Data Analytics

Trip Advisor New-York City restaurants Dataset

Cho-Drugeon Hye-Jin

June, 2022

Table of content

1. Introduction

Trip Advisor -Restaurant data
Can it be possible to trust even if no answers for popular food are huge?

2. Data and data sources

Trip Advisor Newyork City restaurants Dataset 10k+
Newyork City Dataset from Tripadvisor

My Data choice is https://www.kaggle.com/datasets/rayhan32/trip-advisor-newyork-city-restaurants-dataset-10k?resource=download

No answers for popular food are huge (74 percent)

3. Data collection

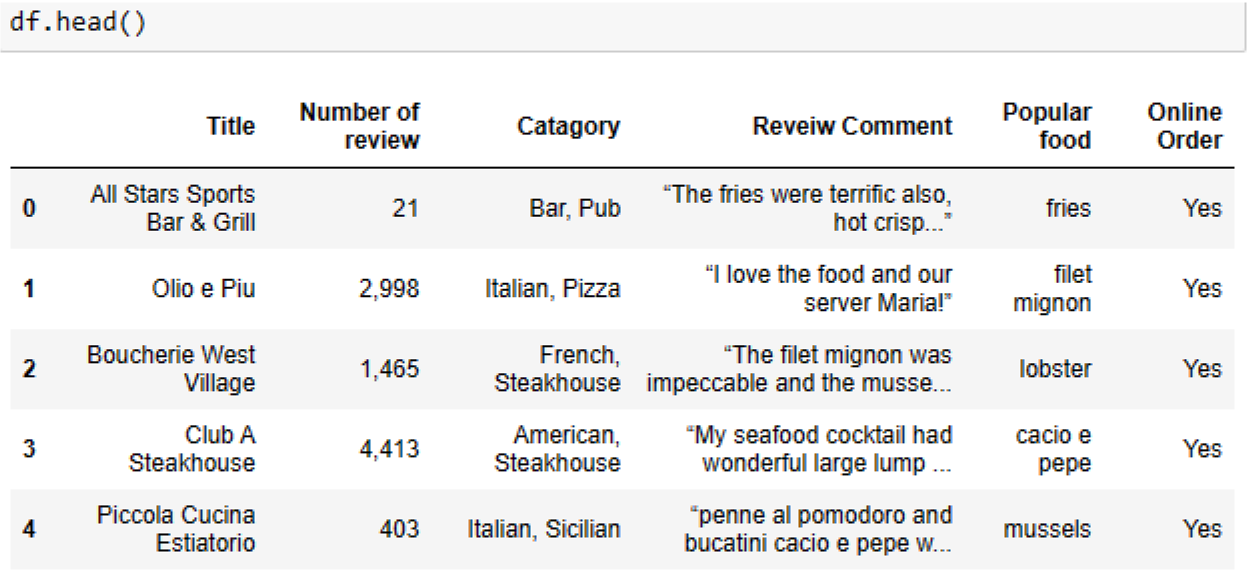
Expected update frequency
Monthly

over 10,000 records of restaurant reviews in New York

7237 unique values

4. Data cleaning and Exploratory data analysis

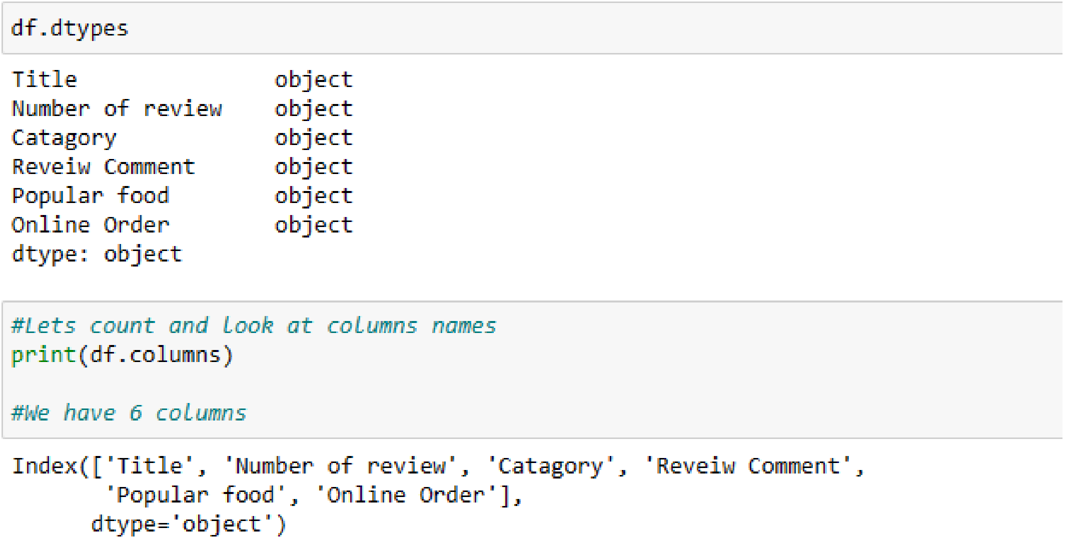
1. **Data Summary: The DataFrame is defined at the initial stage as df and the summary provides essential information of variables.**



```
df.head()
```

	Title	Number of review	Catagory	Reveiw Comment	Popular food	Online Order
0	All Stars Sports Bar & Grill	21	Bar, Pub	“The fries were terrific also, hot crisp...”	fries	Yes
1	Olio e Piu	2,998	Italian, Pizza	“I love the food and our server Maria!”	filet mignon	Yes
2	Boucherie West Village	1,465	French, Steakhouse	“The filet mignon was impeccable and the musse...	lobster	Yes
3	Club A Steakhouse	4,413	American, Steakhouse	“My seafood cocktail had wonderful large lump ...	cacio e pepe	Yes
4	Piccola Cucina Estiatorio	403	Italian, Sicilian	“penne al pomodoro and bucatini cacio e pepe w...	mussels	Yes

2. Data Summary: The DataFrame has variables, only as “object.”



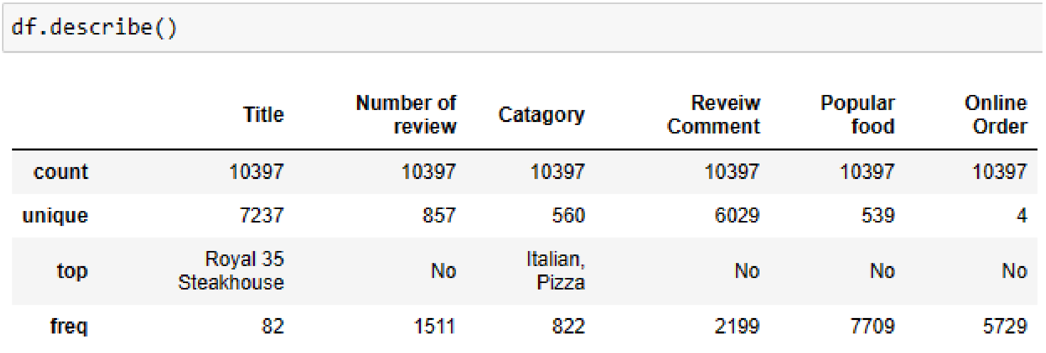
```
df.dtypes
```

```
Title                object
Number of review     object
Catagory             object
Reveiw Comment       object
Popular food         object
Online Order         object
dtype: object
```

```
#Lets count and look at columns names
print(df.columns)

#We have 6 columns
```

```
Index(['Title', 'Number of review', 'Catagory', 'Reveiw Comment',
       'Popular food', 'Online Order'],
      dtype='object')
```

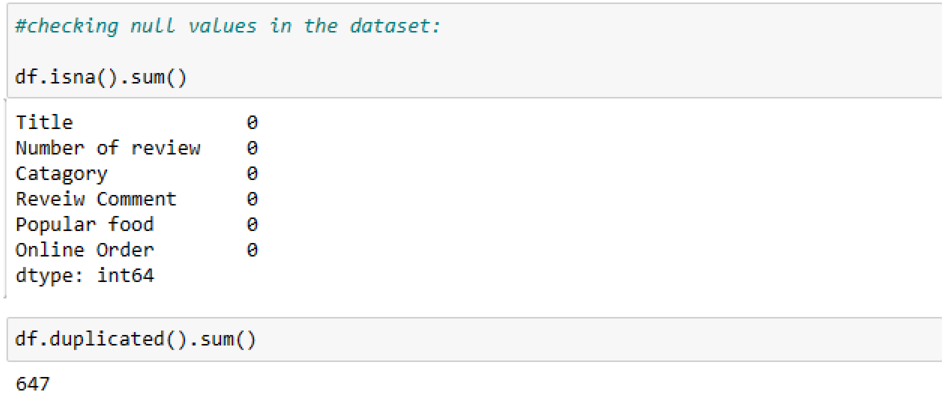
3.Data with no null variables: Every column has same number as of 10397 variables and unique variables are varied.



```
df.describe()
```

	Title	Number of review	Catagory	Reveiw Comment	Popular food	Online Order
count	10397	10397	10397	10397	10397	10397
unique	7237	857	560	6029	539	4
top	Royal 35 Steakhouse	No	Italian, Pizza	No	No	No
freq	82	1511	822	2199	7709	5729

4.Data with duplicated variables



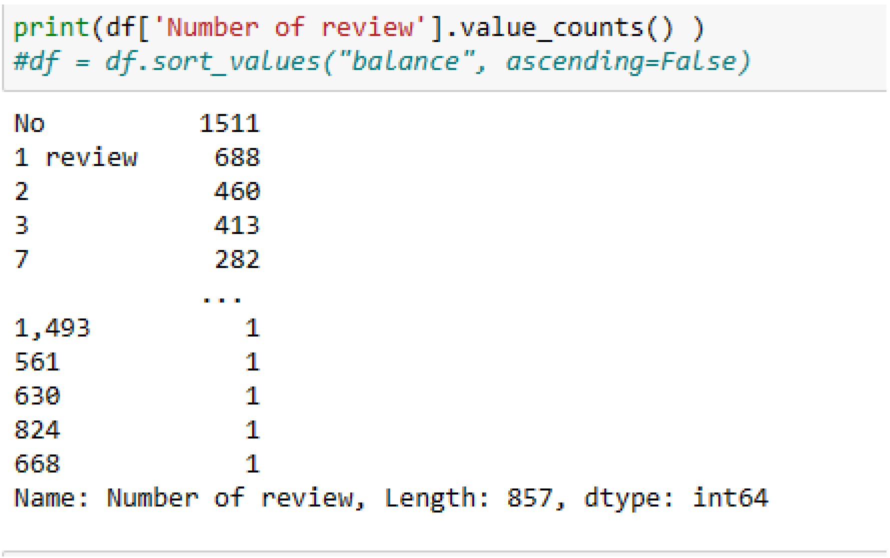
```
#checking null values in the dataset:

df.isna().sum()
```

```
Title              0
Number of review   0
Catagory           0
Reveiw Comment     0
Popular food       0
Online Order       0
dtype: int64
```

```
df.duplicated().sum()
```

```
647
```

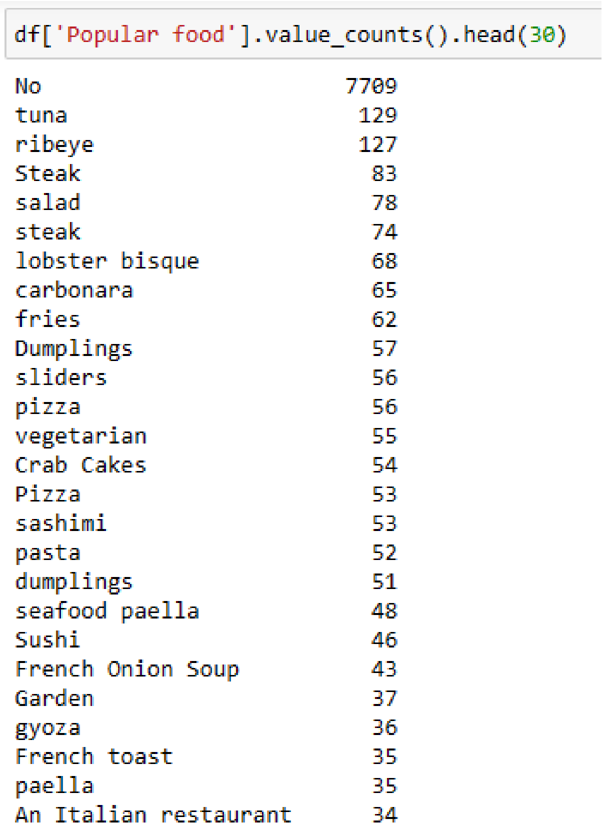
5.Data with huge “no” answers



```
print(df['Number of review'].value_counts() )
#df = df.sort_values("balance", ascending=False)
```

```
No          1511
1 review     688
2            460
3            413
7            282
            ... 
1,493          1
561            1
630            1
824            1
668            1
Name: Number of review, Length: 857, dtype: int64
```

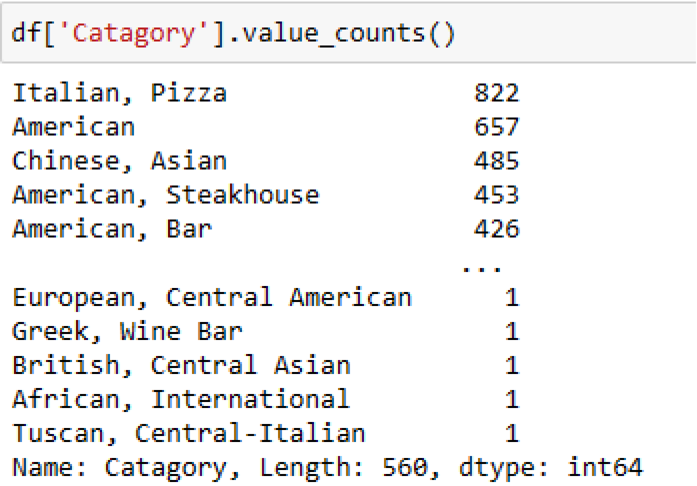
6.Data deficiency, in particular with “Popular Food”



```
df['Popular food'].value_counts().head(30)
```

```
No                         7709
tuna                        129
ribeye                      127
Steak                        83
salad                        78
steak                        74
lobster bisque               68
carbonara                    65
fries                        62
Dumplings                    57
sliders                      56
pizza                        56
vegetarian                   55
Crab Cakes                   54
Pizza                        53
sashimi                      53
pasta                        52
dumplings                    51
seafood paella               48
Sushi                        46
French Onion Soup            43
Garden                       37
gyoza                        36
French toast                 35
paella                       35
An Italian restaurant        34
```

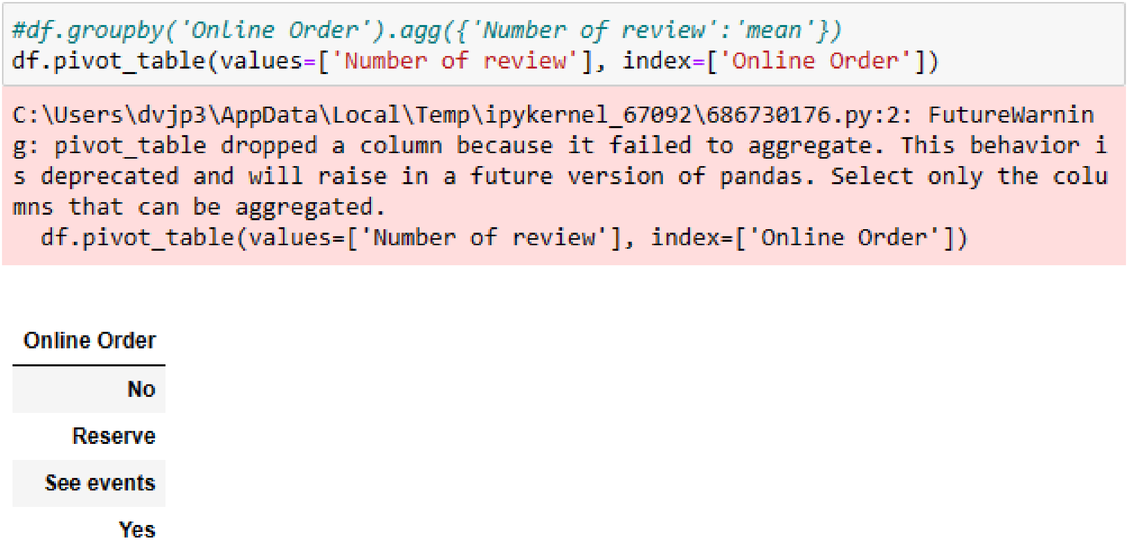
7. Data with Category such as “Italian, Pizza”, “American”…..



```
df['Catagory'].value_counts()
```

```
Italian, Pizza                822
American                      657
Chinese, Asian                485
American, Steakhouse          453
American, Bar                 426
                             ...
European, Central American      1
Greek, Wine Bar                 1
British, Central Asian          1
African, International          1
Tuscan, Central-Italian         1
Name: Catagory, Length: 560, dtype: int64
```

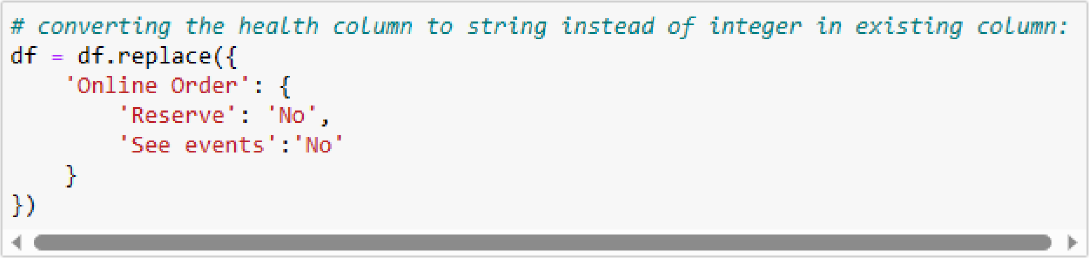
8. Data with answers, non-categorized



```
#df.groupby('Online Order').agg({'Number of review':'mean'})
df.pivot_table(values=['Number of review'], index=['Online Order'])
```

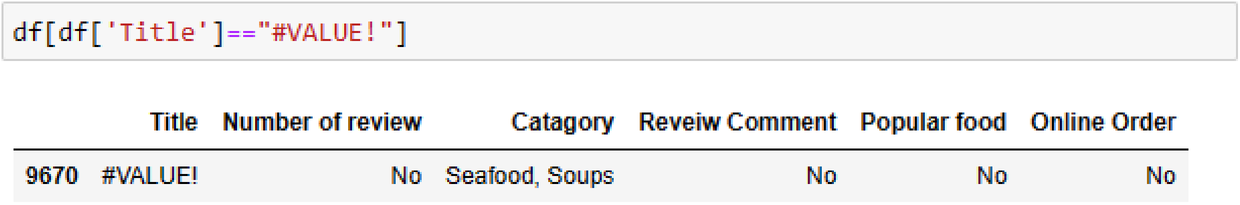
```
C:\Users\dvjp3\AppData\Local\Temp\ipykernel_67092\686730176.py:2: FutureWarnin
g: pivot_table dropped a column because it failed to aggregate. This behavior i
s deprecated and will raise in a future version of pandas. Select only the colu
mns that can be aggregated.
  df.pivot_table(values=['Number of review'], index=['Online Order'])
```

Online Order
No
Reserve
See events
Yes



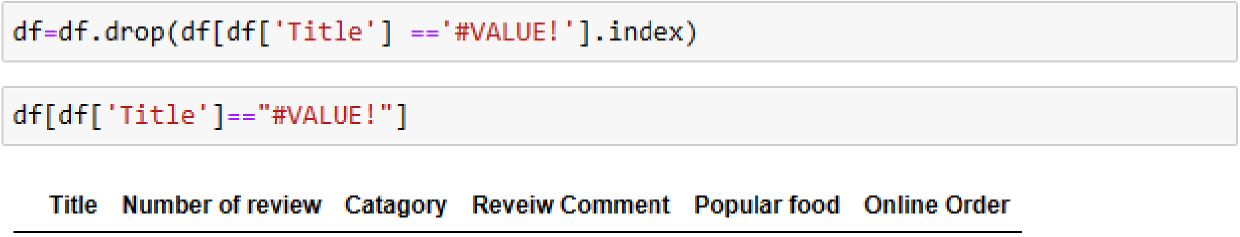
```
# converting the health column to string instead of integer in existing column:
df = df.replace({
    'Online Order': {
        'Reserve': 'No',
        'See events':'No'
    }
})
```

9. A variable with “Non-Values”



```
df[df['Title']=="#VALUE!"]
```

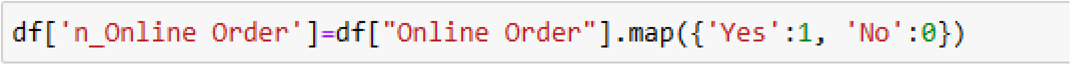
	Title	Number of review	Catagory	Reveiw Comment	Popular food	Online Order
9670	#VALUE!	No	Seafood, Soups	No	No	No



```
df=df.drop(df[df['Title'] =='#VALUE!'].index)
```

```
df[df['Title']=="#VALUE!"]
```

Title	Number of review	Catagory	Reveiw Comment	Popular food	Online Order



```
df['n_Online Order']=df["Online Order"].map({'Yes':1, 'No':0})
```