped\_data = dataframe.groupby('listed\_in(type)')['votes'].sum()

result = pd.DataFrame({'votes': grouped\_data})

plt.plot(result, c="green", marker="o")

plt.xlabel("Type of restaurant", c="red", size=20)

plt.ylabel("Votes", c="red", size=20)

output:



***Conclusion:****Dining restaurants are preferred by a larger number of individuals.*

Now we will determine the restaurant’s name that received the maximum votes based on a given dataframe.

max\_votes = dataframe['votes'].max()

restaurant\_with\_max\_votes = dataframe.loc[dataframe['votes'] == max\_votes, 'name']

print("Restaurant(s) with the maximum votes:")

print(restaurant\_with\_max\_votes)

output:

Restaurant(s) with the maximum votes:

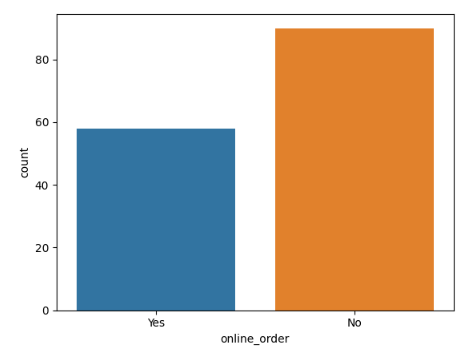
38 Empire Restaurant

Name: name, dtype: object

Let’s explore the online\_order column.

sns.countplot(x=data['online\_order'])

output:



***Conclusion:****This suggests that a majority of the restaurants do not accept online orders.*

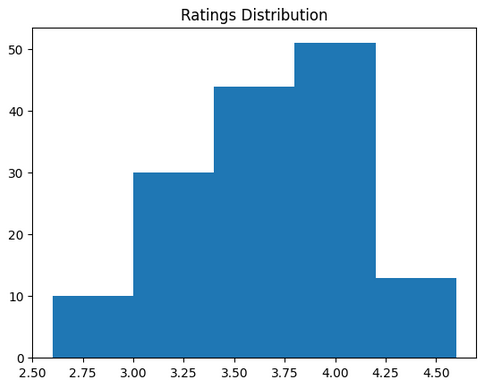
Let’s explore the rate column.

plt.hist(dataframe['rate'],bins=5)

plt.title("Ratings Distribution")

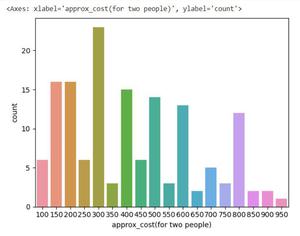
plt.show()

output:



***Conclusion:****The majority of restaurants received ratings ranging from 3.5 to 4.*

Let’s explore the approx\_cost(for two people) column.



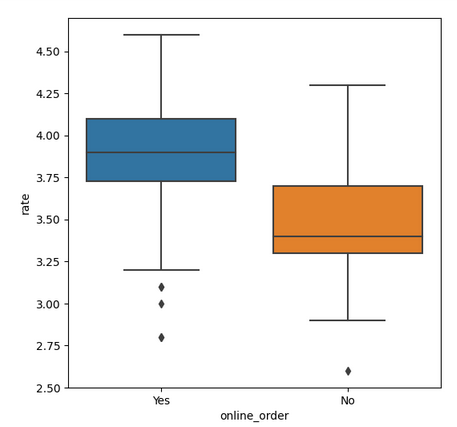
***Conclusion:****The majority of couples prefer restaurants with an approximate cost of 300 rupees.*

Now we will examine whether online orders receive higher ratings than offline orders.

plt.figure(figsize = (6,6))

sns.boxplot(x = 'online\_order', y = 'rate', data = dataframe)

output:



***CONCLUSION:****Offline orders received lower ratings in comparison to online orders, which obtained excellent ratings.*

pivot\_table = dataframe.pivot\_table(index='listed\_in(type)', columns='online\_order', aggfunc='size', fill\_value=0)

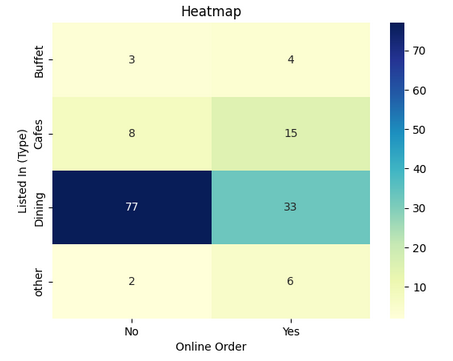
sns.heatmap(pivot\_table, annot=True, cmap="YlGnBu", fmt='d')

plt.title("Heatmap")

plt.xlabel("Online Order")

plt.ylabel("Listed In (Type)")

plt.show()



***CONCLUSION:****Dining restaurants primarily accept offline orders, whereas cafes primarily receive online orders. This suggests that clients prefer to place orders in person at restaurants, but prefer online ordering at cafes.*

# THANK YOU