zerotopandas-course-project

July 6, 2022

1 Zomato Restaurant Dataset Analysis

This dataset analysis project uses Zomato restaurant's in Hyderabad as it's target. The following dataset can be found on kaggle at [https://www.kaggle.com/datasets/deewakarchakraborty/zomato-restaurant-dataset] This notebook contains all the learnings that I could learn from Data Analysis with Python: Zero to Pandas do check out this awesome course if you want to build a cool project from scratch.I would be using Python (including libraries like NumPy , Pandas, Matplotlib , Seaborn). Let's get started!!

1.1 To start off lets download the Dataset

Before we do anything we would have to acquire the dataset.

[1]:
```
!pip install jovian opendatasets --upgrade --quiet
```

Let's begin by downloading the data, and listing the files within the dataset.

[2]:
```
# Here we are putting the url of data set
dataset_url = 'https://www.kaggle.com/datasets/deewakarchakraborty/
 ↪zomato-restaurant-dataset'
```

[3]:
```
import opendatasets as od
od.download(dataset_url)
```

```
Skipping, found downloaded files in "./zomato-restaurant-dataset" (use
force=True to force download)
```

The dataset has been downloaded and extracted.

[4]:
```
# Change this
data_dir = './zomato-restaurant-dataset'
```

[5]:
```
import os
os.listdir(data_dir)
```

[5]:
```
['HyderabadResturants.csv']
```

Let us save and upload our work to Jovian before continuing.

[6]:
```
project_name = "zomato-restaurant-analysis" # change this (use lowercase letters␣
↪and hyphens only)
```

[7]:
```
!pip install jovian --upgrade -q
```

[8]:
```
import jovian
```

[9]:
```
jovian.commit(project=project_name)
```

```
<IPython.core.display.Javascript object>
```

```
[jovian] Updating notebook "shubhammeena712/zomato-restaurant-analysis" on
https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/shubhammeena712/zomato-
restaurant-analysis
```

[9]:
```
'https://jovian.ai/shubhammeena712/zomato-restaurant-analysis'
```

1.2 Data Preparation and Cleaning

Now step 2 is to after loading the dataset let's load the dataset into a data frame using Pandas. After which we will explore the shape , range and values of different columns to get a broad idea of the dataset. Than we will handle the missing values , invalid inputs by a suitable method. These steps are what you may call as data preparation and cleaning. Later we will explore if we need to perform additional steps such as creating additional columns, merging another dataset.

[10]:
```
import numpy as np # linear algebra opertions to dataset
```

[11]:
```
import os
```

[12]:
```
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
```

[13]:
```
ls #this tell us all the list of everything present in the current dictonary
```

```
work/  zerotopandas-course-project.ipynb  zomato-
restaurant-dataset/
```

[14]:
```
os.getcwd() #This gives us our current location in terms of dictonary
```

[14]:
```
'/home/jovyan'
```

[15]:
```
os.listdir('./zomato-restaurant-dataset/')
```

[15]:
```
['HyderabadResturants.csv']
```

[16]:
```
data = pd.read_csv('./zomato-restaurant-dataset/HyderabadResturants.csv') #␣
↪Creating a data frame named data for the dataset
```

[17]:
```
data #calling the dataframe to confirm the import is correct
```

[17]:

	links	\
0	https://www.zomato.com/hyderabad/sahara-bakers...	
1	https://www.zomato.com/hyderabad/kfc-abids/order	
2	https://www.zomato.com/hyderabad/subbaiah-gari...	
3	https://www.zomato.com/hyderabad/paradise-biry...	
4	https://www.zomato.com/hyderabad/pista-house-b...	
..	...	
652	https://www.zomato.com/hyderabad/dr-cakes-banj...	
653	https://www.zomato.com/hyderabad/shahi-naan-am...	
654	https://www.zomato.com/hyderabad/combosthalam-...	
655	https://www.zomato.com/hyderabad/pachadis-by-p...	
656	https://www.zomato.com/hyderabad/tasim-1-himay...	

	names	ratings	\
0	Sahara Bakers	3.7	
1	KFC	3.9	
2	Subbaiah Gari Hotel	4.1	
3	Paradise Biryani	3.9	
4	Pista House Bakery	4.3	
..	...	...	
652	Dr Cakes	3.2	
653	Shahi Naan	-	
654	Combosthalam By Phulkaas	3.8	
655	Pachadis By Phulkaas	-	
656	Tasim	3.4	

	cuisine	price for one
0	Chinese, Bakery, Sichuan, Pizza, Burger	100
1	Burger, Fast Food, Biryani, Desserts, Beverages	100
2	South Indian, Andhra, Mithai	100
3	Biryani, Kebab, Desserts, Beverages	100
4	Fast Food, Sandwich, Pizza, Burger, Wraps, Rol...	100
..	...	...
652	Bakery, Desserts	350
653	North Indian	350
654	North Indian, Chinese	350
655	South Indian	350
656	Chinese, Momos	350

[657 rows x 5 columns]

Now here we are going to give data set some properties such as our desired custom color of the text and background color

1.3 links - Contains links to the order page of the restaurants.

1.4 name - Name of the restaurants.

1.5 ratings - Average of the rating given by the all the customers.

1.6 cuisines - Cuisine served by the restaurants.

1.7 Price For One - Cost of the food for one person in INR.

[18]:
```
data.head().style.set_properties(**{'background-color': 'black',
                                    'color': 'lawngreen',
                                    'border-color': 'white'})
```

[18]:
```
<pandas.io.formats.style.Styler at 0x7f112247eb50>
```

[19]:
```
# Let's get the shape (dimensions)

data.shape
```

[19]:
```
(657, 5)
```

[20]:
```
# What are all the various columns in dataset

data.columns
```

[20]:
```
Index(['links', 'names', 'ratings', 'cuisine', 'price for one'], dtype='object')
```

[21]:
```
data.describe(include='all')
```

[21]:

	links	names	ratings
count	657	657	657
unique	657	636	24
top	https://www.zomato.com/hyderabad/sahara-bakers...	Kwality Wall’s Frozen Dessert and Ice Cream Shop	4
freq	1	3	103
mean	NaN		
std	NaN		
min	NaN		
25%	NaN		
50%	NaN		
75%	NaN		
max	NaN		

```
mean                                                    NaN     NaN
std                                                     NaN     NaN
min                                                     NaN     NaN
25%                                                     NaN     NaN
50%                                                     NaN     NaN
75%                                                     NaN     NaN
max                                                     NaN     NaN
```

```
             cuisine  price for one
count            657     657.000000
unique           396            NaN
top     South Indian            NaN
freq              67            NaN
mean             NaN     169.406393
std              NaN      97.178712
min              NaN      50.000000
25%              NaN     100.000000
50%              NaN     150.000000
75%              NaN     250.000000
max              NaN     400.000000
```

What could be deduced from this :

1. Our dataset has 657 (restaurants) Rows 5 (values associated with each restaurant) Columns
2. Number of unique name (restaurant names) is 636/657
3. There were 396 unique cuisines.
4. Price range varies from a minimum of 50 to 400 maximum.

[22]:
```
#dropping_links
data.drop("links",axis=1,inplace=True)
data.head(10).style.set_properties(**{'background-color': 'black',
                                      'color': 'lawngreen',
                                      'border-color': 'white'})
```

[22]:
```
<pandas.io.formats.style.Styler at 0x7f112247e130>
```

[23]:
```
# Let's look one layer deeper into price for one

data['price for one'].describe()
```

[23]:
```
count    657.000000
mean     169.406393
std       97.178712
min       50.000000
25%      100.000000
50%      150.000000
75%      250.000000
max      400.000000
```

```
Name: price for one, dtype: float64
```

[24]:
```
# #shape before duplicate entries (657,4)
# #Dropping duplicates

data.drop_duplicates(inplace= True)
data.shape
```

[24]:
```
(657, 4)
```

[25]:
```
# Data Cleaning
# Checking not available (NULL)

data.isna().sum()
```

[25]:
```
names            0
ratings          0
cuisine          0
price for one    0
dtype: int64
```

[26]:
```
# Cleaning ratings col

data['ratings'].unique()
```

[26]:
```
array(['3.7', '3.9', '4.1', '4.3', '4', '4.2', '4.4', '4.5', '3.8', '4.6',
       '3.6', 'New', '3.4', '3.5', '3.1', '3.3', '2.7', '3.2', '-', '3',
       '4.7', '2.8', '2.9', '2.6'], dtype=object)
```

[27]:
```
#Removing New and '-' entries in ratings column

def modify(r_t):
    if(r_t=='New' or r_t=='-'): #This loop will iterate over rating and rturn␣
 ↪nan whenever given condition is satisfied
        return np.nan
    else:
        r_t = str(r_t).split('/')
        r_t = r_t[0]
        return float(r_t)
data['ratings'] = data['ratings'].apply(modify)
data['ratings'].fillna(round(data['ratings'].mean()),inplace=True)
data['ratings']
```

[27]:
```
0      3.7
1      3.9
2      4.1
3      3.9
4      4.3
```

```
      ...
652    3.2
653    4.0
654    3.8
655    4.0
656    3.4
Name: ratings, Length: 657, dtype: float64
```

[28]:
```
data.info() #Another check to verify data cleaning
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 657 entries, 0 to 656
Data columns (total 4 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   names           657 non-null    object
 1   ratings         657 non-null    float64
 2   cuisine         657 non-null    object
 3   price for one   657 non-null    int64
dtypes: float64(1), int64(1), object(2)
memory usage: 25.7+ KB
```

[29]:
```
data['cuisine'].value_counts() #This is to count the number how many times were␣
 ↪these specific iteam ordered under cuisine
```

[29]:
```
South Indian
67
Bakery, Desserts
23
Mithai, Street Food
22
Mithai
14
North Indian, Chinese
13
     ..
Chinese, North Indian, Sichuan
1
Mithai, Desserts, Street Food
1
Biryani, Cafe, North Indian, Continental, Street Food, Beverages, Desserts,
Pizza     1
South Indian, Street Food, Juices, Beverages
1
Chinese, Momos
1
Name: cuisine, Length: 396, dtype: int64
```

Now we have ensured there is no duplicate.

[30]:
```
import jovian
```

[31]:
```
jovian.commit()
```

```
<IPython.core.display.Javascript object>
```

```
[jovian] Updating notebook "shubhammeena712/zomato-restaurant-analysis" on
https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/shubhammeena712/zomato-
restaurant-analysis
```

[31]:
```
'https://jovian.ai/shubhammeena712/zomato-restaurant-analysis'
```

1.8 Exploratory Analysis and Visualization

Here we will try to analyze the data by the help of various plots.

Let's begin by importing`matplotlib.pyplot` and `seaborn`.

[32]:
```
import seaborn as sns
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline

sns.set_style('darkgrid')
matplotlib.rcParams['font.size'] = 14
matplotlib.rcParams['figure.figsize'] = (9, 5)
matplotlib.rcParams['figure.facecolor'] = '#00000000'
```

Count of number of cuisines

[33]:
```
#Data Visualization

# Count vs number of cuisines
import seaborn as sns
import matplotlib.pyplot as plt
plt.style.use("dark_background")
plt.figure(figsize = (16,10))
fx=sns.countplot(data['cuisine'])
plt.xticks(rotation=90)
```

```
/opt/conda/lib/python3.9/site-packages/seaborn/_decorators.py:36: FutureWarning:
Pass the following variable as a keyword arg: x. From version 0.12, the only
valid positional argument will be `data`, and passing other arguments without an
explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

[33]:
```
(array([  0,   1,   2,   3,   4,   5,   6,   7,   8,   9,  10,  11,  12,
         13,  14,  15,  16,  17,  18,  19,  20,  21,  22,  23,  24,  25,
```

```
         26,  27,  28,  29,  30,  31,  32,  33,  34,  35,  36,  37,  38,
         39,  40,  41,  42,  43,  44,  45,  46,  47,  48,  49,  50,  51,
         52,  53,  54,  55,  56,  57,  58,  59,  60,  61,  62,  63,  64,
         65,  66,  67,  68,  69,  70,  71,  72,  73,  74,  75,  76,  77,
         78,  79,  80,  81,  82,  83,  84,  85,  86,  87,  88,  89,  90,
         91,  92,  93,  94,  95,  96,  97,  98,  99, 100, 101, 102, 103,
        104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116,
        117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129,
        130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142,
        143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155,
        156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168,
        169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181,
        182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194,
        195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207,
        208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220,
        221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233,
        234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246,
        247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259,
        260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272,
        273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285,
        286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298,
        299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311,
        312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324,
        325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337,
        338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350,
        351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363,
        364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376,
        377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389,
        390, 391, 392, 393, 394, 395]),
 [Text(0, 0, 'Chinese, Bakery, Sichuan, Pizza, Burger'),
  Text(1, 0, 'Burger, Fast Food, Biryani, Desserts, Beverages'),
  Text(2, 0, 'South Indian, Andhra, Mithai'),
  Text(3, 0, 'Biryani, Kebab, Desserts, Beverages'),
  Text(4, 0, 'Fast Food, Sandwich, Pizza, Burger, Wraps, Rolls, Salad,
Desserts'),
  Text(5, 0, 'North Indian, Chinese, Mughlai, Mandi, Sichuan, Shawarma, Seafood,
Beverages'),
  Text(6, 0, 'South Indian, Chinese, North Indian'),
  Text(7, 0, 'North Indian'),
  Text(8, 0, 'North Indian, Biryani, Mughlai'),
  Text(9, 0, 'Ice Cream, Desserts'),
  Text(10, 0, 'Bakery, Desserts, Fast Food, Pizza'),
  Text(11, 0, 'Hyderabadi, Biryani'),
  Text(12, 0, 'South Indian'),
  Text(13, 0, 'Biryani'),
  Text(14, 0, 'Pizza, Fast Food, Desserts, Beverages'),
  Text(15, 0, 'Andhra, Biryani, South Indian, Desserts, Beverages, Hyderabadi'),
```

```
  Text(16, 0, 'Burger, Fast Food'),
  Text(17, 0, 'North Indian, Chinese, Andhra, Seafood'),
  Text(18, 0, 'Biryani, North Indian, Seafood, Chinese, Kebab'),
  Text(19, 0, 'Bakery, Fast Food, Pizza, Burger'),
  Text(20, 0, 'Biryani, South Indian, Mughlai, Andhra, Desserts, Hyderabadi'),
  Text(21, 0, 'Pizza, Pasta, Fast Food'),
  Text(22, 0, 'Bakery, Fast Food, Rolls, Chinese, Pizza, Burger'),
  Text(23, 0, 'Ice Cream'),
  Text(24, 0, 'Bakery, Chinese, Pizza, Fast Food'),
  Text(25, 0, 'Healthy Food, Fast Food, Sandwich, Salad, Wraps, Beverages'),
  Text(26, 0, 'Ice Cream, Desserts, Pizza'),
  Text(27, 0, 'North Indian, Chinese'),
  Text(28, 0, 'Shake, Ice Cream'),
  Text(29, 0, 'Andhra, North Indian, Chinese, Seafood, Beverages, Desserts'),
  Text(30, 0, 'Mithai'),
  Text(31, 0, 'Bakery, Desserts, Ice Cream, Pancake'),
  Text(32, 0, 'Beverages, Ice Cream, Desserts, Shake'),
  Text(33, 0, 'Chinese, North Indian'),
  Text(34, 0, 'Biryani, Mughlai, Kebab, Desserts'),
  Text(35, 0, 'Andhra'),
  Text(36, 0, 'Andhra, South Indian, Biryani, North Indian, Desserts'),
  Text(37, 0, 'North Indian, Biryani, Chinese, Healthy Food, Desserts,
Beverages'),
  Text(38, 0, 'North Indian, South Indian, Street Food, Fast Food, Desserts,
Beverages, Tea'),
  Text(39, 0, 'Shake, Ice Cream, Desserts'),
  Text(40, 0, 'North Indian, Chinese, Biryani, Seafood, Kebab, Mandi, Desserts,
Beverages'),
  Text(41, 0, 'South Indian, North Indian, Chinese'),
  Text(42, 0, 'Bakery, Fast Food'),
  Text(43, 0, 'North Indian, Chinese, Mughlai'),
  Text(44, 0, 'North Indian, South Indian, Chinese'),
  Text(45, 0, 'Mughlai, Biryani'),
  Text(46, 0, 'Fast Food, Pizza, Pasta, Desserts, Beverages, Shake, Coffee'),
  Text(47, 0, 'Rolls'),
  Text(48, 0, 'Momos, Fast Food'),
  Text(49, 0, 'Fast Food, Wraps, North Indian, Desserts, Beverages'),
  Text(50, 0, 'South Indian, North Indian'),
  Text(51, 0, 'Bakery, Desserts, Mithai, Fast Food, Pizza, Burger'),
  Text(52, 0, 'Kebab, Street Food, Wraps, Momos, Biryani, Shawarma'),
  Text(53, 0, 'Momos, Fast Food, Beverages'),
  Text(54, 0, 'Pizza, Fast Food'),
  Text(55, 0, 'Healthy Food, South Indian, North Indian, Biryani, Beverages'),
  Text(56, 0, 'Fast Food, North Indian, Healthy Food, Biryani, Desserts,
Beverages'),
  Text(57, 0, 'Cafe, Beverages, Fast Food, Tea'),
  Text(58, 0, 'Waffle, Desserts, Pancake'),
```

```
  Text(59, 0, 'South Indian, Beverages'),
  Text(60, 0, 'Mithai, Desserts, Street Food, Bakery'),
  Text(61, 0, 'Cafe, Beverages, Fast Food, Desserts, Tea'),
  Text(62, 0, 'Desserts, Ice Cream'),
  Text(63, 0, 'Ice Cream, Desserts, Fast Food, Pizza'),
  Text(64, 0, 'Mithai, Street Food'),
  Text(65, 0, 'North Indian, Mughlai, Chinese, Biryani'),
  Text(66, 0, 'Mexican, Fast Food, Wraps'),
  Text(67, 0, 'Chinese, Biryani'),
  Text(68, 0, 'Biryani, North Indian, Fast Food, Beverages, Desserts'),
  Text(69, 0, 'Ice Cream, Desserts, Waffle'),
  Text(70, 0, 'Kerala, Biryani, South Indian'),
  Text(71, 0, 'Chinese, Asian, Sichuan'),
  Text(72, 0, 'Bakery, Fast Food, Pancake'),
  Text(73, 0, 'Mithai, Street Food, North Indian, South Indian, Fast Food,
Beverages, Desserts, Pizza'),
  Text(74, 0, 'Chinese, South Indian, Sichuan'),
  Text(75, 0, 'South Indian, North Indian, Chinese, Ice Cream, Beverages,
Desserts'),
  Text(76, 0, 'North Indian, Biryani, Chinese, South Indian'),
  Text(77, 0, 'Healthy Food, Salad'),
  Text(78, 0, 'Fast Food, Burger, Beverages, Shake, Juices, Ice Cream'),
  Text(79, 0, 'Bakery, Desserts'),
  Text(80, 0, 'North Indian, Chinese, Biryani, Sichuan'),
  Text(81, 0, 'Bakery, Desserts, Coffee'),
  Text(82, 0, 'Chinese, Bakery'),
  Text(83, 0, 'Chinese, Continental, North Indian'),
  Text(84, 0, 'Bakery, Pizza, Fast Food, Burger'),
  Text(85, 0, 'Mexican, Healthy Food, American, Salad, Continental, Beverages'),
  Text(86, 0, 'Asian, North Indian, Pasta, Biryani'),
  Text(87, 0, 'South Indian, Tea'),
  Text(88, 0, 'Pizza, Rolls, Fast Food, Desserts, Bakery'),
  Text(89, 0, 'Beverages, Desserts, Lebanese, Shake, Shawarma'),
  Text(90, 0, 'Juices, Beverages'),
  Text(91, 0, 'Bengali, Chinese'),
  Text(92, 0, 'North Indian, Biryani, Arabian, Beverages, Mandi'),
  Text(93, 0, 'Arabian, North Indian, Chinese, Mandi, Beverages, Desserts,
Sichuan, Shawarma'),
  Text(94, 0, 'Mithai, Desserts, Juices, Fast Food'),
  Text(95, 0, 'Chinese, North Indian, Kebab, Biryani, Seafood, Beverages,
Mughlai, Sichuan'),
  Text(96, 0, 'North Indian, Chinese, South Indian, Mughlai, Desserts,
Beverages'),
  Text(97, 0, 'North Indian, Chinese, Andhra'),
  Text(98, 0, 'South Indian, Street Food'),
  Text(99, 0, 'North Indian, Street Food, Fast Food, BBQ, Desserts'),
  Text(100, 0, 'South Indian, North Indian, Chinese, Biryani, Beverages,
```

```
Desserts'),
  Text(101, 0, 'Burger, Desserts, Tea, Coffee'),
  Text(102, 0, 'North Indian, Mughlai, Kebab, Chinese, Beverages, Desserts'),
  Text(103, 0, 'Momos'),
  Text(104, 0, 'South Indian, Chinese, North Indian, Biryani'),
  Text(105, 0, 'Biryani, Mughlai, Chinese, Kebab, Sichuan'),
  Text(106, 0, 'Beverages, Juices, Shake, Healthy Food'),
  Text(107, 0, 'Juices, Beverages, Fast Food, Shake'),
  Text(108, 0, 'Desserts, Beverages'),
  Text(109, 0, 'Bakery, Desserts, Ice Cream'),
  Text(110, 0, 'North Indian, South Indian, Chinese, Street Food, Fast Food,
Beverages'),
  Text(111, 0, 'Ice Cream, Desserts, Pizza, Fast Food, Street Food, Beverages'),
  Text(112, 0, 'South Indian, Chinese, North Indian, Juices, Beverages,
Desserts, Sichuan'),
  Text(113, 0, 'South Indian, Desserts, Beverages'),
  Text(114, 0, 'South Indian, Chinese'),
  Text(115, 0, 'Mithai, North Indian, Street Food, South Indian, Beverages'),
  Text(116, 0, 'Beverages, Desserts, Ice Cream, Waffle'),
  Text(117, 0, 'Healthy Food, Desserts'),
  Text(118, 0, 'Fast Food, Sandwich, Shake, Beverages, Burger, Coffee'),
  Text(119, 0, 'Fast Food, Momos'),
  Text(120, 0, 'North Indian, Chinese, Mughlai, Asian, Mandi, Desserts, Biryani,
BBQ'),
  Text(121, 0, 'Chinese, Biryani, North Indian'),
  Text(122, 0, 'South Indian, Street Food, Beverages'),
  Text(123, 0, 'Bakery, Desserts, Fast Food, Street Food, Pizza, Burger'),
  Text(124, 0, 'South Indian, Juices'),
  Text(125, 0, 'South Indian, Chinese, North Indian, Sichuan, Pizza'),
  Text(126, 0, 'Bakery, Chinese, Sandwich, Burger, Pizza, Fast Food'),
  Text(127, 0, 'Juices, Beverages, Ice Cream, Shake'),
  Text(128, 0, 'South Indian, Chinese, North Indian, Beverages, Desserts,
Sandwich'),
  Text(129, 0, 'Desserts, Bakery'),
  Text(130, 0, 'Pizza, Pasta, Fast Food, Desserts, Beverages'),
  Text(131, 0, 'Chinese, Singaporean, Thai, Seafood, Ice Cream, Beverages, North
Indian, Sichuan'),
  Text(132, 0, 'Fast Food, Sandwich, Burger, Pizza, Mexican, Beverages'),
  Text(133, 0, 'Biryani, South Indian, Beverages'),
  Text(134, 0, 'Desserts, Fast Food, Street Food, Pizza, Ice Cream, Beverages'),
  Text(135, 0, 'Cafe, Fast Food, Coffee, Desserts, Beverages'),
  Text(136, 0, 'Beverages, Coffee'),
  Text(137, 0, 'Waffle, Desserts'),
  Text(138, 0, 'North Indian, Street Food'),
  Text(139, 0, 'South Indian, Chinese, Fast Food, North Indian, Street Food,
Biryani, Sandwich, Beverages'),
  Text(140, 0, 'Beverages, Juices, Salad, Shake, Desserts'),
```

```
  Text(141, 0, 'South Indian, Street Food, Juices, Beverages'),
  Text(142, 0, 'Hyderabadi'),
  Text(143, 0, 'Biryani, Cafe, North Indian, Continental, Street Food,
Beverages, Desserts, Pizza'),
  Text(144, 0, 'Mithai, Desserts, Street Food'),
  Text(145, 0, 'North Indian, Fast Food'),
  Text(146, 0, 'Bakery, Desserts, Fast Food'),
  Text(147, 0, 'Bakery'),
  Text(148, 0, 'Chinese, North Indian, Sichuan'),
  Text(149, 0, 'Maharashtrian, Fast Food, Street Food, Beverages'),
  Text(150, 0, 'Cafe, American, Continental, Italian, Coffee, Beverages, Pizza,
Burger'),
  Text(151, 0, 'Healthy Food, Juices, Beverages'),
  Text(152, 0, 'Street Food'),
  Text(153, 0, 'Street Food, Fast Food, North Indian, Maharashtrian, Tea'),
  Text(154, 0, 'Tea, Beverages, Fast Food'),
  Text(155, 0, 'North Indian, Chinese, Fast Food, Beverages, Sichuan, Pizza'),
  Text(156, 0, 'North Indian, Chinese, Biryani'),
  Text(157, 0, 'South Indian, Chinese, Sichuan'),
  Text(158, 0, 'Pizza, Fast Food, Pasta, Sandwich, Beverages, Shake, Ice
Cream'),
  Text(159, 0, 'Cafe, Continental, Beverages'),
  Text(160, 0, 'North Indian, Chinese, Mughlai, Seafood, Kebab, Biryani,
Beverages, Sichuan'),
  Text(161, 0, 'Mexican, Italian, Desserts, Pizza'),
  Text(162, 0, 'North Indian, Biryani, Chinese'),
  Text(163, 0, 'North Indian, Chinese, Mughlai, Seafood, Andhra, Biryani,
Desserts, Beverages'),
  Text(164, 0, 'Beverages, Ice Cream, Salad'),
  Text(165, 0, 'North Indian, Biryani, Mughlai, BBQ'),
  Text(166, 0, 'Beverages, Shake, Ice Cream'),
  Text(167, 0, 'Bakery, Fast Food, Desserts, Sandwich, Burger, Rolls, Pizza'),
  Text(168, 0, 'Mithai, Desserts'),
  Text(169, 0, 'Street Food, South Indian'),
  Text(170, 0, 'Tea, Shake, Beverages, Street Food'),
  Text(171, 0, 'South Indian, Chinese, Street Food'),
  Text(172, 0, 'Cafe, Continental, Italian, Fast Food, Pizza, Desserts,
Beverages, Burger'),
  Text(173, 0, 'Fast Food, Beverages, Shake, Pizza, Momos, Shawarma'),
  Text(174, 0, 'Beverages, Fast Food'),
  Text(175, 0, 'Bakery, Fast Food, Beverages, Pizza, Burger'),
  Text(176, 0, 'North Indian, Kebab, Biryani, Awadhi, Hyderabadi, Modern Indian,
Desserts, Beverages'),
  Text(177, 0, 'South Indian, Beverages, Tea'),
  Text(178, 0, 'Fast Food, Roast Chicken'),
  Text(179, 0, 'Mithai, Fast Food, Street Food'),
  Text(180, 0, 'South Indian, Hyderabadi'),
```

```
  Text(181, 0, 'South Indian, Beverages, Desserts'),
  Text(182, 0, 'Beverages, Juices, Shake'),
  Text(183, 0, 'Ice Cream, Beverages, Desserts, Shake, Tea'),
  Text(184, 0, 'Bakery, Mithai, Fast Food, Desserts'),
  Text(185, 0, 'Beverages, Desserts, Ice Cream, Shake'),
  Text(186, 0, 'North Indian, Kebab, Biryani, Hyderabadi, Mughlai, Desserts'),
  Text(187, 0, 'North Indian, Biryani, Chinese, Mughlai, Andhra, Beverages'),
  Text(188, 0, 'Desserts, Shake'),
  Text(189, 0, 'Bakery, Ice Cream, Desserts'),
  Text(190, 0, 'Fast Food, Momos, Shawarma'),
  Text(191, 0, 'South Indian, North Indian, Chinese, Fast Food, Beverages,
Shake'),
  Text(192, 0, 'Rolls, Fast Food, Sichuan'),
  Text(193, 0, 'Mithai, Street Food, Fast Food'),
  Text(194, 0, 'Fast Food, Coffee, Beverages'),
  Text(195, 0, 'Asian, North Indian, Chinese, Mexican, Italian, Desserts,
Beverages, Sichuan'),
  Text(196, 0, 'Beverages, Fast Food, Pizza'),
  Text(197, 0, 'Chinese, Sichuan'),
  Text(198, 0, 'Cafe, Coffee, Burger, Sandwich, Beverages, Desserts, Tea,
Continental'),
  Text(199, 0, 'Mughlai, North Indian, Andhra, Kebab'),
  Text(200, 0, 'South Indian, Biryani'),
  Text(201, 0, 'North Indian, South Indian, Chinese, Desserts'),
  Text(202, 0, 'Chinese, North Indian, Biryani'),
  Text(203, 0, 'Biryani, Chinese'),
  Text(204, 0, 'Bakery, Fast Food, South Indian, Beverages, Shake, Pizza,
Burger, Tea'),
  Text(205, 0, 'Biryani, North Indian, Chinese'),
  Text(206, 0, 'Burger'),
  Text(207, 0, 'North Indian, Chinese, South Indian, Beverages'),
  Text(208, 0, 'South Indian, North Indian, Biryani, Hyderabadi, Chinese'),
  Text(209, 0, 'Hyderabadi, Chinese'),
  Text(210, 0, 'Cafe, Italian, Mexican, Fast Food, Desserts, Beverages, Sichuan,
Pizza'),
  Text(211, 0, 'Ice Cream, Desserts, Healthy Food'),
  Text(212, 0, 'Cafe, Fast Food, Burger, Chinese, Sandwich, Desserts, Shake,
Beverages'),
  Text(213, 0, 'Street Food, Beverages'),
  Text(214, 0, 'Cafe, Chinese, Fast Food, Beverages, Sichuan, Pizza, Burger,
Momos'),
  Text(215, 0, 'Cafe, Fast Food, Desserts, Beverages, Shake, Tea'),
  Text(216, 0, 'Italian, Beverages, Pizza'),
  Text(217, 0, 'Cafe, Bakery, Desserts, Pizza, Fast Food, Beverages, Italian'),
  Text(218, 0, 'Biryani, Mughlai, Lucknowi, Hyderabadi, Kebab, North Indian,
Desserts, Beverages'),
  Text(219, 0, 'Fast Food, Street Food, Tea, Pizza, Sandwich'),
```

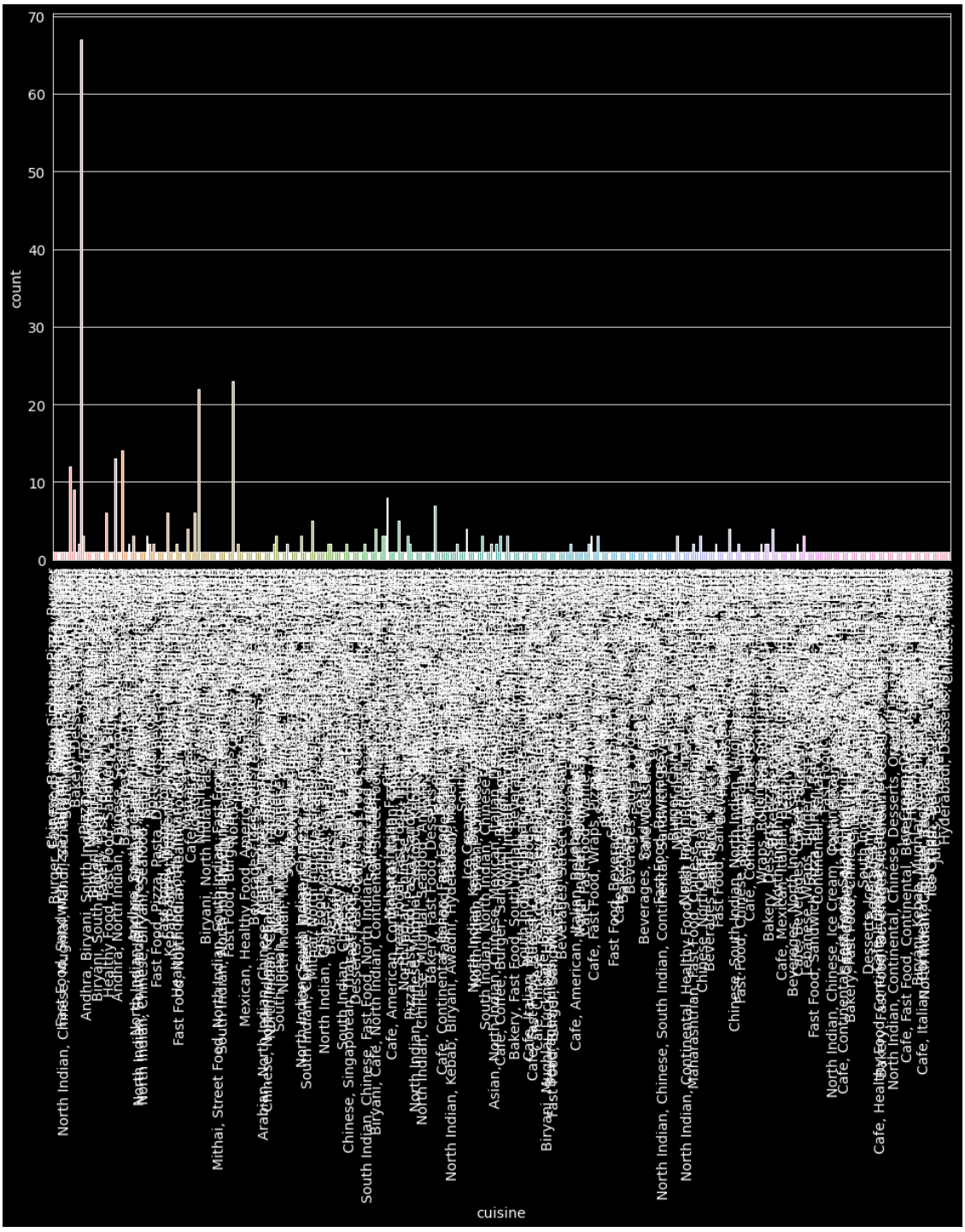
```
  Text(220, 0, 'Beverages, Juices, Shake, Salad, Pizza, Desserts, Healthy Food,
Sandwich'),
  Text(221, 0, 'Fast Food, Burger, Sandwich, Street Food, Chinese, Pizza,
Beverages, Sichuan'),
  Text(222, 0, 'Pizza'),
  Text(223, 0, 'North Indian, Andhra, Chinese, Beverages, Desserts'),
  Text(224, 0, 'Fast Food, Chinese, North Indian'),
  Text(225, 0, 'Beverages, Street Food, Sandwich, Salad, Shake, Juices'),
  Text(226, 0, 'Biryani, Hyderabadi, Kebab'),
  Text(227, 0, 'North Indian, Healthy Food'),
  Text(228, 0, 'Shake, Beverages'),
  Text(229, 0, 'Burger, Fast Food, Beverages, Shake'),
  Text(230, 0, 'Lebanese, Arabian, Rolls, Fast Food, Pizza'),
  Text(231, 0, 'Cafe, American, Italian, Beverages, Fast Food, Shake, Burger,
Waffle'),
  Text(232, 0, 'Waffle, Pancake, Beverages, Shake'),
  Text(233, 0, 'North Indian, South Indian, Chinese, Kebab, Sichuan'),
  Text(234, 0, 'Fast Food, Bakery, Desserts'),
  Text(235, 0, 'Cafe, Fast Food, Waffle, Pancake, Coffee, Beverages'),
  Text(236, 0, 'North Indian, South Indian'),
  Text(237, 0, 'Healthy Food'),
  Text(238, 0, 'Fast Food, Pizza, Momos'),
  Text(239, 0, 'Cafe, Fast Food, Wraps, Pizza, Beverages, Shake, Desserts'),
  Text(240, 0, 'North Indian, Chinese, Sichuan'),
  Text(241, 0, 'Fast Food, Beverages, Desserts'),
  Text(242, 0, 'South Indian, North Indian, Andhra'),
  Text(243, 0, 'South Indian, Biryani, North Indian'),
  Text(244, 0, 'Beverages, Desserts, Bakery, Shake'),
  Text(245, 0, 'Bakery, Fast Food, Shake, Beverages'),
  Text(246, 0, 'Juices'),
  Text(247, 0, 'South Indian, North Indian, Beverages'),
  Text(248, 0, 'Fast Food, Beverages, Coffee, Pizza, Burger, Momos, Tea'),
  Text(249, 0, 'Chinese, Mughlai'),
  Text(250, 0, 'Cafe, Continental, Healthy Food, Italian, Pizza, Tea'),
  Text(251, 0, 'Salad, Desserts, Beverages, Shake'),
  Text(252, 0, 'Bakery, Desserts, Fast Food, Burger, Shawarma'),
  Text(253, 0, 'Tea, Beverages, Fast Food, Shake, Coffee'),
  Text(254, 0, 'Pizza, Italian'),
  Text(255, 0, 'Beverages, Street Food, Shake, Sandwich, Tea'),
  Text(256, 0, 'Shawarma'),
  Text(257, 0, 'Mughlai, Hyderabadi'),
  Text(258, 0, 'Fast Food, Bakery, Desserts, Shake'),
  Text(259, 0, 'Fast Food, Momos, Beverages'),
  Text(260, 0, 'Fast Food, Desserts, Beverages'),
  Text(261, 0, 'Beverages, Sandwich, Desserts, Shake, Healthy Food'),
  Text(262, 0, 'Cafe, Desserts, Pizza, Burger, Waffle, Tea'),
  Text(263, 0, 'Mughlai, Chinese'),
```

```
  Text(264, 0, 'Chinese, Beverages, Shake'),
  Text(265, 0, 'Chinese, Lebanese, Biryani'),
  Text(266, 0, 'Chinese, Sichuan, Momos'),
  Text(267, 0, 'Street Food, Fast Food'),
  Text(268, 0, 'Fast Food, Beverages, Biryani, Desserts, Burger'),
  Text(269, 0, 'North Indian, Chinese, South Indian, Continental, Sandwich,
Beverages, Juices, Ice Cream'),
  Text(270, 0, 'Cafe'),
  Text(271, 0, 'Fast Food, Lebanese'),
  Text(272, 0, 'Hyderabadi, North Indian'),
  Text(273, 0, 'Asian, Biryani, North Indian'),
  Text(274, 0, 'Burger, Chinese, Fast Food, Beverages'),
  Text(275, 0, 'Desserts'),
  Text(276, 0, 'North Indian, Chinese, Biryani, Desserts'),
  Text(277, 0, 'Juices, Beverages, Shake'),
  Text(278, 0, 'Mughlai, North Indian'),
  Text(279, 0, 'North Indian, Rajasthani, Fast Food, Street Food'),
  Text(280, 0, 'North Indian, Continental, Healthy Food, Fast Food, Burger,
Pasta, Beverages, Desserts'),
  Text(281, 0, 'Juices, Healthy Food'),
  Text(282, 0, 'Mughlai'),
  Text(283, 0, 'Maharashtrian, Fast Food, Chinese, North Indian, Bakery, Mithai,
Desserts'),
  Text(284, 0, 'Fast Food, Healthy Food, Beverages'),
  Text(285, 0, 'North Indian, Fast Food, Desserts'),
  Text(286, 0, 'Cafe, Bakery, Pizza, Italian, Healthy Food, Tea'),
  Text(287, 0, 'Chinese, North Indian, Biryani, Mughlai, Sichuan, Shawarma'),
  Text(288, 0, 'Chinese, South Indian'),
  Text(289, 0, 'Biryani, Chinese, Shawarma'),
  Text(290, 0, 'Beverages, Sandwich, Juices, Salad, Healthy Food, Shake'),
  Text(291, 0, 'Fast Food, Street Food, Ice Cream, Beverages'),
  Text(292, 0, 'Fast Food, Shawarma'),
  Text(293, 0, 'Ice Cream, Desserts, Shake'),
  Text(294, 0, 'Fast Food, Sandwich, Ice Cream, Beverages, Shake'),
  Text(295, 0, 'Beverages, Fast Food, Shake, Waffle'),
  Text(296, 0, 'Healthy Food, Beverages, Shake'),
  Text(297, 0, 'Chinese, North Indian, Mughlai'),
  Text(298, 0, 'Beverages'),
  Text(299, 0, 'Cafe, Pizza, Fast Food'),
  Text(300, 0, 'South Indian, Pizza'),
  Text(301, 0, 'Chinese, South Indian, North Indian, Mughlai, Street Food,
Biryani'),
  Text(302, 0, 'Pizza, Burger, Fast Food'),
  Text(303, 0, 'Bakery, Desserts, Shake'),
  Text(304, 0, 'Fast Food, Chinese, North Indian, Biryani, Mughlai, Beverages'),
  Text(305, 0, 'Street Food, Andhra'),
  Text(306, 0, 'Beverages, Shake, Tea, Juices, Street Food'),
```

```
  Text(307, 0, 'Cafe, Fast Food, Chinese, Beverages'),
  Text(308, 0, 'Cafe, Continental, Fast Food, Desserts, Beverages'),
  Text(309, 0, 'Beverages, Tea'),
  Text(310, 0, 'Paan, Ice Cream'),
  Text(311, 0, 'Paan'),
  Text(312, 0, 'Chinese, South Indian, North Indian'),
  Text(313, 0, 'Wraps, Rolls, Fast Food, Desserts, Beverages'),
  Text(314, 0, 'North Indian, Chinese, Seafood'),
  Text(315, 0, 'Andhra, Hyderabadi'),
  Text(316, 0, 'Bakery, Desserts, Pizza, Sandwich, Burger, Fast Food'),
  Text(317, 0, 'Fast Food'),
  Text(318, 0, 'Bakery, Desserts, Fast Food, Beverages'),
  Text(319, 0, 'Ice Cream, Desserts, Beverages'),
  Text(320, 0, 'North Indian, Fast Food, Wraps, Desserts, Shake'),
  Text(321, 0, 'Biryani, Beverages'),
  Text(322, 0, 'Cafe, Mexican, Italian, Pizza, Coffee, Tea, Desserts, Burger'),
  Text(323, 0, 'North Indian, Mughlai, Biryani'),
  Text(324, 0, 'Healthy Food, Fast Food, Beverages'),
  Text(325, 0, 'Burger, Fast Food, Desserts'),
  Text(326, 0, 'Sandwich, Beverages'),
  Text(327, 0, 'Beverages, North Indian, Asian, Continental, Sichuan, Waffle'),
  Text(328, 0, 'Mithai, Street Food, Desserts'),
  Text(329, 0, 'Cafe, Beverages, Coffee, Shake, Tea, Bubble Tea, Pancake'),
  Text(330, 0, 'Fast Food, Shake, Beverages, Sichuan'),
  Text(331, 0, 'South Indian, Andhra'),
  Text(332, 0, 'Beverages, South Indian'),
  Text(333, 0, 'Lebanese, Wraps, Burger, Fast Food, Desserts, Shawarma'),
  Text(334, 0, 'Fast Food, Chinese, Sichuan'),
  Text(335, 0, 'Fast Food, Beverages'),
  Text(336, 0, 'Fast Food, Sandwich, Pizza, Burger, Beverages, Waffle, Tea,
Coffee'),
  Text(337, 0, 'Fast Food, Shake, Beverages'),
  Text(338, 0, 'Continental, Italian, American, Cafe, Beverages'),
  Text(339, 0, 'Biryani, South Indian'),
  Text(340, 0, 'Ice Cream, Desserts, Cafe'),
  Text(341, 0, 'Fast Food, Burger, Pizza, Sandwich'),
  Text(342, 0, 'Biryani, Chinese, Shake'),
  Text(343, 0, 'Andhra, Street Food'),
  Text(344, 0, 'North Indian, Chinese, Ice Cream, Continental, Desserts,
Beverages, Shake'),
  Text(345, 0, 'Paan, Beverages, Street Food, Fast Food'),
  Text(346, 0, 'Lebanese, Sandwich, Burger'),
  Text(347, 0, 'Fast Food, Street Food'),
  Text(348, 0, 'Burger, Sandwich, Fast Food'),
  Text(349, 0, 'Cafe, Continental, Fast Food, Salad, Healthy Food, Shake,
Coffee, Desserts'),
  Text(350, 0, 'South Indian, North Indian, Chinese, Beverages, Desserts'),
```

```
  Text(351, 0, 'Cafe, Coffee, Sandwich, Italian, Mexican, Pasta, Fast Food'),
  Text(352, 0, 'Bakery, Fast Food, Chinese, Beverages, Desserts, Sichuan,
Pizza'),
  Text(353, 0, 'Frozen Yogurt, Desserts'),
  Text(354, 0, 'Chinese, Fast Food, Pizza, Pasta, Italian'),
  Text(355, 0, 'Cafe, Coffee, Beverages, Fast Food'),
  Text(356, 0, 'Street Food, North Indian, Fast Food'),
  Text(357, 0, 'Fast Food, North Indian, Chinese'),
  Text(358, 0, 'South Indian, Street Food, Fast Food, Chinese'),
  Text(359, 0, 'Andhra, South Indian'),
  Text(360, 0, 'Desserts, Ice Cream, Fast Food, Shake, Beverages, Waffle'),
  Text(361, 0, 'North Indian, Chinese, Fast Food'),
  Text(362, 0, 'Beverages, Momos, Shake, Street Food, Fast Food'),
  Text(363, 0, 'Fast Food, Beverages, Shake'),
  Text(364, 0, 'Cafe, European, Italian, Pizza, Burger, Waffle, Tea, Coffee'),
  Text(365, 0, 'Cafe, Healthy Food, Continental, Lebanese, Italian, Beverages,
Shake, North Indian'),
  Text(366, 0, 'Bakery, Fast Food, Beverages, Ice Cream, Desserts, Shake, Pizza,
Waffle'),
  Text(367, 0, 'Cafe, Continental'),
  Text(368, 0, 'Fast Food, Tea'),
  Text(369, 0, 'Lebanese'),
  Text(370, 0, 'Mithai, Street Food, Tea'),
  Text(371, 0, 'North Indian, Continental, Chinese, Desserts, Oriental, Biryani,
Beverages'),
  Text(372, 0, 'Middle Eastern, Shawarma'),
  Text(373, 0, 'Arabian, Mandi'),
  Text(374, 0, 'Beverages, Shake, Burger'),
  Text(375, 0, 'Beverages, Shake'),
  Text(376, 0, 'Ice Cream, Desserts, Shake, Beverages'),
  Text(377, 0, 'Cafe, Fast Food, Continental, Beverages, Desserts, Bakery,
Shake, Tea'),
  Text(378, 0, 'Healthy Food, North Indian, Italian'),
  Text(379, 0, 'Healthy Food, Sandwich, Beverages'),
  Text(380, 0, 'Bakery, Shawarma'),
  Text(381, 0, 'Bakery, Desserts, Beverages'),
  Text(382, 0, 'Biryani, Kebab, Mughlai, North Indian, Desserts, Beverages'),
  Text(383, 0, 'Fast Food, Burger'),
  Text(384, 0, 'Cafe, Italian, Continental, Burger, Fast Food, Desserts,
Beverages, Pizza'),
  Text(385, 0, 'North Indian, Chinese, Beverages, Biryani, Desserts, Sichuan'),
  Text(386, 0, 'Fast Food, Desserts'),
  Text(387, 0, 'Fast Food, Street Food, Tea, Beverages, Coffee'),
  Text(388, 0, 'Ice Cream, Desserts, Pizza, Waffle, Tea, Coffee'),
  Text(389, 0, 'Fast Food, Biryani'),
  Text(390, 0, 'Juices, Beverages, Shake, Desserts, Salad'),
  Text(391, 0, 'Beverages, Fast Food, Desserts'),
```

```
Text(392, 0, 'Seafood'),
Text(393, 0, 'North Indian, Kebab'),
Text(394, 0, 'Hyderabadi, Desserts, Sandwich, Salad'),
Text(395, 0, 'Chinese, Momos')])
```



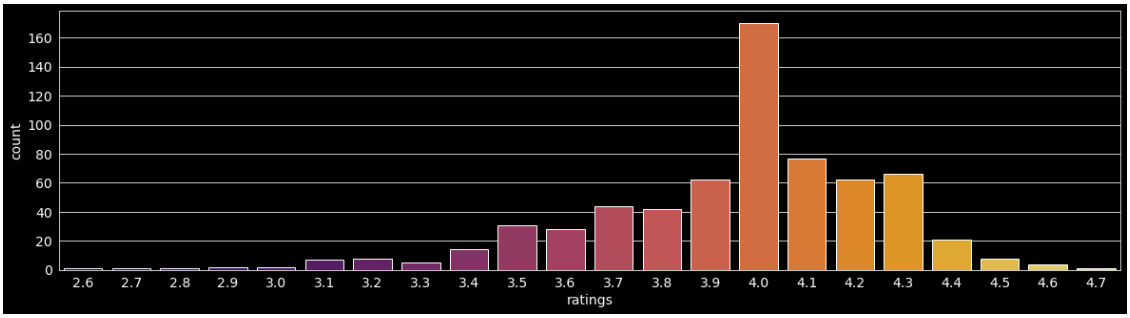
2 Ratings distribution curve

2.1 Ques.1 What is the distribution of ratings by customers across all of the restaurants listed in Zomato Hyderabad ?

[34]:
```
plt.figure(figsize=(20,5))
sns.countplot(data['ratings'],palette = 'inferno')
```

```
/opt/conda/lib/python3.9/site-packages/seaborn/_decorators.py:36: FutureWarning:
Pass the following variable as a keyword arg: x. From version 0.12, the only
valid positional argument will be `data`, and passing other arguments without an
explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

[34]: <AxesSubplot:xlabel='ratings', ylabel='count'>



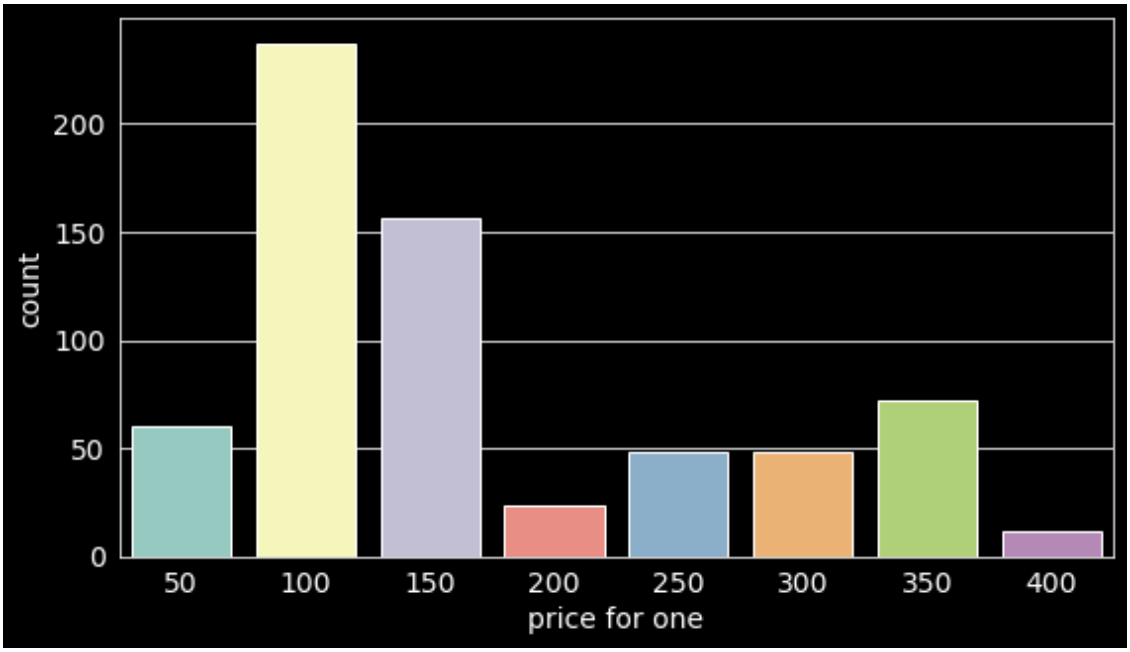
Visually we can conclude majority of rating given is from 3.5 above to 4.3 So we can infer that majority of customers are happy with the services in this city.

2.2 Ques.2 For all the orders find the expected order price for one person ?

[77]:
```
#What is Price For One
sns.countplot(data["price for one"])
```

```
/opt/conda/lib/python3.9/site-packages/seaborn/_decorators.py:36: FutureWarning:
Pass the following variable as a keyword arg: x. From version 0.12, the only
valid positional argument will be `data`, and passing other arguments without an
explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

[77]: <AxesSubplot:xlabel='price for one', ylabel='count'>



From above distribution we can easily confirm that a customer is most likely to order a cuisine in the range of 100-150 INR.

2.3 Ques3. Is there any co-relation between the ratings awarded and price spent by per person ?

[36]:
```
# Co - relation  between ratings and Price for one

corr = data.corr()
sns.heatmap(corr,xticklabels=corr.columns,yticklabels=corr.columns)
```

[36]: <AxesSubplot:>

From the above co-relation plot we can say that there is close to 0 relationship between the above two variable. And it shows that the rating awarded by customers is influenced by maybe some other factors which may be the taste of the food the satisfaction they feel, etc.

TODO - Explore one or more columns by plotting a graph below, and add some explanation about it

[37]:
```
data_1 = pd.concat([data.drop('cuisine', axis=1),
                    data.cuisine.str.split('\s*,\s*', expand=True).stack().str.
 ↪get_dummies().groupby(level=0).sum()] , axis=1)

data_1.describe(include='all').T
```

[37]:

	count	unique	top	\
names	657	636	Kwality Wall’s Frozen Dessert and Ice Cream Shop	
ratings	657.0	NaN	NaN	
price for one	657.0	NaN	NaN	
American	657.0	NaN	NaN	
Andhra	657.0	NaN	NaN	
...	...	...	...	
Street Food	657.0	NaN	NaN	
Tea	657.0	NaN	NaN	
Thai	657.0	NaN	NaN	
Waffle	657.0	NaN	NaN	
Wraps	657.0	NaN	NaN	

```
                freq        mean        std   min    25%    50%    75%    max
names              3         NaN        NaN   NaN    NaN    NaN    NaN    NaN
ratings          NaN    3.951598    0.30549   2.6    3.8    4.0    4.1    4.7
price for one    NaN  169.406393  97.178712  50.0  100.0  150.0  250.0  400.0
American         NaN    0.006088   0.077849   0.0    0.0    0.0    0.0    1.0
Andhra           NaN     0.03653   0.187747   0.0    0.0    0.0    0.0    1.0
...              ...         ...        ...   ...    ...    ...    ...    ...
Street Food      NaN    0.115677   0.320081   0.0    0.0    0.0    0.0    1.0
Tea              NaN    0.047184   0.212194   0.0    0.0    0.0    0.0    1.0
Thai             NaN    0.001522   0.039014   0.0    0.0    0.0    0.0    1.0
Waffle           NaN    0.024353    0.15426   0.0    0.0    0.0    0.0    1.0
Wraps            NaN    0.013699   0.116325   0.0    0.0    0.0    0.0    1.0

[62 rows x 11 columns]
```

[51]:
```
data_1 # creating new data set after splitting each cusinie iteam into seperate␣
↪column and filling their counts.
```

[51]:
```
                         names  ratings  price for one  American  Andhra  \
0                Sahara Bakers      3.7            100         0       0
1                          KFC      3.9            100         0       0
2          Subbaiah Gari Hotel      4.1            100         0       1
3             Paradise Biryani      3.9            100         0       0
4           Pista House Bakery      4.3            100         0       0
..                         ...      ...            ...       ...     ...
652                  Dr Cakes      3.2            350         0       0
653                Shahi Naan      4.0            350         0       0
654  Combosthalam By Phulkaas      3.8            350         0       0
655      Pachadis By Phulkaas      4.0            350         0       0
656                     Tasim      3.4            350         0       0

     Arabian  Asian  Awadhi  BBQ  Bakery  ...  Shake  Shawarma  Sichuan  \
0          0      0       0    0       1  ...      0         0        1
1          0      0       0    0       0  ...      0         0        0
2          0      0       0    0       0  ...      0         0        0
3          0      0       0    0       0  ...      0         0        0
4          0      0       0    0       0  ...      0         0        0
..       ...    ...     ...  ...     ...  ...    ...       ...      ...
652        0      0       0    0       1  ...      0         0        0
653        0      0       0    0       0  ...      0         0        0
654        0      0       0    0       0  ...      0         0        0
655        0      0       0    0       0  ...      0         0        0
656        0      0       0    0       0  ...      0         0        0

     Singaporean  South Indian  Street Food  Tea  Thai  Waffle  Wraps
0              0             0            0    0     0       0      0
```

```
1              0              0              0    0    0       0      0
2              0              1              0    0    0       0      0
3              0              0              0    0    0       0      0
4              0              0              0    0    0       0      1
..           ...            ...            ...  ...  ...     ...    ...
652            0              0              0    0    0       0      0
653            0              0              0    0    0       0      0
654            0              0              0    0    0       0      0
655            0              1              0    0    0       0      0
656            0              0              0    0    0       0      0

[657 rows x 62 columns]
```

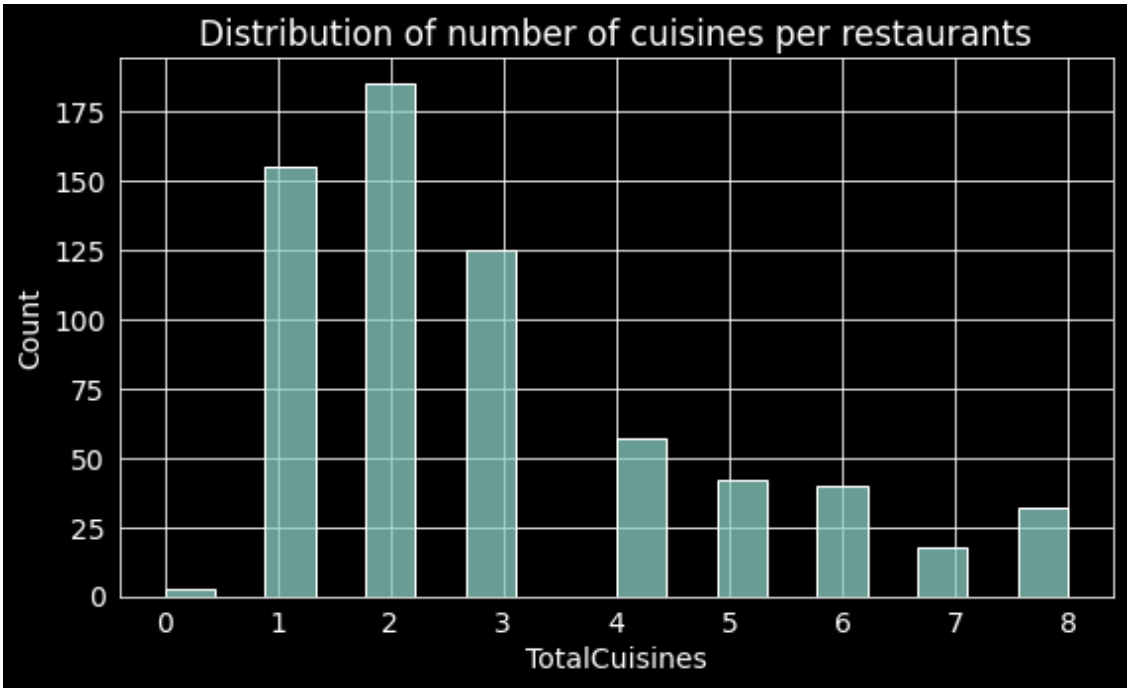
[53]:
```
data #This dataframe remains unchanged
```

[53]:
```
                        names  ratings  \
0                Sahara Bakers      3.7
1                          KFC      3.9
2          Subbaiah Gari Hotel      4.1
3             Paradise Biryani      3.9
4           Pista House Bakery      4.3
..                         ...      ...
652                  Dr Cakes      3.2
653                Shahi Naan      4.0
654  Combosthalam By Phulkaas      3.8
655      Pachadis By Phulkaas      4.0
656                      Tasim      3.4

                                              cuisine  price for one  \
0              Chinese, Bakery, Sichuan, Pizza, Burger            100
1       Burger, Fast Food, Biryani, Desserts, Beverages            100
2                         South Indian, Andhra, Mithai            100
3                 Biryani, Kebab, Desserts, Beverages            100
4    Fast Food, Sandwich, Pizza, Burger, Wraps, Rol...            100
..                                                 ...            ...
652                                Bakery, Desserts            350
653                                    North Indian            350
654                             North Indian, Chinese            350
655                                    South Indian            350
656                                  Chinese, Momos            350

     TotalCuisines
0                5
1                5
2                2
3                4
4                8
```

```
..              ...
652               2
653               1
654               2
655               1
656               2

[657 rows x 5 columns]
```

2.4 Ques4. What is the distribution of number of cuisines per restaurants in the city ?

[49]:
```
plt.title("Distribution of number of cuisines per restaurants")
data["TotalCuisines"] = data_1.iloc[:,6:].sum(axis=1)
sns.histplot(x=data["TotalCuisines"])#, shade=True)
plt.show()
```



2.4.1 Most of the restaurants have just 1,2 or 3 cuisines

2.5 Ques5. Do restaurants with more cuisines get higher ratings?

[55]:
```
plt.title("Do restaurants with more cuisines get higher ratings?")
sns.regplot(data=data[data['ratings'] != 0], x='TotalCuisines',y='ratings')#,␣
↪shade=True)
plt.show()
```

[56]:
```
# No correlation between No. of cuisines and ratings.
#  So the answer to above question is no
```

[57]:
```
import jovian
```

[58]:
```
jovian.commit()
```

```
<IPython.core.display.Javascript object>
```

```
[jovian] Updating notebook "shubhammeena712/zomato-restaurant-analysis" on
https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/shubhammeena712/zomato-
restaurant-analysis
```

[58]:
```
'https://jovian.ai/shubhammeena712/zomato-restaurant-analysis'
```

3 Inferences and Conclusion

Based upon the used data set we made following Inferences 1. What are the top 10 highest rated restaurant in Hyderabad ? 2. Top 10 most popular Cuisines in the city ? 3. What is the average rating of restaurants in the city listed on zomato ? 4. Do customers rating get affected by the cost of order? 5. How much cuisines a restaurant is likely to list ? 6. Can we predict the range of expected order cost for any customer ?

3.1 1. Whart are the top 10 highest rated restaurant in Hyderabad ?

[60]:
```
rating_df = data.sort_values('ratings', ascending=False).head(10)
rating_df
```

[60]:
```
                        names  ratings              cuisine  price for one  \
450         Sri Krishna Sweets      4.7               Mithai            100
184                   Euphoria      4.6    Desserts, Bakery            350
611         Mimee's Chocolates      4.6             Desserts            150
72                 Almond House      4.6  Mithai, Street Food            350
496                US Live Pops      4.6            Fast Food            150
214              Madhur Sweets      4.5    Mithai, Desserts            100
173        Emerald Mithai Shop      4.5               Mithai             50
559   Sri Shagun Mithai Vatika      4.5               Mithai            150
150  Abhiruchi Swaghruha Foods      4.5               Mithai            150
22     NIC - Natural Ice Creams      4.5  Ice Cream, Desserts            150

     TotalCuisines
450              1
184              2
611              1
72               2
496              1
214              2
173              1
559              1
150              1
22               2
```

3.2 2. Top 10 most popular Cuisines in the city ?

[82]:
```
n=10
data["cuisine"].value_counts()
```

[82]:
```
South Indian
67
Bakery, Desserts
23
Mithai, Street Food
22
Mithai
14
North Indian, Chinese
13
     ..
Chinese, North Indian, Sichuan
1
```

```
Mithai, Desserts, Street Food
1
Biryani, Cafe, North Indian, Continental, Street Food, Beverages, Desserts,
Pizza     1
South Indian, Street Food, Juices, Beverages
1
Chinese, Momos
1
Name: cuisine, Length: 396, dtype: int64
```

3.3 3. What is the average rating of restaurants in the city listed on zomato ?

[73]:
```
rating = data['ratings'].mean()

print(" %.2f" % rating) #Average rating(up to 2 decimal places)
```

```
 3.95
```

3.4 4. Do customers rating get affected by the cost of order?

3.5 No,

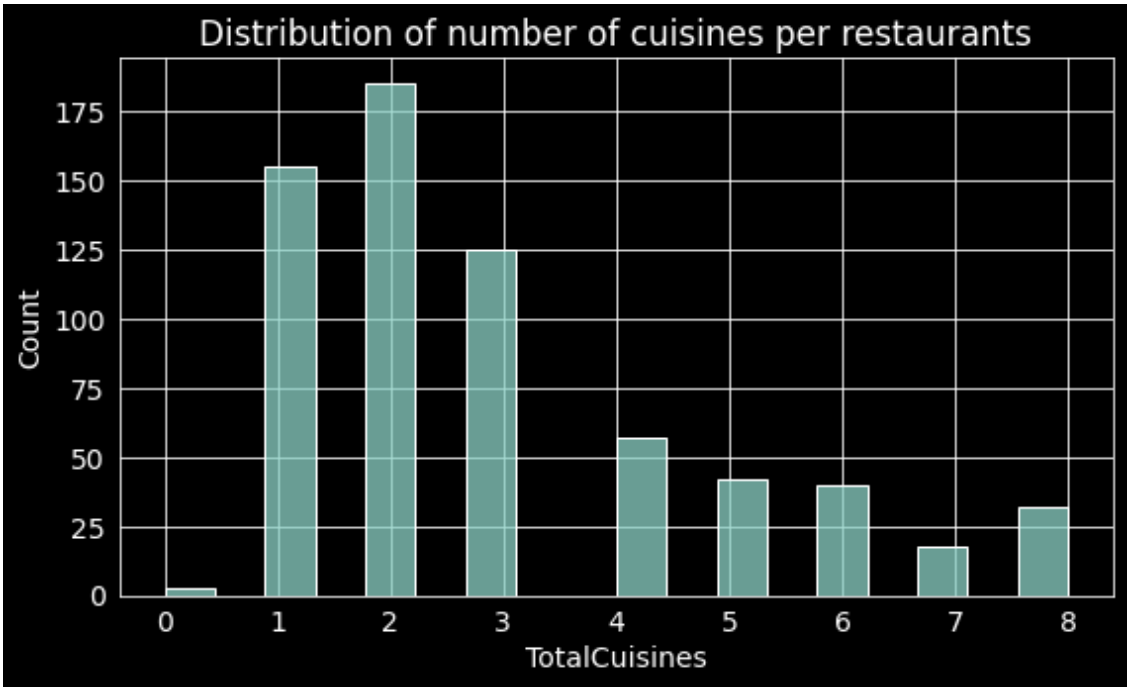
[75]:
```
corr = data.corr()
sns.heatmap(corr,xticklabels=corr.columns,yticklabels=corr.columns)
```

[75]: <AxesSubplot:>

5. How much cuisines a restaurant is likely to list ?

3.6 Most of the restaurants have just 1,2 or 3 cuisines

[76]:
```
plt.title("Distribution of number of cuisines per restaurants")
data["TotalCuisines"] = data_1.iloc[:,6:].sum(axis=1)
sns.histplot(x=data["TotalCuisines"])#, shade=True)
plt.show()
```



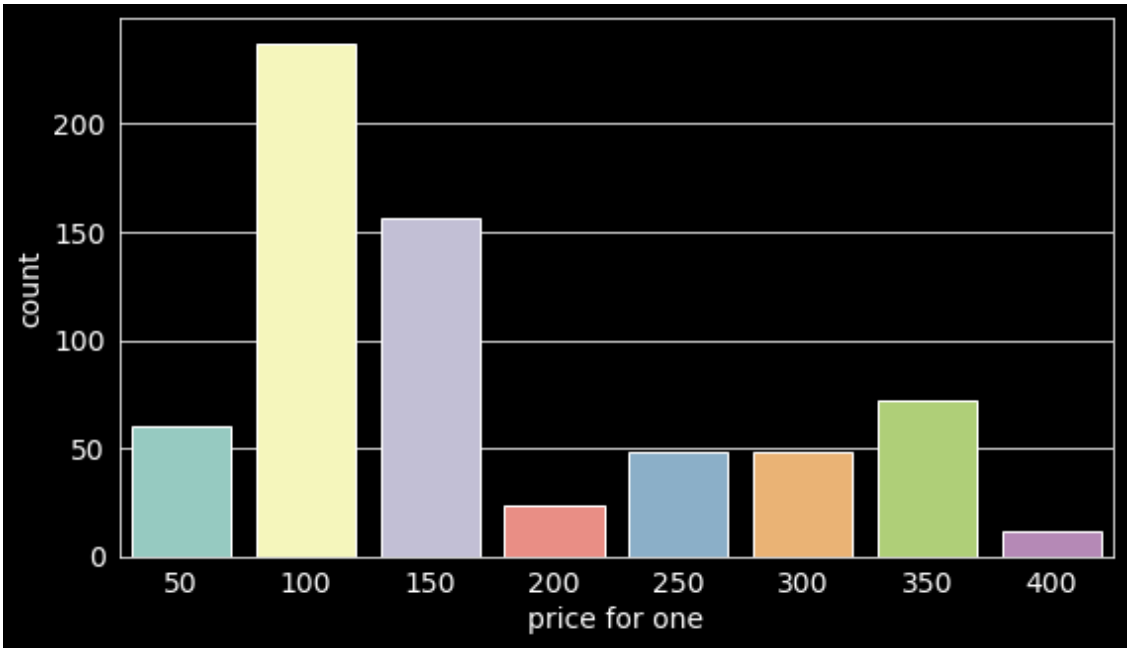
3.7 6. Can we predict the range of expected order cost for any customer ?

3.8 Yes, a customer is most likely to order a cuisine in the range of 100-150 INR

[78]:
```
sns.countplot(data["price for one"])
```

```
/opt/conda/lib/python3.9/site-packages/seaborn/_decorators.py:36: FutureWarning:
Pass the following variable as a keyword arg: x. From version 0.12, the only
valid positional argument will be `data`, and passing other arguments without an
explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

[78]:
```
<AxesSubplot:xlabel='price for one', ylabel='count'>
```



[45]:
```
import jovian
```

[46]:
```
jovian.commit()
```

```
<IPython.core.display.Javascript object>
```

```
[jovian] Updating notebook "shubhammeena712/zomato-restaurant-analysis" on
https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/shubhammeena712/zomato-
restaurant-analysis
```

[46]:
```
'https://jovian.ai/shubhammeena712/zomato-restaurant-analysis'
```

4 References

4.0.1 1. https://pandas.pydata.org/docs/reference/api/pandas.io.formats.style.Styler.set_properties.html

4.0.2 2.https://github.com/JovianML/opendatasets/blob/master/README.md#kaggle-credentials

[79]:
```
import jovian
```

[80]:
```
jovian.commit()
```

```
<IPython.core.display.Javascript object>
```

```
[jovian] Updating notebook "shubhammeena712/zomato-restaurant-analysis" on
https://jovian.ai
```

```
[jovian] Committed successfully! https://jovian.ai/shubhammeena712/zomato-
restaurant-analysis
```

[80]:
```
'https://jovian.ai/shubhammeena712/zomato-restaurant-analysis'
```

[ ]:
```

```