Peer-graded Assignment: Capstone Project - The Battle of Neighborhoods (Week 2)

Keep calm and go to London

Import required libs

In [1]:
```
!conda install -c conda-forge folium=0.5.0 --yes
import folium
import numpy as np
import pandas as pd
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)
from bs4 import BeautifulSoup
import requests
!conda install -c conda-forge geopy --yes
from geopy.geocoders import Nominatim
import json # library to handle JSON files
from pandas.io.json import json_normalize
print('Libraries imported.')
```

```
Fetching package metadata .............
Solving package specifications: .

# All requested packages already installed.
# packages in environment at /opt/conda/envs/DSX-Python35:
#
folium                    0.5.0                      py_0    conda-forge
Fetching package metadata .............
Solving package specifications: .

# All requested packages already installed.
# packages in environment at /opt/conda/envs/DSX-Python35:
#
geopy                     1.18.1                     py_0    conda-forge
Libraries imported.
```

Read Postcode data and create dataframe from CSV and then select required columns and print the final shape

In [2]:
```
!wget -q -O 'london_postcode.csv' https://www.doogal.co.uk/UKPostcodesCSV.ashx?area=London
print('Data downloaded!')

london_postcode_raw_df = pd.read_csv("london_postcode.csv")
london_postcode_raw_df.shape
```

```
Data downloaded!
```

Out[2]:
```
(319385, 43)
```

In [3]:
```
london_postcode_df = london_postcode_raw_df[['Postcode', 'Latitude', 'Longitude', 'District', 'Ward', 'London zone', 'Nearest station', 'Distance to station', 'Postcode area', 'Postcode district']]
london_postcode_df.shape
```

Out[3]:
```
(319385, 10)
```

Create dataframe from top level venue categories (https://developer.foursquare.com/docs/resources/categories (https://developer.foursquare.com/docs/resources/categories))

In [4]:
```
vdata =  {'Arts & Entertainment':'4d4b7104d754a06370d81259',
      'Food':'4d4b7105d754a06374d81259',
      'Museum': '4bf58dd8d48988d181941735',
      'College & University':'4d4b7105d754a06372d81259',
      'Nightlife Spot': '4d4b7105d754a06376d81259',
      'Outdoors & Recreation':'4d4b7105d754a06377d81259',
      'Clothing Store':'4bf58dd8d48988d103951735',
     }

df_venue_categories = pd.DataFrame(list(vdata.items()), columns=['Venue', 'Valu
e'])
```

Find geograpical coordinate of the London

In [5]:
```
address = 'London , UK'
geolocator = Nominatim(user_agent="week5-assignment")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print('The geograpical coordinate of the London are {}, {}.'.format(latitude, l
ongitude))
london_map = folium.Map(location=[latitude, longitude], zoom_start=11)
```

```
The geograpical coordinate of the London are 51.5073219, -0.1276474.
```

Four Foursquare ID and password

In [6]:
```
CLIENT_ID = 'XXXXXXXXXX' # your Foursquare ID
CLIENT_SECRET = 'XXXXXXXXXXX' # your Foursquare Secret
VERSION = '20180605' # Foursquare API version
```

Util function that extracts the category of the venue

In [7]:
```
def get_category_type(row):
    try:
        categories_list = row['categories']
    except:
        categories_list = row['venue.categories']

    if len(categories_list) == 0:
        return None
    else:
        return categories_list[0]['name']
```

Util function to return foursquare json results as dataset

In [8]:
```
def get_venues_as_df(results):
    venues = results['response']['groups'][0]['items']
    nearby_venues = json_normalize(venues)

    # filter columns
    filtered_columns = ['venue.name', 'venue.categories', 'venue.location.lat',
'venue.location.lng','venue.location.postalCode']
    nearby_venues_df = nearby_venues.loc[:, filtered_columns]

    # filter the category for each row
    nearby_venues_df['venue.categories'] = nearby_venues_df.apply(get_category_
type, axis=1)

    # clean columns
    nearby_venues_df.columns = [col.split(".")[-1] for col in nearby_venues_df.
columns]
    nearby_venues_df.columns = ['Venue', 'Venue Category', 'Venue Latitude','Ve
nue Longitude','Postcode']
    nearby_venues_df.head
    return nearby_venues_df
```

Request foursquare and build top venus in selected categories and then merge the dataframes

In [9]:
```
radius = 40000
LIMIT = 100

cat_top_df = pd.DataFrame([])
cat_top_10_df = pd.DataFrame([])

for index, row in df_venue_categories.iterrows():
    venue_category = row['Value']
    url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_se
cret={}&v={}&ll={},{}&radius={}&limit={}&categoryId={}'.format(
    CLIENT_ID, CLIENT_SECRET, VERSION, latitude, longitude, radius,LIMIT,venue_
category)
    results = requests.get(url).json()
    #print(results)
    result_df = get_venues_as_df(results)
    cat_top_df = cat_top_df.append(result_df, ignore_index=True)
    cat_top_10_df= cat_top_10_df.append(result_df.head(10), ignore_index=True)

cat_top_df.shape
```

Out[9]:
```
(700, 5)
```

In [10]:
```
cat_top_df.head()
```

Out[10]:

	Venue	Venue Category	Venue Latitude	Venue Longitude	Postcode
0	National Gallery	Art Museum	51.508876	-0.128478	WC2N 5DN
1	National Portrait Gallery	Art Gallery	51.509438	-0.128032	WC2H 0HE
2	Churchill War Rooms (Churchill Museum & Cabine...	Historic Site	51.502079	-0.129305	SW1A 2AQ
3	Royal Academy of Arts	Art Museum	51.508848	-0.139327	W1J 0BD
4	British Museum	History Museum	51.518988	-0.126510	WC1B 3DG

In [11]:
```
london_postcode_df.shape
cat_top_v2_df = pd.merge(london_postcode_df, cat_top_df, on=['Postcode'], how='
inner')
cat_top_v2_df = cat_top_v2_df[['Venue', 'Venue Category', 'Venue Latitude','Ven
ue Longitude','Postcode','District','London zone', 'Nearest station']]


cat_top_10_v2_df = pd.merge(london_postcode_df, cat_top_10_df, on=['Postcode'],
how='inner')
cat_top_10_v2_df = cat_top_10_v2_df[['Venue', 'Venue Category', 'Venue Latitude
','Venue Longitude','Postcode','District','London zone', 'Nearest station']]

cat_top_v2_df.head()
```

Out[11]:

	Venue	Venue Category	Venue Latitude	Venue Longitude	Postcode	District	London zone	Nearest station
0	Kings College London	General College & University	51.407757	-0.033425	BR3 4PR	Bromley	4	Clock House
1	Marks & Spencer	Clothing Store	51.377423	-0.102015	CR9 1SH	Croydon	5	West Croydon
2	Imparando	College Classroom	51.514875	-0.067404	E1 1LP	Tower Hamlets	1	Aldgate East
3	Francis Bancroft Building	College Academic Building	51.524670	-0.040718	E1 4AH	Tower Hamlets	2	Stepney Green
4	People's Palace	College Academic Building	51.523127	-0.041107	E1 4NS	Tower Hamlets	2	Stepney Green

List how many venues were returned for each London boroughs

In [12]:
```
cat_top_v2_df.groupby('District').count()
```

Out[12]:

District	Venue	Venue Category	Venue Latitude	Venue Longitude	Postcode	London zone	Nearest station
Barnet	2	2	2	2	2	2	2
Brent	2	2	2	2	2	2	2
Bromley	3	3	3	3	3	3	3
Camden	81	81	81	81	81	81	81
City of London	21	21	21	21	21	21	21
Croydon	1	1	1	1	1	1	1
Ealing	4	4	4	4	4	4	4
Greenwich	16	16	16	16	16	16	16
Hackney	21	21	21	21	21	21	21
Hammersmith and Fulham	10	10	10	10	10	10	10
Haringey	5	5	5	5	5	5	5
Hounslow	3	3	3	3	3	3	3
Islington	27	27	27	27	27	27	27
Kensington and Chelsea	43	43	43	43	43	43	43
Lambeth	22	22	22	22	22	22	22
Lewisham	4	4	4	4	4	4	4
Merton	4	4	4	4	4	4	4
Newham	5	5	5	5	5	5	5
Redbridge	1	1	1	1	1	1	1
Richmond upon Thames	16	16	16	16	16	16	16
Southwark	30	30	30	30	30	30	30
Tower Hamlets	17	17	17	17	17	17	17
Waltham Forest	4	4	4	4	4	4	4
Wandsworth	3	3	3	3	3	3	3
Westminster	271	271	271	271	271	271	271

Analyze each boroughs and then group rows and by taking the mean of the frequency of occurrence of each category

In [13]:
```
cat_top_v2_df_grouped = cat_top_v2_df.groupby('District').mean().reset_index()
```

In [14]:
```
london_onehot = pd.get_dummies(cat_top_v2_df[['Venue Category']], prefix="", pr
efix_sep="")

london_onehot['District'] = cat_top_v2_df['District']

fixed_columns = [london_onehot.columns[-1]] + list(london_onehot.columns[:-1])
london_onehot = london_onehot[fixed_columns]

london_onehot.head()
```

Out[14]:

	District	American Restaurant	Argentinian Restaurant	Art Gallery	Art Museum	Asian Restaurant	Athletics & Sports	BBQ Joint	Bakery	Bar	E
0	Bromley	0	0	0	0	0	0	0	0	0	0
1	Croydon	0	0	0	0	0	0	0	0	0	0
2	Tower Hamlets	0	0	0	0	0	0	0	0	0	0
3	Tower Hamlets	0	0	0	0	0	0	0	0	0	0
4	Tower Hamlets	0	0	0	0	0	0	0	0	0	0

In [15]:
```
london_grouped = london_onehot.groupby('District').mean().reset_index()
london_grouped
```

Out[15]:

	District	American Restaurant	Argentinian Restaurant	Art Gallery	Art Museum	Asian Restaurant	Athletics & Sports	BBQ Joint	B
0	Barnet	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.00000	0.0
1	Brent	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.00000	0.0
2	Bromley	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.00000	0.0
3	Camden	0.00000	0.00000	0.012346	0.012346	0.012346	0.00000	0.00000	0.0
4	City of London	0.00000	0.00000	0.047619	0.000000	0.000000	0.00000	0.00000	0.0
5	Croydon	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.00000	0.0
6	Ealing	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.00000	0.0
7	Greenwich	0.00000	0.00000	0.125000	0.000000	0.000000	0.00000	0.00000	0.0
8	Hackney	0.00000	0.00000	0.047619	0.000000	0.000000	0.00000	0.00000	0.0
9	Hammersmith and Fulham	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.00000	0.0
10	Haringey	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.00000	0.0
11	Hounslow	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.00000	0.0
12	Islington	0.00000	0.00000	0.037037	0.037037	0.000000	0.00000	0.00000	0.0
13	Kensington and Chelsea	0.00000	0.00000	0.023256	0.069767	0.000000	0.00000	0.00000	0.0
14	Lambeth	0.00000	0.00000	0.045455	0.000000	0.000000	0.00000	0.00000	0.0
15	Lewisham	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.00000	0.0
16	Merton	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.00000	0.0
17	Newham	0.00000	0.00000	0.000000	0.000000	0.000000	0.20000	0.00000	0.0
18	Redbridge	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.00000	0.0
19	Richmond upon Thames	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.00000	0.0
20	Southwark	0.00000	0.00000	0.100000	0.133333	0.000000	0.00000	0.00000	0.0
21	Tower Hamlets	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.00000	0.0
22	Waltham Forest	0.00000	0.00000	0.750000	0.000000	0.000000	0.00000	0.00000	0.0
23	Wandsworth	0.00000	0.00000	0.000000	0.000000	0.000000	0.00000	0.00000	0.0
24	Westminster	0.00738	0.00738	0.033210	0.025830	0.007380	0.00369	0.00369	0.0

Print top boroughs along with the top 5 most common venues

In [16]:
```
num_top_venues = 5

for hood in london_onehot['District']:
    print("----"+hood+"----")
    temp = london_grouped[london_grouped['District'] == hood].T.reset_index()
    temp.columns = ['venue','freq']
    temp = temp.iloc[1:]
    temp['freq'] = temp['freq'].astype(float)
    temp = temp.round({'freq': 2})
    print(temp.sort_values('freq', ascending=False).reset_index(drop=True).head
(num_top_venues))
    print('\n')
```

```
----Bromley----
                         venue  freq
0                         Park  0.33
1  General College & University  0.33
2             Sculpture Garden  0.33
3          American Restaurant  0.00
4                       Museum  0.00


----Croydon----
                 venue  freq
0       Clothing Store   1.0
1  American Restaurant   0.0
2          Music Venue   0.0
3          Planetarium   0.0
4          Pizza Place   0.0


----Tower Hamlets----
                      venue  freq
0  College Academic Building  0.24
1                     Castle  0.12
2                     Museum  0.12
3             History Museum  0.12
4               Cocktail Bar  0.06


----Tower Hamlets----
                      venue  freq
0  College Academic Building  0.24
1                     Castle  0.12
2                     Museum  0.12
3             History Museum  0.12
4               Cocktail Bar  0.06


----Tower Hamlets----
                      venue  freq
0  College Academic Building  0.24
1                     Castle  0.12
2                     Museum  0.12
3             History Museum  0.12
4               Cocktail Bar  0.06


----Tower Hamlets----
                      venue  freq
0  College Academic Building  0.24
1                     Castle  0.12
2                     Museum  0.12
3             History Museum  0.12
4               Cocktail Bar  0.06


----Tower Hamlets----
                      venue  freq
0  College Academic Building  0.24
1                     Castle  0.12
2                     Museum  0.12
3             History Museum  0.12
4               Cocktail Bar  0.06


----Hackney----
            venue  freq
0  Clothing Store  0.19
1    Cocktail Bar  0.14
2            Park  0.10
3         Theater  0.05
4     Climbing Gym  0.05
```

Print top London boroughs with popular venues

In [17]:
```
top_boroughs = cat_top_v2_df.groupby('District').count()
top_boroughs.sort_values("Postcode", inplace=True,ascending=False)
top_boroughs.iloc[0:10,0:1]
```

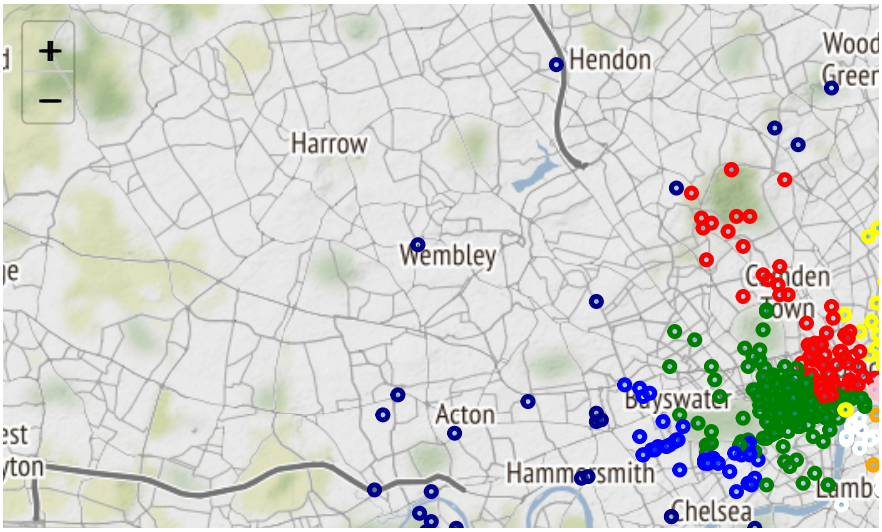
Out[17]:

	Venue
District	
Westminster	271
Camden	81
Kensington and Chelsea	43
Southwark	30
Islington	27
Lambeth	22
Hackney	21
City of London	21
Tower Hamlets	17
Greenwich	16

Put it all on a map

In [18]:
```
# function to color per borough

def regioncolors(borough):
    if borough == 'Westminster':
        return 'green'
    elif borough == 'Lambeth':
        return 'white'
    elif borough == 'Camden':
        return 'red'
    elif borough == 'Kensington and Chelsea':
        return 'blue'
    elif borough== 'Southwark':
        return 'orange'
    elif borough == 'Islington':
        return 'yellow'
    elif borough == 'Hackney':
        return 'olive'
    elif borough == 'City of London':
        return 'pink'
    elif borough == 'Tower Hamlets':
        return 'purple'
    elif borough == 'Greenwich':
        return 'rose'
    else:
        return 'darkblue'



london_map = folium.Map(location=[latitude, longitude], tiles='Stamen Terrain',
zoom_start=11,height=500)


for lat, lng, name, pcode, borough in zip(cat_top_v2_df['Venue Latitude'], cat_
top_v2_df['Venue Longitude'],cat_top_v2_df['Venue'], cat_top_v2_df['Postcode'],
cat_top_v2_df['District'] ):
    label = '{}, {}'.format(name, pcode)
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [float(lat), float(lng)],
        radius=3,
        popup=label,
        color=regioncolors(borough),
        fill=True,
        fill_color='#3199cc',
        fill_opacity=0.3,
        parse_html=False).add_to(london_map)

london_map
```

Out[18]:



List all parent categories

In [19]:
```
df_venue_categories
```

Out[19]:

	Venue	Value
0	Museum	4bf58dd8d48988d181941735
1	Clothing Store	4bf58dd8d48988d103951735
2	Food	4d4b7105d754a06374d81259
3	Outdoors & Recreation	4d4b7105d754a06377d81259
4	College & University	4d4b7105d754a06372d81259
5	Arts & Entertainment	4d4b7104d754a06370d81259
6	Nightlife Spot	4d4b7105d754a06376d81259

List top venus from each of the categories

In [20]:
```
cat_top_10_v2_df
```

Out[20]:

	Venue	Venue Category	Venue Latitude	Venue Longitude	Postcode	District	London zone	Nearest station
0	Kings College London	General College & University	51.407757	-0.033425	BR3 4PR	Bromley	4	Clock House
1	Bishopsgate Institute	Historic Site	51.518874	-0.079114	EC2M 4QH	City of London	1	Liverpool Street
2	Wellcome Library	Library	51.526163	-0.134052	NW1 2BE	Camden	1	Euston
3	Imperial War Museum	History Museum	51.495985	-0.108642	SE1 6HZ	Southwark	1	Lambeth North
4	The London Eye	Scenic Lookout	51.503287	-0.119594	SE1 7PB	Lambeth	1	Westminster
5	Shakespeare's Globe Theatre	Theater	51.508115	-0.096946	SE1 9DT	Southwark	1	Mansion House
6	Tate Modern	Art Museum	51.507704	-0.099456	SE1 9TG	Southwark	1	Blackfriars
7	Dukes Bar	Hotel Bar	51.505632	-0.139481	SW1A 1NY	Westminster	1	Green Park
8	Churchill War Rooms (Churchill Museum & Cabine...	Historic Site	51.502079	-0.129305	SW1A 2AQ	Westminster	1	Westminster
9	Horse Guards Parade	Plaza	51.504847	-0.126590	SW1A 2AX	Westminster	1	Charing Cross
10	St James's Park	Park	51.503253	-0.132995	SW1A 2BJ	Westminster	1	Westminster
11	Tate Britain	Art Museum	51.490772	-0.126965	SW1P 4RG	Westminster	1	Pimlico
12	Milos	Greek Restaurant	51.508117	-0.133341	SW1Y 4NR	Westminster	1	Piccadilly Circus
13	Her Majesty's Theatre	Theater	51.508289	-0.131641	SW1Y 4QL	Westminster	1	Piccadilly Circus
14	Ole & Steen	Bakery	51.509219	-0.132597	SW1Y 4RN	Westminster	1	Piccadilly Circus
15	The Royal Automobile Club	Lounge	51.506333	-0.135002	SW1Y 5HS	Westminster	1	Piccadilly Circus
16	Tramp	Nightclub	51.508332	-0.137836	SW1Y 6DN	Westminster	1	Piccadilly Circus
17	Burberry	Clothing Store	51.510694	-0.138894	W1B 4TB	Westminster	1	Piccadilly Circus
18	COS	Clothing Store	51.513544	-0.140967	W1B 5BD	Westminster	1	Oxford Circus
19	UNIQLO	Clothing Store	51.510067	-0.137079	W1B 5RR	Westminster	1	Piccadilly Circus
20	Tommy Hilfiger	Clothing Store	51.511300	-0.138972	W1B 5SG	Westminster	1	Piccadilly Circus

Finally who fancies a curry? In case you like other cuisine or want to do something else then please chose the category for activity from the following page from https://developer.foursquare.com/docs/resources/categories (https://developer.foursquare.com/docs/resources/categories)

In [21]:
```
print('Enter a category to fetch top venue,eg. 4bf58dd8d48988d10f941735 for Ind
ian Restaurant')
entered_venue_category = input()
```

```
Enter a category to fetch top venue,eg. 4bf58dd8d48988d10f941735 for Indian Re
staurant
4bf58dd8d48988d10f941735
```

In [22]:
```
url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret
={}&v={}&ll={},{}&radius={}&limit={}&categoryId={}'.format(
    CLIENT_ID, CLIENT_SECRET, VERSION, latitude, longitude, radius,LIMIT,entere
d_venue_category)
nearby_venus = requests.get(url).json()
nearby_venus_df = get_venues_as_df(nearby_venus)


nearby_venus_df = pd.merge(london_postcode_df, nearby_venus_df, on=['Postcode']
, how='inner')
nearby_venus_df = nearby_venus_df[['Venue', 'Venue Category', 'Venue Latitude',
'Venue Longitude','Postcode','District','London zone', 'Nearest station']]

nearby_venus_df.head(10)
```

Out[22]:

	Venue	Venue Category	Venue Latitude	Venue Longitude	Postcode	District	London zone	Nearest station
0	Needoo Grill	Indian Restaurant	51.517070	-0.062379	E1 1HH	Tower Hamlets	2	Whitechapel
1	Zaza's	Indian Restaurant	51.518066	-0.064340	E1 1HJ	Tower Hamlets	2	Whitechapel
2	Tayyabs	North Indian Restaurant	51.517240	-0.063476	E1 1JU	Tower Hamlets	2	Whitechapel
3	Lahore Kebab House	Indian Restaurant	51.514483	-0.062912	E1 1PY	Tower Hamlets	2	Shadwell
4	Dosa World	Indian Restaurant	51.520245	-0.071358	E1 5JL	Tower Hamlets	1	Shoreditch High Street
5	Tifinbox	Indian Restaurant	51.516345	-0.077195	E1 7DB	City of London	1	Aldgate
6	Gunpowder	Indian Restaurant	51.518436	-0.074732	E1 7NF	Tower Hamlets	1	Aldgate East
7	Cafe Spice Namaste	Indian Restaurant	51.511485	-0.070693	E1 8AZ	Tower Hamlets	1	Tower Gateway
8	Halal Restaurant	Indian Restaurant	51.513429	-0.071699	E1 8DJ	Tower Hamlets	1	Aldgate East
9	Taste Of India	Indian Restaurant	51.542572	0.050107	E12 6PH	Newham	3	East Ham