

Project: APIs (Zomato API)

Necessary Python code for further problems:

Required libraries:

```
import requests
import json
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import operator
```

Creating data frames:

```
df = pd.read_csv(r"C:\Users\Ayush Deval\Downloads\zomato.csv", encoding
= 'ISO-8859-1')
df1 = df[df['Country Code'] == 1]
```

Problem Statement – 1

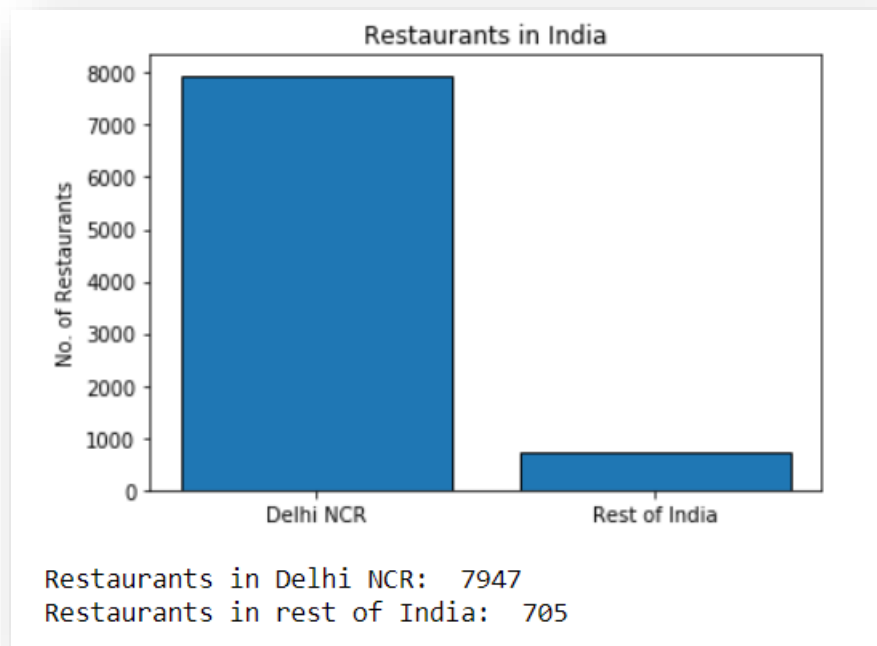
The dataset is highly skewed toward the cities included in Delhi-NCR. So, we will summarise all the other cities in Rest of India while those in New Delhi, Ghaziabad, Noida, Gurgaon, Faridabad to Delhi-NCR. Doing this would make our analysis turn toward Delhi-NCR v Rest of India.

1. Plot the bar graph of number of restaurants present in Delhi NCR vs Rest of India.

Sol. As per the dataset provided, I have plotted a graph based on the number of restaurants present in Delhi NCR and rest of India.

Below the code used for plotting the required graph:

```
a = 0
b = 0
for i in df1.index:
    if df1['City'][i] in ['New Delhi', 'Noida', 'Ghaziabad', 'Gurgaon',
        'Faridabad']:
        a += 1
    else:
        b += 1
x = [a, b]
y = ['Delhi NCR', 'Rest of India']
plt.bar(y, x, edgecolor = "black")
plt.ylabel('No. of Restaurants')
plt.title('Restaurants in India')
plt.show()
print('Restaurants in Delhi NCR: ', a)
print('Restaurants in rest of India: ', b)
```



As we can see in the graph, we can tell that the no. of restaurants in Delhi NCR are very much higher in number than in Rest of India.

2. Find the cuisines which are not present in restaurant of Delhi NCR but present in rest of India. Check using Zomato API whether this cuisines are actually not served in restaurants of Delhi-NCR or just it due to incomplete dataset.

Sol. In different parts of India, people of different states eat different cuisines. In this problem, we need to find out that what are the cuisines in rest of India that are not served in Delhi NCR. To find this out, I have used a code to filter out such cuisines.

```
a = set()
b = set()
for i in df1.index:
    temp = df1['Cuisines'][i].split(',')
    if df1['City'][i] in ['New Delhi', 'Noida', 'Ghaziabad',
                        'Gurgaon', 'Faridabad']:
        for j in temp:
            a.add(j.strip())
    else:
        for j in temp:
            b.add(j.strip())
c = b-a
print('Cuisines that are not available in Delhi NCR...')
j = 1
for i in c:
    print("%d. " %j, i)
    j += 1
```

Using this code, we filtered out the cuisines which are not served in Delhi NCR restaurants. But, to confirm this, I used Zomato API to find out if the details provided by this dataset is actually correct. Below is the code I used to confirm the result from the dataset.

```

r = requests.get('https://developers.zomato.com/api/v2.1/cuisines',
headers = {'user-key':'00ea204558b9356ba4573dc4092970bd'}, params =
{'city_id':1})

data = r.json()
data = data['cuisines']
x = set()
for i in data:
    x.add(i['cuisine']['cuisine_name'])
print("Above mentioned cuisines are not actually served in Delhi NCR:
",c not in x)

```

As per the results from above two codes, there are few cuisines that are not served in Delhi NCR which are given below in the form of outputs we received from the above codes.

```

Cuisines that are not available in Delhi NCR restaurants...
1. German
2. Malwani
3. Cajun
4. BBQ
Above mentioned cuisines are not actually served in Delhi NCR: True

```

3. Find the top 10 cuisines served by maximum number of restaurants in Delhi NCR and rest of India.

Sol. Now, as per the problem given, we need to find out the top 10 cuisines in Delhi NCR and Rest of India. It is really important for the people in cooking business to know the type of cuisines that are most liked by the people and are served in majority of food stalls or restaurants. Below is the code used to find out the top 10 cuisines in India:

```

count1 = dict()
count2 = dict()
count = dict()

for i in df1.index:
    temp = df1['Cuisines'][i].split(',')
    if df1['City'][i] in ['New Delhi', 'Noida', 'Ghaziabad', 'Gurgaon', 'Faridabad']:
        for j in temp:
            if j.strip() in count1:
                count1[j.strip()] += 1
            else:
                count1[j.strip()] = 1
    else:
        for j in temp:
            if j.strip() in count2:
                count2[j.strip()] += 1
            else:
                count2[j.strip()] = 1
        for j in temp:

```

```
if j.strip() in count:
    count[j.strip()] += 1
else:
    count[j.strip()] = 1
```

```
count = dict(sorted(count.items(), key=operator.itemgetter(1),reverse=True))
count1 = dict(sorted(count1.items(), key=operator.itemgetter(1),reverse=True))
count2 = dict(sorted(count2.items(), key=operator.itemgetter(1),reverse=True))
```

```
print('Top 10 cuisines in Delhi NCR...')
a = 0
for key in count1:
    if a == 10:
        break
    print("%d. " %(a+1), key, count1[key])
    a += 1
```

```
print()
print('Top 10 cuisines in rest of India...')
a = 0
for key in count2:
    if a == 10:
        break
    print("%d. " %(a+1), key, count2[key])
    a += 1
```

```
print()
print('Top 10 cuisines in whole India...')
a = 0
for key in count:
    if a == 10:
        break
    print("%d. " %(a+1), key, count[key])
    a += 1
```

Using this code, we found out some interesting facts:

- North Indian and Chinese food is something that is liked by both the categories i.e., Delhi NCR and Rest of India.
- People of Delhi NCR like Bakery and Street Food while rest of India don't like cuisines that much.
- Same as above, people of rest of India are more interested in Mexican and Cafes while this is not the case in Delhi NCR.
- We also found that out of 10, 8 are the common cuisines that are served in both regions.

Below are the top 10 cuisines in Delhi NCR and rest of India:

Top 10 cuisines in Delhi NCR...

1. North Indian 3597
2. Chinese 2448
3. Fast Food 1866
4. Mughlai 933
5. Bakery 697
6. South Indian 569
7. Continental 547
8. Desserts 542
9. Street Food 538
10. Italian 535

Top 10 cuisines in rest of India...

1. North Indian 349
2. Chinese 242
3. Continental 177
4. Italian 147
5. Cafe 136
6. Fast Food 97
7. South Indian 62
8. Mughlai 59
9. Desserts 55
10. Mexican 50

Top 10 cuisines in whole India...

1. North Indian 3946
2. Chinese 2690
3. Fast Food 1963
4. Mughlai 992
5. Bakery 726
6. Continental 724
7. Italian 682
8. South Indian 631
9. Cafe 627
10. Desserts 597

4. Write a short detailed analysis of how cuisine served is different from Delhi NCR to Rest of India. Plot the suitable graph to explain your inference.

Sol. I have done a small analysis in how cuisines served in India. What I did was that I took top 8 cuisines which are common in both the regions and tried to find out that what percentage of cuisines are being served in Delhi NCR and Rest of India so that I can tell the preference of different cuisines that restaurants like to serve.

Below is the code I used:

```
x = [3597, 2448, 1866, 535, 933, 569, 547, 542]
```

```
y = [349, 242, 97, 147, 59, 62, 177, 55]
```

here we are taking 5 cuisines into account which are among the top 10 cuisines of both the regions.

Those cuisines are:

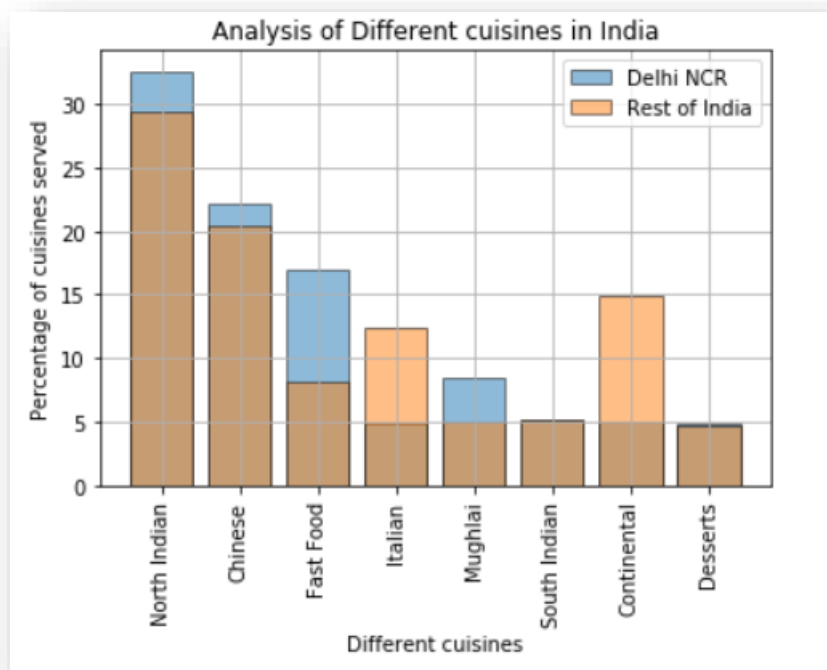
```
# North Indian
# Chinese
# Fast Food
# Italian
# Mughlai
# South Indian
# Continental
# Desserts
```

```
sum1 = sum(x)
sum2 = sum(y)
```

```
for i in range(8):
    x[i] = x[i]*100/sum1
    y[i] = y[i]*100/sum2
```

```
name = ["North Indian", 'Chinese', 'Fast Food', 'Italian', 'Mughlai', 'South Indian',
'Continental', 'Desserts']
plt.bar(name, x, edgecolor = "black", alpha = 0.5, label = 'Delhi NCR')
plt.bar(name, y, edgecolor = "black", alpha = 0.5, label = 'Rest of India')
plt.legend()
plt.xticks(name, rotation = 90)
plt.grid()
plt.title('Analysis of Different cuisines in India')
plt.show()
```

Below is the bar graph that shows the preference of different cuisines in different parts of India i.e., Delhi NCR and Rest of India.



As we can see in the graph,

- More than 30% restaurants in Delhi NCR serve North Indian while in rest of India it is less 30%. But the fact the almost similar. Same is the case with Chinese cuisine but they are quite less served in Delhi NCR and rest of India i.e., about 20-22% restaurants serve Chinese food.
- Fast food is quite popular in Delhi NCR. One side it is served in more than 15% restaurants in Delhi NCR on the other side it is served in less than 10% restaurants in rest of India.
- Similar case like above comes with Mughlai cuisine. It is being served in about 8% restaurants in Delhi NCR and in rest of India, it is less than 5%.
- Italian and Continental are quite popular in rest of India than in Delhi NCR. About 25% restaurants in Rest of India serve Italian and Continental while the numbers in Delhi NCR are comparably less i.e., about 10%.
- South Indian and Desserts are something that are equally served in Delhi NCR and rest of India.

Problem Statement – 2

User Rating of a restaurant plays a crucial role in selecting a restaurant or ordering the food from the restaurant.

- 1. Write a short detail analysis of how the rating is affected by restaurant due following features: Plot a suitable graph to explain your inference.**

Sol. To answer this, I have analysed the data of New Delhi.

a) Number of Votes given Restaurant

Sol. To find out the answer of this question, I have analysed the effect the higher and lower number of votes given to restaurants. Below is the code I used:

```

votes = dict()
rating = dict()
for i in df1.index:
    if df1['City'][i] == 'New Delhi':
        key = df1['Restaurant Name'][i]
        votes[key] = df1['Votes'][i]
        rating[key] = df1['Aggregate rating'][i]

a = 0
b = 0
c = 0
d = 0
for i in votes1:
    if votes1[i] > 1000:
        if rating[i] >= 4:
            a += 1
        else:
            b += 1
    if votes1[i] < 500 and votes1[i] > 100:
        if rating[i] >= 4:
            c += 1

```

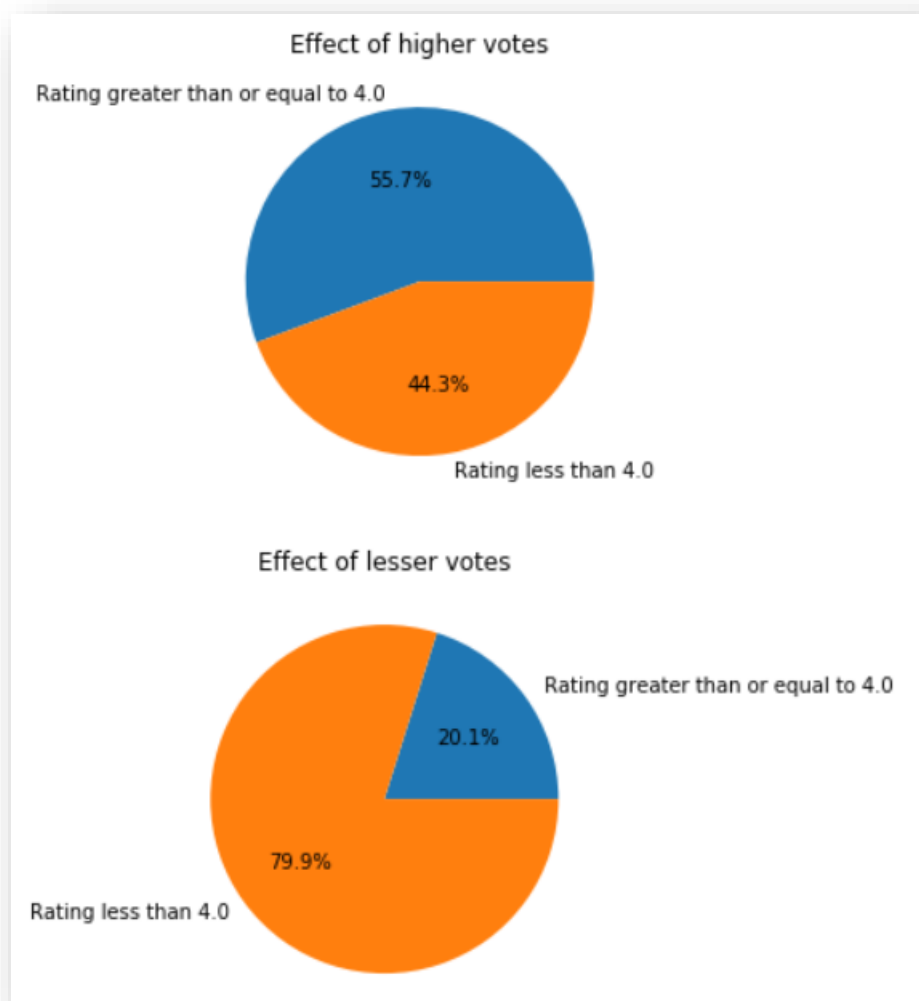
```

        else:
            d += 1
x = [a, b]
y = [c, d]

plt.pie(x, labels = [' Rating greater than or equal to 4.0',
'Rating less than 4.0'], autopct = '%.1f%%')
plt.title('Effect of higher votes')
plt.show()
plt.pie(y, labels = [' Rating greater than or equal to 4.0',
'Rating less than 4.0'], autopct = '%.1f%%')
plt.title('Effect of lesser votes')
plt.show()

```

Below is the pie chart of the analysis:



As we can see in the above charts, restaurants with higher votes have better rating than the restaurants with lesser votes. And it is obvious. Even in our personal life, we prefer the restaurant with more no. of votes and better rating and most of the time the food is good enough to give a good rating. While on the other hand we don't trust the restaurants with less votes easily as they might be fake votes to bring the customers.

b) Restaurant serving more number of cuisines

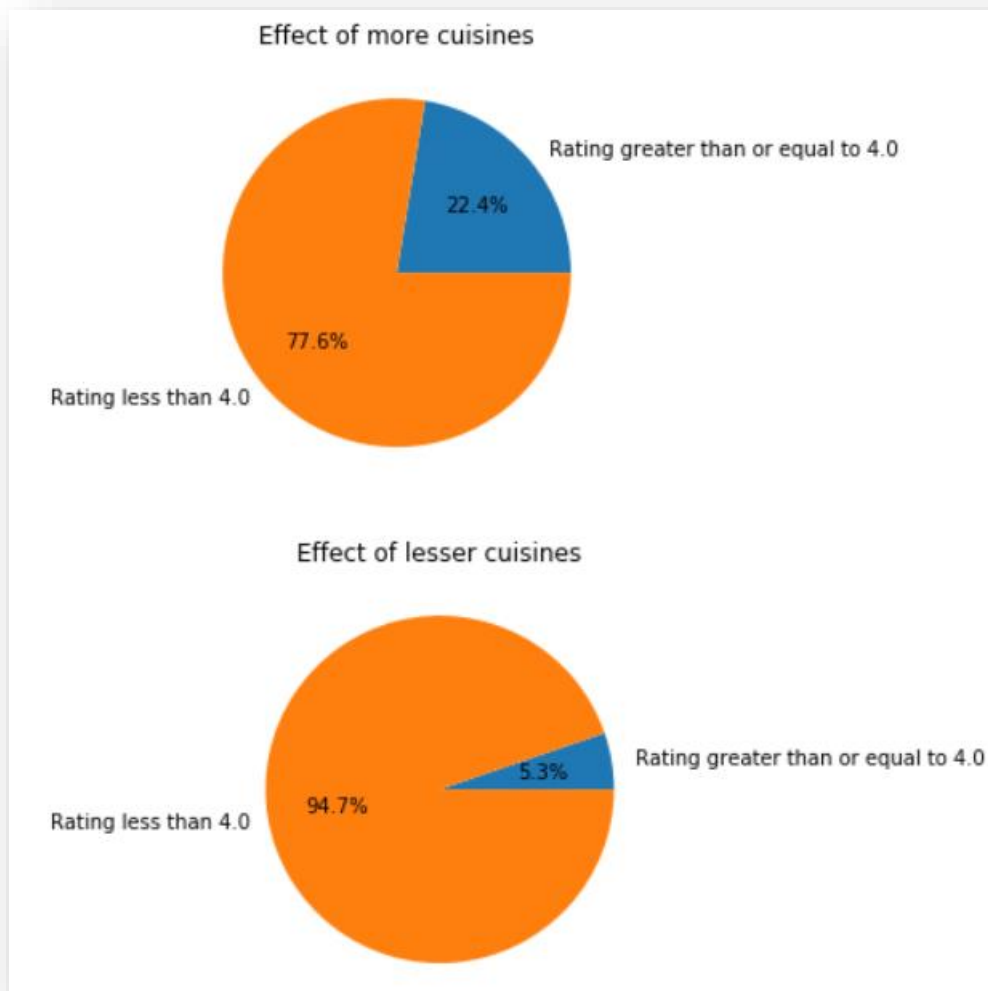
Sol. Personally, I prefer the restaurant with more number of cuisines so that I can choose the variety of food for me at a same place. But to find the effect of no. of cuisines on rating, I used the code given below:

```
cuisines = dict()
rating = dict()
for i in df1.index:
    temp = df1['Cuisines'][i].split(',')
    if df1['City'][i] == 'New Delhi':
        cuisines[df1['Restaurant Name'][i]] = len(temp)
        rating[df1['Restaurant Name'][i]] = df1['Aggregate rating'][i]
```

```
a = 0
b = 0
c = 0
d = 0
for i in cuisines:
    if cuisines[i] >= 4 :
        if rating[i] >= 4:
            a += 1
        else:
            b += 1
    if cuisines[i] < 4 and cuisines[i] > 0:
        if rating[i] >= 4:
            c += 1
        else:
            d += 1
```

```
plt.pie([a, b], labels = [' Rating greater than or equal to 4.0', 'Rating less than 4.0'],
autopct = '%.1f%%')
plt.title('Effect of more cuisines')
plt.show()
plt.pie([c, d], labels = [' Rating greater than or equal to 4.0', 'Rating less than 4.0'],
autopct = '%.1f%%')
plt.title('Effect of lesser cuisines')
plt.show()
```

Lets see what result we get:



As we can see in the above result, people tend to prefer the restaurants with more number of cuisines. One side there are only 5.3% 4+ rating while on the other side, it is 22.4% 4+ rating. This analysis says it all. Rating is directly proportional to no. of cuisines.

c) Average Cost of Restaurant

Sol. When it comes to average price, every restaurant try to give their best price to pitch the customer or to show them that they are good enough to provide so and so food at so and so price. As we all notice in our day to day life that most of the time cheap food is unhygenic and expensive food is well treated with hygiene.

Taste is not the only factor that can get food a good rating but hygiene is also an important thing. So, as we all know that big and famous restaurants provide food at higher price but we should also know that they also try to keep the food as tastier and hygenic as their price. To know the actual stats, I used a code that can tell me how the average price is an important factor for the restaurant's rating.

```
cost = dict()
rating = dict()
for i in df1.index:
    if df1['City'][i] == 'New Delhi':
```

```

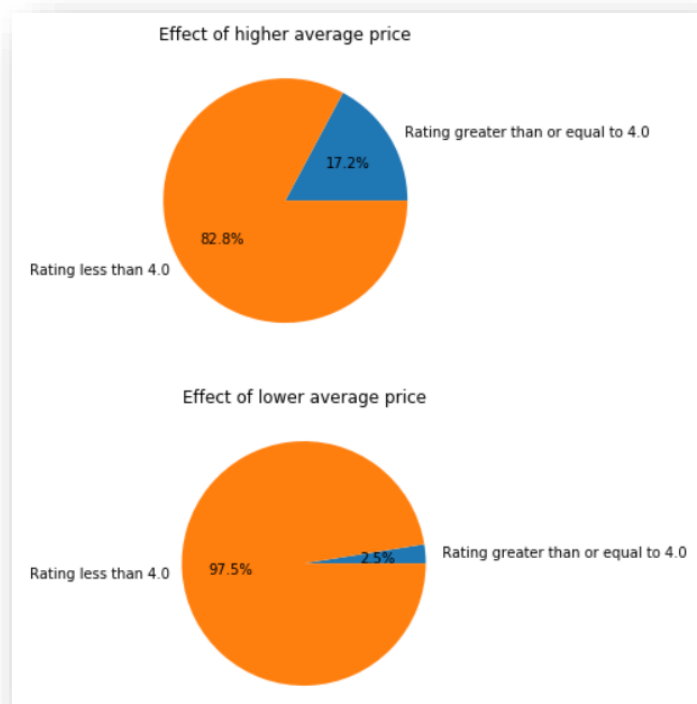
cost[df1['Restaurant Name'][i]] = df1['Average Cost for
two'] [i]
rating[df1['Restaurant Name'][i]] = df1['Aggregate
rating'] [i]

a = 0
b = 0
c = 0
d = 0
for i in cost:
    if cost[i] >= 1000:
        if rating[i] > 4:
            a += 1
        else:
            b += 1
    if cost[i] <= 200 and cost[i] > 0:
        if rating[i] >= 4:
            c += 1
        else:
            d += 1

plt.pie([a, b], labels = [' Rating greater than or equal to 4.0',
'Rating less than 4.0'], autopct = '%.1f%%')
plt.title('Effect of higher average price')
plt.show()
plt.pie([c, d], labels = [' Rating greater than or equal to 4.0',
'Rating less than 4.0'], autopct = '%.1f%%')
plt.title('Effect of lower average price')
plt.show()

```

Below are the pie charts to show the stats on the basis of price and rating:



As we can see from the above charts, restaurants with higher average price have more 4+ rating and the difference is no less. So, in some ways we can say that rating is directly proportional to average price and average price is directly proportional to taste and hygiene.

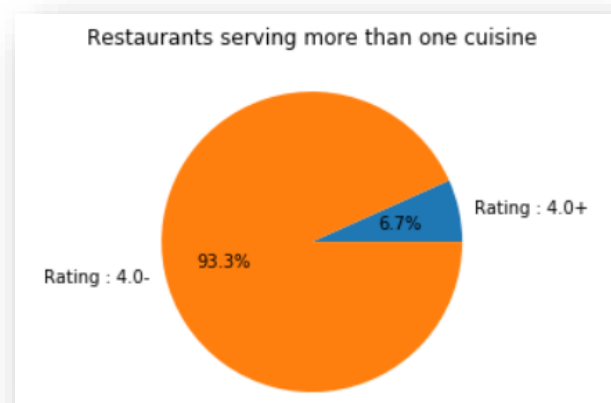
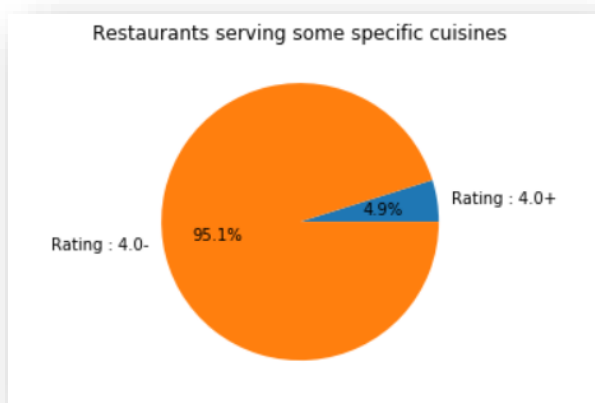
d) Restaurant serving some specific cuisines

Sol. To find this out, I thought that why not create a code that can tell me that how rating depends on serving some specific cuisine.

```
cuisines = dict()
rating = dict()
for i in df1.index:
    temp = df1['Cuisines'][i].split(',')
    if df1['City'][i] == 'New Delhi':
        if len(temp) == 1:
            if df1['Aggregate rating'][i] >= 4:
                a += 1
            else:
                b += 1
        else:
            if df1['Aggregate rating'][i] >= 4:
                c += 1
            else:
                d += 1

plt.pie([a, b], labels = ['Rating : 4.0+', 'Rating : 4.0-'],
        autopct = '%.1f%%')
plt.title('Restaurants serving some specific cuisines')
plt.show()
plt.pie([c, d], labels = ['Rating : 4.0+', 'Rating : 4.0-'],
        autopct = '%.1f%%')
plt.title('Restaurants serving more than one cuisine')
plt.show()
```

Below are the charts that shows how selling one cuisine and selling more than one cuisine depends on rating:



So, as we can see, serving some specific cuisine doesn't really help. Providing good and tasty food to the customers is what that matters whether you are serving one specific cuisine or not.

2. Find the weighted restaurant rating of each locality and find out the top 10 localities with more weighted restaurant rating?

*Weighted Restaurant Rating = $\Sigma (\text{number of votes} * \text{rating}) / \Sigma (\text{number of votes})$.*

Sol. Below is the code for finding the Top 10 localities in India with more weighted restaurant rating:

```
local = dict()
vote = dict()
for i in df1.index:
    key = df1['Locality'][i]
    if key in local:
        local[key] += round(df1['Votes'][i]*df1['Aggregate
rating'][i], 2)
        vote[key] += df1['Votes'][i]
    else:
        local[key] = round(df1['Votes'][i]*df1['Aggregate
rating'][i], 2)
        vote[key] = df1['Votes'][i]
for key in local:
    if local[key] == 0 or vote[key] == 0:
        continue
    local[key] = round(local[key]/vote[key], 2)

local = dict(sorted(local.items(), key = operator.itemgetter(1),
reverse = True))
print('Top 10 localities with more weighted restaurant rating...')
a = 0
for key in local:
    if a == 10:
        break
    print('%d. ' %(a+1), key, local[key])
    a += 1
```

Below is the list of top 10 localities I found out with more weighted restaurant rating:

```
Top 10 localities with more weighted restaurant rating...
1. Hotel Clarks Amer, Malviya Nagar 4.9
2. Aminabad 4.9
3. Friends Colony 4.89
4. Powai 4.84
5. Kirlampudi Layout 4.82
6. Express Avenue Mall, Royapettah 4.8
7. Deccan Gymkhana 4.8
8. Banjara Hills 4.72
9. Sector 5, Salt Lake 4.71
10. Riverside Mall, Gomti Nagar 4.7
```

Problem Statement – 3

Visualization

- 1. Plot the bar graph top 15 restaurants have a maximum number of outlets.**

Sol. Below is the code for the above problem:

```
outlet = dict()

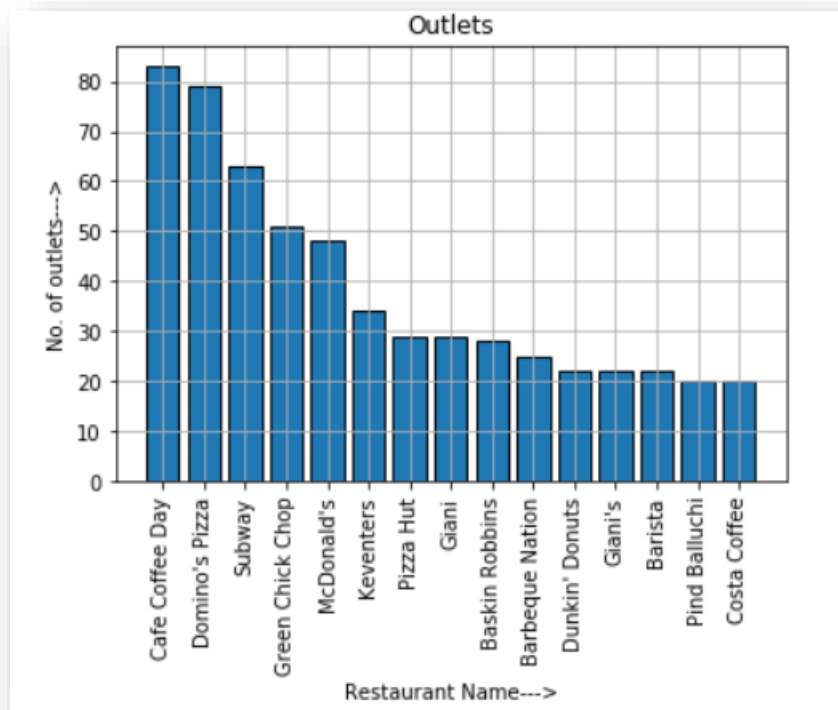
for i in df1.index:
    key = df1['Restaurant Name'][i]
    if key in outlet:
        outlet[key] += 1
    else:
        outlet[key] = 1

outlet = dict(sorted(outlet.items(), key = operator.itemgetter(1),
reverse = True))

a = 0
x = []
y = []
for i in outlet:
    if a == 15:
        break
    x.append(i)
    y.append(outlet[i])
    a += 1

plt.bar(x, y, edgecolor = "black")
plt.xticks(rotation = 90)
plt.xlabel('Restaurant Name--->')
plt.ylabel('No. of outlets--->')
plt.title('Outlets')
plt.grid()
plt.show()
```

Below is the bar graph from the above code:

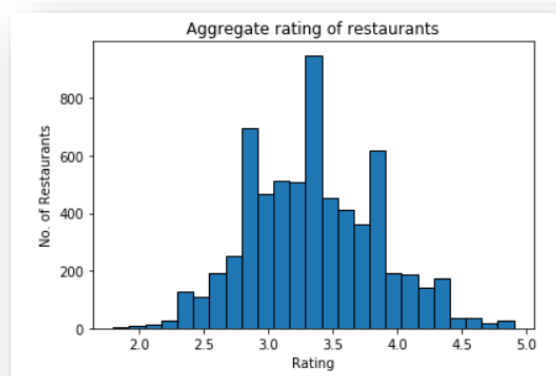


2. Plot the histogram of aggregate rating of restaurant(drop the unrated restaurant).

Sol. Below is the code for the above problem:

```
x = []
for i in df1.index:
    if df1['Aggregate rating'][i] == 0:
        continue
    x.append(df1['Aggregate rating'][i])
plt.hist(x, edgecolor = "black", bins = 25)
plt.xlabel('Rating')
plt.ylabel('No. of Restaurants')
plt.title('Aggregate rating of restaurants')
plt.show()
```

Below is the histogram from the above code:



From the above histogram, we can tell that most of the restaurants have rating between 2.8 and 4.0.

3. Plot the bar graph top 10 restaurants in the data with the highest number of votes.

Sol. Below is the code for the above problem:

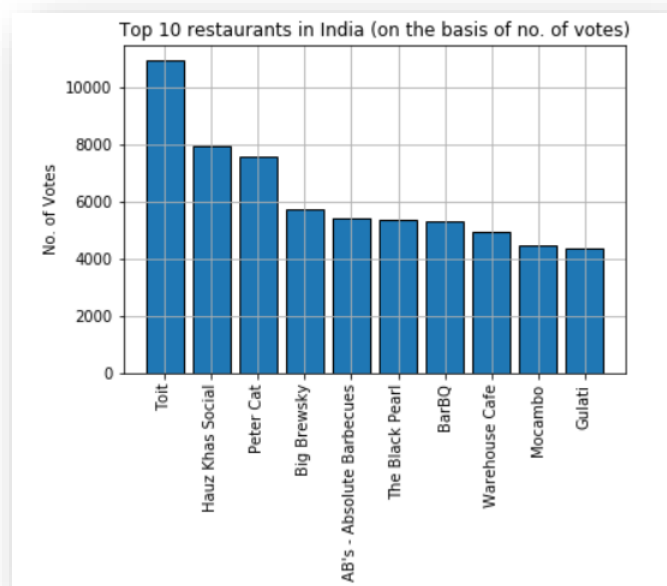
```
votes = dict()
for i in df1.index:
    key = df1['Restaurant Name'][i]
    votes[key] = df1['Votes'][i]

votes = dict(sorted(votes.items(), key = operator.itemgetter(1),
reverse = True))

x = []
y = []
a = 0
for i in votes:
    if a == 10:
        break
    x.append(i)
    y.append(votes[i])
    a += 1

plt.bar(x, y, edgecolor = "black")
plt.xticks(rotation = 90)
plt.title('Top 10 restaurants in India (on the basis of no. of
votes)')
plt.ylabel('No. of Votes')
plt.grid()
plt.show()
```

Below is the bar graph for the above problem:



4. Plot the pie graph of top 10 cuisines present in restaurants in the USA.

Sol. Below is the code for the above problem:

```
df2 = df[df['Country Code'] == 216]

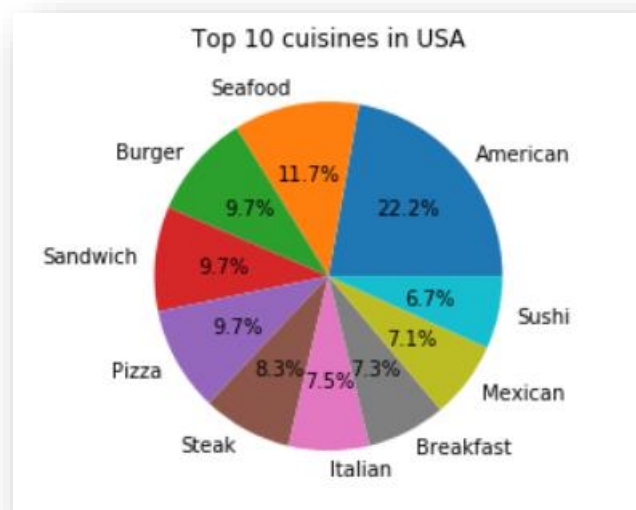
cuisine = dict()

for i in df2.index:
    temp = df2['Cuisines'][i]
    if type(temp) != str:
        continue
    temp = temp.split(',')
    for key in temp:
        if key.strip() in cuisine:
            cuisine[key.strip()] += 1
        else:
            cuisine[key.strip()] = 1

cuisine = dict(sorted(cuisine.items(), key =
operator.itemgetter(1), reverse = True))
x = []
y = []
a = 0
for i in cuisine:
    if a == 10:
        break
    x.append(i)
    y.append(cuisine[i])
    a += 1

plt.pie(y, labels = x, autopct = "%.1f%%")
plt.title('Top 10 cuisines in USA')
plt.show()
```

Below is the pie graph of the top 10 cuisines in United States of America:



5. Plot the bubble graph of a number of Restaurants present in the city of India and keeping the weighted restaurant rating of the city in a bubble.

Sol. Below is the code for the above problem:

```
city = dict()
vote = dict()
count = dict()
for i in df1.index:
    key = df1['City'][i]
    if key in city:
        city[key] += round(df1['Votes'][i]*df1['Aggregate
rating'][i], 2)
        vote[key] += df1['Votes'][i]
        count[key] += 1
    else:
        city[key] = round(df1['Votes'][i]*df1['Aggregate
rating'][i], 2)
        vote[key] = df1['Votes'][i]
        count[key] = 1
for key in city:
    if vote[key] == 0:
        continue
    city[key] = round(city[key]/vote[key], 2)

x = []
y = []
z = []

for i in city:
    x.append(i)
    y.append(city[i])
    z.append(count[i])

c = np.arange(len(x))
plt.scatter(x, z, s = y, c = c)
plt.xticks(rotation = 90)
plt.xlabel('Cities')
plt.ylabel('No. of restaurants')
plt.title('Bubble graph of no. of restaurants present in the city')
plt.show()
```

Below is the bubble graph that shows the no. of restaurants present in the city and their weighted restaurant ratings are in the bubbles of different colours.

Bubble graph of no. of restaurants present in the city

