

# Zomato

July 25, 2021

## 1 Geospatial Analysis

```
[1]: #importando as Bibliotecas
import pandas as pd
import numpy as np
import seaborn as sns
# Biblioteca que trata dos avisos
import warnings
warnings.filterwarnings('ignore')
```

```
[2]: df = pd.read_csv('zomato.csv')
```

```
[3]: #Metodo que Mostra as primeiras Linhas do dataset
df.head()
```

```
[3]:
```

	url \
0	https://www.zomato.com/bangalore/jalsa-banasha...
1	https://www.zomato.com/bangalore/spice-elephan...
2	https://www.zomato.com/SanchurroBangalore?cont...
3	https://www.zomato.com/bangalore/addhuri-udupi...
4	https://www.zomato.com/bangalore/grand-village...

  

	address	name \
0	942, 21st Main Road, 2nd Stage, Banashankari, ...	Jalsa
1	2nd Floor, 80 Feet Road, Near Big Bazaar, 6th ...	Spice Elephant
2	1112, Next to KIMS Medical College, 17th Cross...	San Churro Cafe
3	1st Floor, Annakuteera, 3rd Stage, Banashankar...	Addhuri Udupi Bhojana
4	10, 3rd Floor, Lakshmi Associates, Gandhi Baza...	Grand Village

  

	online_order	book_table	rate	votes	phone \
0	Yes	Yes	4.1/5	775	080 42297555\r\n+91 9743772233
1	Yes	No	4.1/5	787	080 41714161
2	Yes	No	3.8/5	918	+91 9663487993
3	No	No	3.7/5	88	+91 9620009302
4	No	No	3.8/5	166	+91 8026612447\r\n+91 9901210005

  

	location	rest_type \
--	----------	-------------

```

0 Banashankari      Casual Dining
1 Banashankari      Casual Dining
2 Banashankari Cafe, Casual Dining
3 Banashankari      Quick Bites
4 Basavanagudi      Casual Dining

```

```

                                dish_liked \
0 Pasta, Lunch Buffet, Masala Papad, Paneer Laja...
1 Momos, Lunch Buffet, Chocolate Nirvana, Thai G...
2 Churros, Cannelloni, Minestrone Soup, Hot Choc...
3                                     Masala Dosa
4                                Panipuri, Gol Gappe

```

```

                                cuisines approx_cost(for two people) \
0 North Indian, Mughlai, Chinese      800
1      Chinese, North Indian, Thai      800
2      Cafe, Mexican, Italian      800
3      South Indian, North Indian      300
4      North Indian, Rajasthani      600

```

```

                                reviews_list menu_item \
0 [('Rated 4.0', 'RATED\n A beautiful place to ...      []
1 [('Rated 4.0', 'RATED\n Had been here for din...      []
2 [('Rated 3.0', 'RATED\n Ambience is not that ...      []
3 [('Rated 4.0', 'RATED\n Great food and proper...      []
4 [('Rated 4.0', 'RATED\n Very good restaurant ...      []

```

```

listed_in(type) listed_in(city)
0      Buffet      Banashankari
1      Buffet      Banashankari
2      Buffet      Banashankari
3      Buffet      Banashankari
4      Buffet      Banashankari

```

```
[4]: # Metodo para mostrar valores nulos e Metodo Sum() para somar os valores
df.isna().sum()
```

```

[4]: url      0
     address    0
     name      0
     online_order  0
     book_table  0
     rate      7775
     votes      0
     phone     1208
     location   21
     rest_type  227

```

```

dish_liked          28078
cuisines             45
approx_cost(for two people) 346
reviews_list         0
menu_item            0
listed_in(type)      0
listed_in(city)      0
dtype: int64

```

```

[5]: #Metodo que remove todos os valores nulos.
      #Se o inplace estiver True atualiza o dataset Fisico
      df.dropna(axis='index', subset=['location'], inplace=True)

```

```

[6]: #Contar Nulos again
      df.isna().sum()

```

```

[6]: url          0
      address      0
      name         0
      online_order 0
      book_table   0
      rate         7754
      votes        0
      phone        1187
      location     0
      rest_type    206
      dish_liked   28057
      cuisines     24
      approx_cost(for two people) 325
      reviews_list 0
      menu_item    0
      listed_in(type) 0
      listed_in(city) 0
      dtype: int64

```

```

[7]: #Metodo que Mostra valores unicos na tabela Location
      df['location'].unique()

```

```

[7]: array(['Banashankari', 'Basavanagudi', 'Mysore Road', 'Jayanagar',
            'Kumaraswamy Layout', 'Rajarajeshwari Nagar', 'Vijay Nagar',
            'Uttarahalli', 'JP Nagar', 'South Bangalore', 'City Market',
            'Nagarbhavi', 'Bannerghatta Road', 'BTM', 'Kanakapura Road',
            'Bommanahalli', 'CV Raman Nagar', 'Electronic City', 'HSR',
            'Marathahalli', 'Sarjapur Road', 'Wilson Garden', 'Shanti Nagar',
            'Koramangala 5th Block', 'Koramangala 8th Block', 'Richmond Road',
            'Koramangala 7th Block', 'Jalahalli', 'Koramangala 4th Block',
            'Bellandur', 'Whitefield', 'East Bangalore', 'Old Airport Road',

```

```
'Indiranagar', 'Koramangala 1st Block', 'Frazer Town', 'RT Nagar',
'MG Road', 'Brigade Road', 'Lavelle Road', 'Church Street',
'Ulsoor', 'Residency Road', 'Shivajinagar', 'Infantry Road',
'St. Marks Road', 'Cunningham Road', 'Race Course Road',
'Commercial Street', 'Vasanth Nagar', 'HBR Layout', 'Domlur',
'Ejipura', 'Jeevan Bhima Nagar', 'Old Madras Road', 'Malleshwaram',
'Seshadripuram', 'Kammanahalli', 'Koramangala 6th Block',
'Majestic', 'Langford Town', 'Central Bangalore', 'Sanjay Nagar',
'Brookefield', 'ITPL Main Road, Whitefield',
'Varthur Main Road, Whitefield', 'KR Puram',
'Koramangala 2nd Block', 'Koramangala 3rd Block', 'Koramangala',
'Hosur Road', 'Rajajinagar', 'Banaswadi', 'North Bangalore',
'Nagawara', 'Hennur', 'Kalyan Nagar', 'New BEL Road', 'Jakkur',
'Rammurthy Nagar', 'Thippasandra', 'Kaggadasapura', 'Hebbal',
'Kengeri', 'Sankey Road', 'Sadashiv Nagar', 'Basaveshwara Nagar',
'Yeshwantpur', 'West Bangalore', 'Magadi Road', 'Yelahanka',
'Sahakara Nagar', 'Peenya'], dtype=object)
```

```
[8]: #Metodo que conta os valores unicos na coluna Location
len(df['location'].unique())
```

```
[8]: 93
```

```
[9]: #Criar um dataframe com os dados de localizacao unicos
locations=pd.DataFrame()
```

```
[10]: #Atribuindo nome da coluna e os dados unicos
locations['name']=df['location'].unique()
```

```
[11]: locations.head()
```

```
[11]:
```

	name
0	Banashankari
1	Basavanagudi
2	Mysore Road
3	Jayanagar
4	Kumaraswamy Layout

```
[12]: # Instalar biblioteca para analise Espacial, Bliiblioteca que extrai cordenadas:
→ Latitude e Longitude
!pip install geopy
```

```
Requirement already satisfied: geopy in c:\users\madalena\anaconda3\lib\site-
packages (2.2.0)
```

```
Requirement already satisfied: geographiclib<2,>=1.49 in
c:\users\madalena\anaconda3\lib\site-packages (from geopy) (1.52)
```

```
[13]: # De Geopy importamos geocoders e de geocoders vamos importar @Nominatim que
      ↳ e' responsavel por extrair as longitudes
      # e as latitudes
      from geopy.geocoders import Nominatim
```

```
[14]: # passamos o parametro @user_agent='app'
      geolocator=Nominatim(user_agent='app')
```

```
[15]: #vamos criar duas listas, uma de longitude e outra de latitude
      #vamos criar um instrutura de repeticao para coletar os dados de lon e lat em
      ↳ cada cidade na tabela acima
      #vamos converter a localizacao para Int, as localizacoes vem em forma de String
      lat=[]
      lon=[]
      for location in locations['name']:
          #convertendo a localizacao em formato geografico
          location = geolocator.geocode(location)
          if location is None:
              lat.append(np.nan)
              lon.append(np.nan)
          else:
              lat.append(location.latitude)
              lon.append(location.longitude)
```

```
[16]: print(lat,lon)
```

```
[15.8876779, 12.9417261, 12.3872141, 27.64392675, 12.9081487, 12.9274413,
22.8359967, 12.9055682, 12.2655944, 13.0646907, 39.76880625, 12.9546741,
12.887979, 45.95485055, 12.5607431, 12.9089453, 17.2510682, -6.2659285, 18.1475,
12.9552572, 12.9242381, 12.9489339, 12.9575547, 12.9343774, 12.9417812,
50.7721586, 12.9302645, 11.9917786, 12.93433385, 12.93577245, 44.3730577,
13.0215466, 62.442403, 12.9732913, 14.5395813, 12.996845, 13.0227204,
10.4545762, 40.28745, 40.7652844, 51.373656, 12.9778793, 38.7385916, 18.5322493,
34.977289, 51.5227651, 31.89376, 1.306731, 51.5164765, 12.988721250000001,
13.0358698, 12.9624669, 12.945245, 12.9678074, 12.9970537, 13.0027353,
12.9931876, 13.0093455, 12.9400321, 1.2847055, 12.957998, 13.0101286,
23.1485712, 33.5935063, 12.967576, 12.9414662, 13.007516, 12.9243509,
12.9271867, 13.2923988, 12.773175, 12.9882338, 13.0141618, 13.0217151, 13.2227,
13.0258087, 13.0221416, 13.0346384, 13.0784743, nan, 12.973936, 12.9846713,
13.0382184, 12.9176571, 38.7801076, 15.8782951, 12.9932739, 13.02383,
13.0011289, 12.945048, 13.1006982, 13.0621474, 13.0329419] [75.7046777,
77.5755021, 76.6669626, 83.05280519687284, 77.5553179, 77.5155224, 69.3405962,
77.5455438, 76.6465404, 77.49626895712257, -86.15345077251979, 77.5121724,
77.5970812, -112.49659530324134, 77.4258375, 77.6239038, 80.1651978,
106.7842561, 41.538889, 77.6984163, 77.6289059, 77.5968273, 77.5979099,
77.628415, 77.6160146, 0.09772783661369303, 77.6332585, 76.5066292,
77.63040639553275, 77.66676103753434, -71.6118577, 77.7640586, -114.3987951,
77.6404672, 121.070371, 77.6130165, 77.595715, 76.1233302, -76.964526,
```

```
-76.373824, -0.1042366, 77.6246697, -77.5275749, 73.8499601124847, -78.974578,
-0.7354457, -88.066644, 103.8497839, -0.0728317, 77.58516877601824, 77.6323597,
77.6381958, 77.6269144, 77.6568367, 77.669804, 77.5703253, 77.5753419,
77.6377094, 77.6203272, 103.84320655721689, 77.6037312, 77.5548006, 81.6048241,
-79.0345627, 77.7150877, 77.7470942, 77.695935, 77.6255562, 77.6266252,
77.7519261, 77.7831871, 77.554883, 77.6518539, 77.7660547, 78.5541977,
77.6305067, 77.6403368, 77.5681733, 77.6068938, nan, 77.6509982, 77.6790908,
77.5919, 77.4837568, -121.5056438, 74.5084834, 77.5388099, 77.5529215,
77.6325617, 77.263004, 77.5963454, 77.58006135480495, 77.5273253]
```

```
[17]: # vamos adicionar novas colunas com lat e lon na tabela locations
locations['lat']=lat
locations['lon']=lon
```

```
[18]: locations.head()
```

```
[18]:
```

	name	lat	lon
0	Banashankari	15.887678	75.704678
1	Basavanagudi	12.941726	77.575502
2	Mysore Road	12.387214	76.666963
3	Jayanagar	27.643927	83.052805
4	Kumaraswamy Layout	12.908149	77.555318

```
[19]: # Salvando o dataframe(Table Locations) em formato csv
locations.to_csv('zomato_locations.csv',index=False)
```

```
[20]: #Conta de restaurante em cada localizacao
df['location'].value_counts().reset_index()
```

```
[20]:
```

	index	location
0	BTM	5124
1	HSR	2523
2	Koramangala 5th Block	2504
3	JP Nagar	2235
4	Whitefield	2144
..	...	...
88	West Bangalore	6
89	Yelahanka	6
90	Jakkur	3
91	Rajarajeshwari Nagar	2
92	Peenya	1

```
[93 rows x 2 columns]
```

```
[21]: #vamos personalizar os nomes das colunas
Reset_locations=df['location'].value_counts().reset_index()
```

```
[22]: #vamos atribuir os valores em formato de lista
Reset_locations.columns=['name','count']
Reset_locations
```

```
[22]:
```

	name	count
0	BTM	5124
1	HSR	2523
2	Koramangala 5th Block	2504
3	JP Nagar	2235
4	Whitefield	2144
..	...	...
88	West Bangalore	6
89	Yelahanka	6
90	Jakkur	3
91	Rajarajeshwari Nagar	2
92	Peenya	1

[93 rows x 2 columns]

```
[23]: #Em ordem para analise espacial precisamos da lat e lon, nesse caso vamos fazer
↳um append(merge) do antigo dataset e o novo
#com as lat e lon de cada localizacao com base no nome da localizacao das duas
↳tabelas.
Restaurant_locations=Reset_locations.merge(locations,on='name',how='left').
↳dropna()
Restaurant_locations.head()
```

```
[23]:
```

	name	count	lat	lon
0	BTM	5124	45.954851	-112.496595
1	HSR	2523	18.147500	41.538889
2	Koramangala 5th Block	2504	12.934377	77.628415
3	JP Nagar	2235	12.265594	76.646540
4	Whitefield	2144	44.373058	-71.611858

```
[24]: Restaurant_locations
```

```
[24]:
```

	name	count	lat	lon
0	BTM	5124	45.954851	-112.496595
1	HSR	2523	18.147500	41.538889
2	Koramangala 5th Block	2504	12.934377	77.628415
3	JP Nagar	2235	12.265594	76.646540
4	Whitefield	2144	44.373058	-71.611858
..	...	...	...	...
88	West Bangalore	6	13.001129	77.632562
89	Yelahanka	6	13.100698	77.596345
90	Jakkur	3	13.078474	77.606894
91	Rajarajeshwari Nagar	2	12.927441	77.515522

92 Peenya 1 13.032942 77.527325

[92 rows x 4 columns]

```
[25]: # vamos instalar a biblioteca para visualizacao de mapa geoespacial
!pip install folium
```

```
Requirement already satisfied: folium in c:\users\madalena\anaconda3\lib\site-
packages (0.12.1)
Requirement already satisfied: numpy in c:\users\madalena\anaconda3\lib\site-
packages (from folium) (1.20.1)
Requirement already satisfied: branca>=0.3.0 in
c:\users\madalena\anaconda3\lib\site-packages (from folium) (0.4.2)
Requirement already satisfied: jinja2>=2.9 in
c:\users\madalena\anaconda3\lib\site-packages (from folium) (2.11.3)
Requirement already satisfied: requests in c:\users\madalena\anaconda3\lib\site-
packages (from folium) (2.25.1)
Requirement already satisfied: MarkupSafe>=0.23 in
c:\users\madalena\anaconda3\lib\site-packages (from jinja2>=2.9->folium) (1.1.1)
Requirement already satisfied: certifi>=2017.4.17 in
c:\users\madalena\anaconda3\lib\site-packages (from requests->folium)
(2020.12.5)
Requirement already satisfied: chardet<5,>=3.0.2 in
c:\users\madalena\anaconda3\lib\site-packages (from requests->folium) (4.0.0)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in
c:\users\madalena\anaconda3\lib\site-packages (from requests->folium) (1.26.4)
Requirement already satisfied: idna<3,>=2.5 in
c:\users\madalena\anaconda3\lib\site-packages (from requests->folium) (2.10)
```

```
[26]: #Vamos definir uma funcao, para sempre Basemap
def generatebasemap(default_location=[12.97,77.59],default_zoom_start=12):
    basemap=folium.Map(location=default_location,zoom_start=default_zoom_start)
    return basemap
```

```
[27]: # vamos importar a biblioteca folium
import folium
#chamando a funcao e atribuindo a variavel basemap
basemap=generatebasemap()
```

```
[28]: basemap
```

```
[28]: <folium.folium.Map at 0x28151577b20>
```

```
[29]: # Vamos importar Heatmap from Folium
from folium.plugins import HeatMap
```

```
[30]: # a tabela que queremos criar o HeatMap
Reset_locations
```



```
[30]:
```

	name	count
0	BTM	5124
1	HSR	2523
2	Koramangala 5th Block	2504
3	JP Nagar	2235
4	Whitefield	2144
..	""	""
88	West Bangalore	6
89	Yelahanka	6
90	Jakkur	3
91	Rajarajeshwari Nagar	2
92	Peenya	1

[93 rows x 2 columns]

```
[31]: HeatMap(Restaurant_locations[['lat', 'lon', 'count']], zoom=20).add_to(basemap)
```

```
[31]: <folium.plugins.heat_map.HeatMap at 0x28153000340>
```

```
[32]: basemap
```

```
[32]: <folium.folium.Map at 0x28151577b20>
```

### 1.0.1 Marker Cluster Map

```
[33]: # Vamos importar o plugin da Biblioteca folium
from folium.plugins import FastMarkerCluster
```

```
[34]: FastMarkerCluster(Restaurant_locations[['lat', 'lon', 'count']], zoom=20).
      ↪add_to(basemap)
```

```
[34]: <folium.plugins.fast_marker_cluster.FastMarkerCluster at 0x28152fac790>
```

```
[35]: basemap
```

```
[35]: <folium.folium.Map at 0x28151577b20>
```

```
[36]: df.head()
```

```
[36]:
```

	url \	address	name \
0	https://www.zomato.com/bangalore/jalsa-banasha...		Jalsa
1	https://www.zomato.com/bangalore/spice-elephan...		
2	https://www.zomato.com/SanchurroBangalore?cont...		
3	https://www.zomato.com/bangalore/addhuri-udupi...		
4	https://www.zomato.com/bangalore/grand-village...		
0		942, 21st Main Road, 2nd Stage, Banashankari, ...	

1	2nd Floor, 80 Feet Road, Near Big Bazaar, 6th ...	Spice Elephant
2	1112, Next to KIMS Medical College, 17th Cross...	San Churro Cafe
3	1st Floor, Annakuteera, 3rd Stage, Banashankar...	Addhuri Udupi Bhojana
4	10, 3rd Floor, Lakshmi Associates, Gandhi Baza...	Grand Village

	online_order	book_table	rate	votes	phone \
0	Yes	Yes	4.1/5	775	080 42297555\r\n+91 9743772233
1	Yes	No	4.1/5	787	080 41714161
2	Yes	No	3.8/5	918	+91 9663487993
3	No	No	3.7/5	88	+91 9620009302
4	No	No	3.8/5	166	+91 8026612447\r\n+91 9901210005

	location	rest_type \
0	Banashankari	Casual Dining
1	Banashankari	Casual Dining
2	Banashankari Cafe,	Casual Dining
3	Banashankari	Quick Bites
4	Basavanagudi	Casual Dining

	dish_liked \
0	Pasta, Lunch Buffet, Masala Papad, Paneer Laja...
1	Momos, Lunch Buffet, Chocolate Nirvana, Thai G...
2	Churros, Cannelloni, Minestrone Soup, Hot Choc...
3	Masala Dosa
4	Panipuri, Gol Gappe

	cuisines approx_cost(for two people) \
0	North Indian, Mughlai, Chinese 800
1	Chinese, North Indian, Thai 800
2	Cafe, Mexican, Italian 800
3	South Indian, North Indian 300
4	North Indian, Rajasthani 600

	reviews_list menu_item \
0	[('Rated 4.0', 'RATED\n A beautiful place to ... []
1	[('Rated 4.0', 'RATED\n Had been here for din... []
2	[('Rated 3.0', 'RATED\n Ambience is not that ... []
3	[('Rated 4.0', 'RATED\n Great food and proper... []
4	[('Rated 4.0', 'RATED\n Very good restaurant ... []

	listed_in(type)	listed_in(city)
0	Buffet	Banashankari
1	Buffet	Banashankari
2	Buffet	Banashankari
3	Buffet	Banashankari
4	Buffet	Banashankari

```
[37]: # Vamos verificar os dados unicos na tabela RATE
df['rate'].unique()
```

```
[37]: array(['4.1/5', '3.8/5', '3.7/5', '3.6/5', '4.6/5', '4.0/5', '4.2/5',
        '3.9/5', '3.1/5', '3.0/5', '3.2/5', '3.3/5', '2.8/5', '4.4/5',
        '4.3/5', 'NEW', '2.9/5', '3.5/5', nan, '2.6/5', '3.8 /5', '3.4/5',
        '4.5/5', '2.5/5', '2.7/5', '4.7/5', '2.4/5', '2.2/5', '2.3/5',
        '3.4 /5', '-', '3.6 /5', '4.8/5', '3.9 /5', '4.2 /5', '4.0 /5',
        '4.1 /5', '3.7 /5', '3.1 /5', '2.9 /5', '3.3 /5', '2.8 /5',
        '3.5 /5', '2.7 /5', '2.5 /5', '3.2 /5', '2.6 /5', '4.5 /5',
        '4.3 /5', '4.4 /5', '4.9/5', '2.1/5', '2.0/5', '1.8/5', '4.6 /5',
        '4.9 /5', '3.0 /5', '4.8 /5', '2.3 /5', '4.7 /5', '2.4 /5',
        '2.1 /5', '2.2 /5', '2.0 /5', '1.8 /5'], dtype=object)
```

```
[38]: # Descartando os valores vazios da coluna RATE
df.dropna(axis=0,subset=['rate'],inplace=True)
```

```
[39]: # vamos definir uma funcao que vai remover o separador / e o valor 5 do
      ↪ dataframe @Split
def split(x):
    return x.split('/')[0]
```

```
[40]: # vamos aplicar o metode e gravar como uma nova coluna na nosso df
df['rating']=df['rate'].apply(split)
```

```
[41]: df.head()
```

```
[41]:
```

	url \	address	name \
0	https://www.zomato.com/bangalore/jalsa-banasha...	942, 21st Main Road, 2nd Stage, Banashankari, ...	Jalsa
1	https://www.zomato.com/bangalore/spice-elephan...	2nd Floor, 80 Feet Road, Near Big Bazaar, 6th ...	Spice Elephant
2	https://www.zomato.com/SanchurroBangalore?cont...	1112, Next to KIMS Medical College, 17th Cross...	San Churro Cafe
3	https://www.zomato.com/bangalore/addhuri-udupi...	1st Floor, Annakuteera, 3rd Stage, Banashankar...	Addhuri Udupi Bhojana
4	https://www.zomato.com/bangalore/grand-village...	10, 3rd Floor, Lakshmi Associates, Gandhi Baza...	Grand Village

  

	online_order	book_table	rate	votes	phone \
0	Yes	Yes	4.1/5	775	080 42297555\r\n+91 9743772233
1	Yes	No	4.1/5	787	080 41714161
2	Yes	No	3.8/5	918	+91 9663487993
3	No	No	3.7/5	88	+91 9620009302
4	No	No	3.8/5	166	+91 8026612447\r\n+91 9901210005

	location	rest_type \
0	Banashankari	Casual Dining
1	Banashankari	Casual Dining
2	Banashankari	Cafe, Casual Dining
3	Banashankari	Quick Bites
4	Basavanagudi	Casual Dining

	dish_liked \
0	Pasta, Lunch Buffet, Masala Papad, Paneer Laja...
1	Momos, Lunch Buffet, Chocolate Nirvana, Thai G...
2	Churros, Cannelloni, Minestrone Soup, Hot Choc...
3	Masala Dosa
4	Panipuri, Gol Gappe

	cuisines approx_cost(for two people) \
0	North Indian, Mughlai, Chinese 800
1	Chinese, North Indian, Thai 800
2	Cafe, Mexican, Italian 800
3	South Indian, North Indian 300
4	North Indian, Rajasthani 600

	reviews_list menu_item \
0	[('Rated 4.0', 'RATED\n A beautiful place to ... []
1	[('Rated 4.0', 'RATED\n Had been here for din... []
2	[('Rated 3.0', 'RATED\n Ambience is not that ... []
3	[('Rated 4.0', 'RATED\n Great food and proper... []
4	[('Rated 4.0', 'RATED\n Very good restaurant ... []

	listed_in(type)	listed_in(city)	rating
0	Buffet	Banashankari	4.1
1	Buffet	Banashankari	4.1
2	Buffet	Banashankari	3.8
3	Buffet	Banashankari	3.7
4	Buffet	Banashankari	3.8

```
[42]: df['rating'].unique()
```

```
[42]: array(['4.1', '3.8', '3.7', '3.6', '4.6', '4.0', '4.2', '3.9', '3.1',
        '3.0', '3.2', '3.3', '2.8', '4.4', '4.3', 'NEW', '2.9', '3.5',
        '2.6', '3.8 ', '3.4', '4.5', '2.5', '2.7', '4.7', '2.4', '2.2',
        '2.3', '3.4 ', '-', '3.6 ', '4.8', '3.9 ', '4.2 ', '4.0 ', '4.1 ',
        '3.7 ', '3.1 ', '2.9 ', '3.3 ', '2.8 ', '3.5 ', '2.7 ', '2.5 ',
        '3.2 ', '2.6 ', '4.5 ', '4.3 ', '4.4 ', '4.9', '2.1', '2.0', '1.8',
        '4.6 ', '4.9 ', '3.0 ', '4.8 ', '2.3 ', '4.7 ', '2.4 ', '2.1 ',
        '2.2 ', '2.0 ', '1.8 '], dtype=object)
```

```
[43]: # vamos substituir o a string New por 0
df.replace('NEW',0,inplace=True)
```

```
[44]: df.replace('-',0,inplace=True)
```

```
[45]: df['rating'].unique()
```

```
[45]: array(['4.1', '3.8', '3.7', '3.6', '4.6', '4.0', '4.2', '3.9', '3.1',
        '3.0', '3.2', '3.3', '2.8', '4.4', '4.3', 0, '2.9', '3.5', '2.6',
        '3.8 ', '3.4 ', '4.5', '2.5', '2.7', '4.7', '2.4', '2.2', '2.3',
        '3.4 ', '3.6 ', '4.8', '3.9 ', '4.2 ', '4.0 ', '4.1 ', '3.7 ',
        '3.1 ', '2.9 ', '3.3 ', '2.8 ', '3.5 ', '2.7 ', '2.5 ', '3.2 ',
        '2.6 ', '4.5 ', '4.3 ', '4.4 ', '4.9', '2.1', '2.0', '1.8', '4.6 ',
        '4.9 ', '3.0 ', '4.8 ', '2.3 ', '4.7 ', '2.4 ', '2.1 ', '2.2 ',
        '2.0 ', '1.8 '], dtype=object)
```

```
[46]: #Vamos verificar os tipos de dados
df.dtypes
```

```
[46]: url                object
address                object
name                  object
online_order          object
book_table            object
rate                  object
votes                 int64
phone                 object
location              object
rest_type             object
dish_liked            object
cuisines              object
approx_cost(for two people) object
reviews_list          object
menu_item             object
listed_in(type)       object
listed_in(city)       object
rating                object
dtype: object
```

```
[47]: #convertendo os dados da coluna rating de object para numerico e salvando na
      ↳propria coluna
df['rating']=pd.to_numeric(df['rating'])
```

```
[48]: #agrupando os dados e mostrando a media de Rating por localizacao
df.groupby('location')['rating'].mean()
```

```
[48]: location
      BTM          3.296128
      Banashankari  3.373292
      Banaswadi     3.362926
      Bannerghatta Road 3.271677
      Basavanagudi  3.478185
      ...
      West Bangalore 2.020000
      Whitefield     3.384170
      Wilson Garden  3.257635
      Yelahanka      3.640000
      Yeshwantpur    3.502679
      Name: rating, Length: 92, dtype: float64
```

```
[49]: #ordenando os valores do maior para menor
      df.groupby('location')['rating'].mean().sort_values(ascending=False)
```

```
[49]: location
      Lavelle Road      4.042886
      St. Marks Road    4.017201
      Koramangala 3rd Block 3.978756
      Sankey Road        3.965385
      Church Street      3.963091
      ...
      Electronic City     3.041909
      Bommanahalli        2.926752
      Hebbal              2.880000
      North Bangalore     2.385714
      West Bangalore      2.020000
      Name: rating, Length: 92, dtype: float64
```

```
[50]: #Mostra todas valores medio de Rating
      avg_rating=df.groupby('location')['rating'].mean().sort_values(ascending=False).
      ↪values
```

```
[51]: avg_rating
```

```
[51]: array([4.04288577, 4.01720117, 3.97875648, 3.96538462, 3.96309091,
        3.90151197, 3.90105263, 3.85      , 3.84457237, 3.83965517,
        3.81435185, 3.80740741, 3.79642857, 3.74784206, 3.74055024,
        3.7270073 , 3.72222222, 3.70492958, 3.68801262, 3.68783784,
        3.66823708, 3.66246625, 3.65216942, 3.64      , 3.625      ,
        3.61525029, 3.6075      , 3.60645161, 3.59584871, 3.583174  ,
        3.58095238, 3.56487889, 3.54565217, 3.54139845, 3.52914439,
        3.51111111, 3.50267857, 3.49980952, 3.49891697, 3.48695652,
        3.48406955, 3.47894737, 3.4787234 , 3.47818471, 3.47362637,
        3.47355822, 3.45555556, 3.44615385, 3.44551724, 3.4375      ,
```

```

3.42977099, 3.42238193, 3.41978022, 3.41292591, 3.40053227,
3.4          , 3.38911917, 3.38554779, 3.38417011, 3.3746988 ,
3.37329193, 3.36292585, 3.36          , 3.33333333, 3.32358974,
3.3202381 , 3.3202381 , 3.30983302, 3.29892473, 3.29612767,
3.2940678 , 3.27927928, 3.278125   , 3.27167674, 3.26393782,
3.25763547, 3.24929577, 3.24333333, 3.21818182, 3.21515152,
3.2          , 3.18181818, 3.10970874, 3.10714286, 3.09539474,
3.09389313, 3.05472973, 3.04190871, 2.92675159, 2.88          ,
2.38571429, 2.02          ])

```

```

[52]: #mostra todas localizacoes
location=df.groupby('location')['rating'].mean().sort_values(ascending=False).
      ↪index

```

```

[53]: location

```

```

[53]: Index(['Lavelle Road', 'St. Marks Road', 'Koramangala 3rd Block',
'Sankey Road', 'Church Street', 'Koramangala 5th Block',
'Cunningham Road', 'Rajarajeshwari Nagar', 'Residency Road',
'Sadashiv Nagar', 'Koramangala 4th Block', 'Langford Town',
'Infantry Road', 'Koramangala 7th Block', 'MG Road', 'Race Course Road',
'Kengeri', 'Seshadripuram', 'Richmond Road', 'Hosur Road',
'Malleswaram', 'Koramangala 6th Block', 'Indiranagar', 'Yelahanka',
'Central Bangalore', 'Jayanagar', 'Koramangala 8th Block',
'Koramangala', 'Brigade Road', 'New BEL Road', 'Vasanth Nagar',
'Frazer Town', 'Koramangala 2nd Block', 'Ulsoor', 'Kalyan Nagar',
'Uttarahalli', 'Yeshwantpur', 'Kammanahalli', 'Shivajinagar',
'Jalahalli', 'HSR', 'Kanakapura Road', 'Sahakara Nagar', 'Basavanagudi',
'Kaggadasapura', 'Sarjapur Road', 'Mysore Road', 'City Market',
'Basaveshwara Nagar', 'Magadi Road', 'Jeevan Bhima Nagar',
'Rajajinagar', 'South Bangalore', 'JP Nagar', 'Marathahalli',
'Nagarbhavi', 'Old Airport Road', 'Domlur', 'Whitefield', 'Brookefield',
'Banashankari', 'Banaswadi', 'Sanjay Nagar', 'Nagawara', 'Shanti Nagar',
'ITPL Main Road, Whitefield', 'Kumaraswamy Layout', 'Bellandur',
'Varthur Main Road, Whitefield', 'BTM', 'Majestic', 'HBR Layout',
'RT Nagar', 'Bannerghatta Road', 'Koramangala 1st Block',
'Wilson Garden', 'Vijay Nagar', 'East Bangalore', 'KR Puram',
'CV Raman Nagar', 'Peenya', 'Old Madras Road', 'Commercial Street',
'Rammurthy Nagar', 'Thippasandra', 'Hennur', 'Ejipura',
'Electronic City', 'Bommanahalli', 'Hebbal', 'North Bangalore',
'West Bangalore'],
dtype='object', name='location')

```

```

[54]: #vamos criar novo dataframe
rating=pd.DataFrame()

```

```
[55]: #vamos criar duas listas, uma de longitude e outra de latitude
#vamos criar um instrutura de repeticao para coletar os dados de lon e lat em
      ↳ cada cidade na tabela acima
#vamos converter a localizacao para Int, as localizacoes vem em forma de String
lat=[]
lon=[]
for loc in location:
    #convertendo a localizacao em formato geografico
    loc = geolocator.geocode(loc)
    if loc is None:
        lat.append(np.nan)
        lon.append(np.nan)
    else:
        lat.append(loc.latitude)
        lon.append(loc.longitude)
```

```
[56]: # vamos poular o nosso dataframe
rating['location']=location
rating['lat']=lat
rating['lon']=lon
rating['avg_rating']=avg_rating
```

```
[57]: rating.head()
```

```
[57]:
```

	location	lat	lon	avg_rating
0	Lavelle Road	40.765284	-76.373824	4.042886
1	St. Marks Road	51.522765	-0.735446	4.017201
2	Koramangala 3rd Block	12.927187	77.626625	3.978756
3	Sankey Road	38.780108	-121.505644	3.965385
4	Church Street	51.373656	-0.104237	3.963091

```
[58]: rating.isna().sum()
```

```
[58]: location      0
lat              1
lon              1
avg_rating       0
dtype: int64
```

```
[59]: rating.dropna(inplace=True)
```

```
[60]: rating.isna().sum()
```

```
[60]: location      0
lat              0
lon              0
avg_rating       0
```



dtype: int64

```
[61]: #dataset que vamos passar para o Heatmap
rating[['lat', 'lon', 'avg_rating']]
```

```
[61]:
```

	lat	lon	avg_rating
0	40.765284	-76.373824	4.042886
1	51.522765	-0.735446	4.017201
2	12.927187	77.626625	3.978756
3	38.780108	-121.505644	3.965385
4	51.373656	-0.104237	3.963091
..	...	...	...
87	-6.265929	106.784256	3.041909
88	12.908945	77.623904	2.926752
89	13.038218	77.591900	2.880000
90	13.021715	77.766055	2.385714
91	13.001129	77.632562	2.020000

[91 rows x 3 columns]

```
[62]: HeatMap(rating[['lat', 'lon', 'avg_rating']]).add_to(basemap)
```

```
[62]: <folium.plugins.heat_map.HeatMap at 0x2815381a8e0>
```

```
[63]: basemap
```

```
[63]: <folium.folium.Map at 0x28151577b20>
```

```
[64]: df.head()
```

```
[64]:
```

	url \	address	name \
0	https://www.zomato.com/bangalore/jalsa-banasha...	942, 21st Main Road, 2nd Stage, Banashankari, ...	Jalsa
1	https://www.zomato.com/bangalore/spice-elephan...	2nd Floor, 80 Feet Road, Near Big Bazaar, 6th ...	Spice Elephant
2	https://www.zomato.com/SanchurroBangalore?cont...	1112, Next to KIMS Medical College, 17th Cross...	San Churro Cafe
3	https://www.zomato.com/bangalore/addhuri-udupi...	1st Floor, Annakuteera, 3rd Stage, Banashankar...	Addhuri Udupi Bhojana
4	https://www.zomato.com/bangalore/grand-village...	10, 3rd Floor, Lakshmi Associates, Gandhi Baza...	Grand Village

  

	online_order	book_table	rate	votes	phone \
0	Yes	Yes	4.1/5	775	080 42297555\r\n+91 9743772233
1	Yes	No	4.1/5	787	080 41714161
2	Yes	No	3.8/5	918	+91 9663487993

3	No	No	3.7/5	88	+91 9620009302
4	No	No	3.8/5	166	+91 8026612447\r\n+91 9901210005

	location	rest_type \
0	Banashankari	Casual Dining
1	Banashankari	Casual Dining
2	Banashankari	Cafe, Casual Dining
3	Banashankari	Quick Bites
4	Basavanagudi	Casual Dining

	dish_liked \
0	Pasta, Lunch Buffet, Masala Papad, Paneer Laja...
1	Momos, Lunch Buffet, Chocolate Nirvana, Thai G...
2	Churros, Cannelloni, Minestrone Soup, Hot Choc...
3	Masala Dosa
4	Panipuri, Gol Gappe

	cuisines approx_cost(for two people) \
0	North Indian, Mughlai, Chinese 800
1	Chinese, North Indian, Thai 800
2	Cafe, Mexican, Italian 800
3	South Indian, North Indian 300
4	North Indian, Rajasthani 600

	reviews_list menu_item \
0	[('Rated 4.0', 'RATED\n A beautiful place to ... []
1	[('Rated 4.0', 'RATED\n Had been here for din... []
2	[('Rated 3.0', 'RATED\n Ambience is not that ... []
3	[('Rated 4.0', 'RATED\n Great food and proper... []
4	[('Rated 4.0', 'RATED\n Very good restaurant ... []

	listed_in(type)	listed_in(city)	rating
0	Buffet	Banashankari	4.1
1	Buffet	Banashankari	4.1
2	Buffet	Banashankari	3.8
3	Buffet	Banashankari	3.7
4	Buffet	Banashankari	3.8

```
[65]: #Vamos filter Cuisines de North da Indian
filter=df['cuisines']=="North Indian"
#criamos um df
df2=df[filter]
```

```
[66]: df2.head()
```

```
[66]: url \
5 https://www.zomato.com/bangalore/timepass-dinn...
```

50 https://www.zomato.com/bangalore/petoo-banasha...  
 72 https://www.zomato.com/bangalore/spicy-tandoor...  
 87 https://www.zomato.com/bangalore/krishna-sagar...  
 94 https://www.zomato.com/bangalore/nandhini-delu...

	address	name \
5	37, 5-1, 4th Floor, Bosco Court, Gandhi Bazaar...	Timepass Dinner
50	276, Ground Floor, 100 Feet Outer Ring Road, B...	Petoo
72	Opposite ICICI Bank, Hanuman Nagar, Banashanka...	Spicy Tandoor
87	38, 22nd Main, 22nd Cross, Opposite BDA, 2nd S...	Krishna Sagar
94	304, Opposite Apollo Public School, 100 Feet R...	Nandhini Deluxe

	online_order	book_table	rate	votes	phone \
5	Yes	No	3.8/5	286	+91 9980040002\r\n+91 9980063005
50	No	No	3.7/5	21	+91 8026893211
72	No	No	0	0	+91 8050884222
87	No	No	3.5/5	31	+91 8892752997\r\n+91 7204780429
94	No	No	2.6/5	283	080 26890011\r\n080 26890033

	location	rest_type \
5	Basavanagudi	Casual Dining
50	Banashankari	Quick Bites
72	Banashankari	Quick Bites
87	Banashankari	Quick Bites
94	Banashankari	Casual Dining

	dish_liked	cuisines \
5	Onion Rings, Pasta, Kadhai Paneer, Salads, Sal...	North Indian
50		NaN North Indian
72		NaN North Indian
87		NaN North Indian
94	Biryani, Chicken Guntur, Thali, Buttermilk, Ma...	North Indian

	approx_cost(for two people) \
5	600
50	450
72	150
87	200
94	600

	reviews_list	menu_item \
5	[('Rated 3.0', 'RATED\n Food 3/5\nAmbience 3/...	[]
50	[('Rated 2.0', 'RATED\n This is a neatly made...	[]
72	[('Rated 4.0', 'RATED\n cost for chicken roll...	[]
87	[('Rated 1.0', 'RATED\n Worst experience with...	[]
94	[('Rated 3.0', 'RATED\n Ididnt like much.\n\n...	[]

	listed_in(type)	listed_in(city)	rating
5	Buffet	Banashankari	3.8
50	Delivery	Banashankari	3.7
72	Delivery	Banashankari	0.0
87	Delivery	Banashankari	3.5
94	Delivery	Banashankari	2.6

```
[67]: #Agrupando location e url de North da Indian
df2.groupby('location')['url'].count().reset_index()
```

```
[67]:
```

	location	url
0	BTM	274
1	Banashankari	35
2	Banaswadi	9
3	Bannerghatta Road	60
4	Basavanagudi	17
..	...	...
59	Varthur Main Road, Whitefield	3
60	Vasanth Nagar	12
61	Whitefield	148
62	Wilson Garden	37
63	Yeshwantpur	3

[64 rows x 2 columns]

```
[68]: #criando novo dataframe
north_india=df2.groupby('location')['url'].count().reset_index()
#renomeando as clunas
north_india.columns=['name','count']
```

```
[69]: north_india
```

```
[69]:
```

	name	count
0	BTM	274
1	Banashankari	35
2	Banaswadi	9
3	Bannerghatta Road	60
4	Basavanagudi	17
..	...	...
59	Varthur Main Road, Whitefield	3
60	Vasanth Nagar	12
61	Whitefield	148
62	Wilson Garden	37
63	Yeshwantpur	3

[64 rows x 2 columns]

```
[70]: # vamos fazer um merge com o dataframe de location
north_india=north_india.merge(locations,on='name',how='left').dropna()
north_india
```

```
[70]:
```

	name	count	lat	lon
0	BTM	274	45.954851	-112.496595
1	Banashankari	35	15.887678	75.704678
2	Banaswadi	9	13.014162	77.651854
3	Bannerghatta Road	60	12.887979	77.597081
4	Basavanagudi	17	12.941726	77.575502
..	...	...	...	...
59	Varthur Main Road, Whitefield	3	12.941466	77.747094
60	Vasanth Nagar	12	12.988721	77.585169
61	Whitefield	148	44.373058	-71.611858
62	Wilson Garden	37	12.948934	77.596827
63	Yeshwantpur	3	13.023830	77.552921

[64 rows x 4 columns]

```
[71]: HeatMap(north_india[['lat','lon','count']],zoom=20,radius=15).add_to(basemap)
```

```
[71]: <folium.plugins.heat_map.HeatMap at 0x281537fc160>
```

```
[72]: basemap
```

```
[72]: <folium.folium.Map at 0x28151577b20>
```

```
[73]: df['cuisines'].unique()
```

```
[73]: array(['North Indian, Mughlai, Chinese', 'Chinese, North Indian, Thai',
        'Cafe, Mexican, Italian', ..., 'Tibetan, Nepalese',
        'North Indian, Street Food, Biryani',
        'North Indian, Chinese, Arabian, Momos'], dtype=object)
```

```
[74]: # vamos criar uma funcao para automacao de criacao de Heatmap
def Heatmap_zone(zone):
    #Vamos filter Cuisines de North da Indian
    filter=df['cuisines']==zone
    #criamos um df
    df2=df[filter]
    #criando novo dataframe
    df_zone=df2.groupby('location')['url'].count().reset_index()
    #renomeando as clunas
    df_zone.columns=['name','count']
    df_zone=df_zone.merge(locations,on='name',how='left').dropna()
    HeatMap(df_zone[['lat','lon','count']],zoom=20,radius=15).add_to(basemap)
    return basemap
```

```
[77]: Heatmap_zone('North Indian')
```

```
[77]: <folium.folium.Map at 0x28151577b20>
```

```
[76]: Heatmap_zone('Mexican')
```

```
[76]: <folium.folium.Map at 0x28151577b20>
```

```
[78]: !pip install --upgrade --user nbconvert
```

```
Requirement already satisfied: nbconvert in  
c:\users\madalena\anaconda3\lib\site-packages (6.0.7)
```

```
WARNING: The script jupyter-nbconvert.exe is installed in  
'C:\Users\Madalena\AppData\Roaming\Python\Python38\Scripts' which is not on  
PATH.
```

```
Consider adding this directory to PATH or, if you prefer to suppress this  
warning, use --no-warn-script-location.
```

```
ERROR: pip's dependency resolver does not currently take into account all the  
packages that are installed. This behaviour is the source of the following  
dependency conflicts.
```

```
spyder 4.2.5 requires pyqt5<5.13, which is not installed.
```

```
spyder 4.2.5 requires PyQtWebEngine<5.13, which is not installed.
```

```
Collecting nbconvert
```

```
Downloading nbconvert-6.1.0-py3-none-any.whl (551 kB)
```

```
Requirement already satisfied: pandocfilters>=1.4.1 in  
c:\users\madalena\anaconda3\lib\site-packages (from nbconvert) (1.4.3)
```

```
Requirement already satisfied: jupyterlab-pygments in  
c:\users\madalena\anaconda3\lib\site-packages (from nbconvert) (0.1.2)
```

```
Requirement already satisfied: Jinja2>=2.4 in  
c:\users\madalena\anaconda3\lib\site-packages (from nbconvert) (2.11.3)
```

```
Requirement already satisfied: nbclient<0.6.0,>=0.5.0 in  
c:\users\madalena\anaconda3\lib\site-packages (from nbconvert) (0.5.3)
```

```
Requirement already satisfied: entrypoints>=0.2.2 in  
c:\users\madalena\anaconda3\lib\site-packages (from nbconvert) (0.3)
```

```
Requirement already satisfied: Pygments>=2.4.1 in  
c:\users\madalena\anaconda3\lib\site-packages (from nbconvert) (2.8.1)
```

```
Requirement already satisfied: bleach in c:\users\madalena\anaconda3\lib\site-  
packages (from nbconvert) (3.3.0)
```

```
Requirement already satisfied: mistune<2,>=0.8.1 in  
c:\users\madalena\anaconda3\lib\site-packages (from nbconvert) (0.8.4)
```

```
Requirement already satisfied: testpath in c:\users\madalena\anaconda3\lib\site-  
packages (from nbconvert) (0.4.4)
```

```
Requirement already satisfied: nbformat>=4.4 in  
c:\users\madalena\anaconda3\lib\site-packages (from nbconvert) (5.1.3)
```

```
Requirement already satisfied: traitlets>=5.0 in  
c:\users\madalena\anaconda3\lib\site-packages (from nbconvert) (5.0.5)
```

Requirement already satisfied: defusedxml in  
c:\users\madalena\anaconda3\lib\site-packages (from nbconvert) (0.7.1)

Requirement already satisfied: jupyter-core in  
c:\users\madalena\anaconda3\lib\site-packages (from nbconvert) (4.7.1)

Requirement already satisfied: MarkupSafe>=0.23 in  
c:\users\madalena\anaconda3\lib\site-packages (from jinja2>=2.4->nbconvert)  
(1.1.1)

Requirement already satisfied: nest-asyncio in  
c:\users\madalena\anaconda3\lib\site-packages (from  
nbclient<0.6.0,>=0.5.0->nbconvert) (1.5.1)

Requirement already satisfied: async-generator in  
c:\users\madalena\anaconda3\lib\site-packages (from  
nbclient<0.6.0,>=0.5.0->nbconvert) (1.10)

Requirement already satisfied: jupyter-client>=6.1.5 in  
c:\users\madalena\anaconda3\lib\site-packages (from  
nbclient<0.6.0,>=0.5.0->nbconvert) (6.1.12)

Requirement already satisfied: pyzmq>=13 in  
c:\users\madalena\anaconda3\lib\site-packages (from jupyter-  
client>=6.1.5->nbclient<0.6.0,>=0.5.0->nbconvert) (20.0.0)

Requirement already satisfied: tornado>=4.1 in  
c:\users\madalena\anaconda3\lib\site-packages (from jupyter-  
client>=6.1.5->nbclient<0.6.0,>=0.5.0->nbconvert) (6.1)

Requirement already satisfied: python-dateutil>=2.1 in  
c:\users\madalena\anaconda3\lib\site-packages (from jupyter-  
client>=6.1.5->nbclient<0.6.0,>=0.5.0->nbconvert) (2.8.1)

Requirement already satisfied: pywin32>=1.0 in  
c:\users\madalena\anaconda3\lib\site-packages (from jupyter-core->nbconvert)  
(227)

Requirement already satisfied: ipython-genutils in  
c:\users\madalena\anaconda3\lib\site-packages (from nbformat>=4.4->nbconvert)  
(0.2.0)

Requirement already satisfied: jsonschema!=2.5.0,>=2.4 in  
c:\users\madalena\anaconda3\lib\site-packages (from nbformat>=4.4->nbconvert)  
(3.2.0)

Requirement already satisfied: six>=1.11.0 in  
c:\users\madalena\anaconda3\lib\site-packages (from  
jsonschema!=2.5.0,>=2.4->nbformat>=4.4->nbconvert) (1.15.0)

Requirement already satisfied: pyrsistent>=0.14.0 in  
c:\users\madalena\anaconda3\lib\site-packages (from  
jsonschema!=2.5.0,>=2.4->nbformat>=4.4->nbconvert) (0.17.3)

Requirement already satisfied: attrs>=17.4.0 in  
c:\users\madalena\anaconda3\lib\site-packages (from  
jsonschema!=2.5.0,>=2.4->nbformat>=4.4->nbconvert) (20.3.0)

Requirement already satisfied: setuptools in  
c:\users\madalena\anaconda3\lib\site-packages (from  
jsonschema!=2.5.0,>=2.4->nbformat>=4.4->nbconvert) (52.0.0.post20210125)

Requirement already satisfied: webencodings in  
c:\users\madalena\anaconda3\lib\site-packages (from bleach->nbconvert) (0.5.1)

```
Requirement already satisfied: packaging in  
c:\users\madalena\anaconda3\lib\site-packages (from bleach->nbconvert) (20.9)  
Requirement already satisfied: pyparsing>=2.0.2 in  
c:\users\madalena\anaconda3\lib\site-packages (from  
packaging->bleach->nbconvert) (2.4.7)  
Installing collected packages: nbconvert  
Successfully installed nbconvert-6.1.0
```

[ ]: