Capstone Project - The Battle of Neig

August 21, 2019

0.0.1 BUSINESS SITE ANALYSIS: A LOOK AT SOME CITIES IN NIGERIA

Applied-Data-Science-Capstone on Coursera
By: Ezeonyebuchi E.C

1 A. Introduction:

Nigeria is the most populous country in Africa, as such has a big inpact on the economy of all African nations.

Within the Nigerian landspace, there are differences in the population of the cities therin. This is one factors one has to bear in mind when establishing any kind of business because business is about people and site is important to business.

Lagos is a city in Nigeria with a high population density. It accommodates all the ethnic groups and social classes in the country. When we think of investing, investors, prefer the districts where there is high population density with equally, the right population that can afford the service(s) they wish to offer and the type of business they want to establish is high in demand.

It's often not easy to establish these variables unless one takes extra care in investigation and analysis.

This is what this project is going to unravel:

1. Someone who wishes to establish a restaurant
2. A contractor who intends to open up a business

2 B. Data Sourcing

It was a difficult adventure that took a lot of time to acquire the right data for this project as most African countries are yet to be precisely geo-mapped like their counterparts in the other continents. Variables like postal codes and neighborhood mapping are none existent or jumbled. Google and other geo-mapping companies don't find it viable to venture into remote Africa because most of the roads are bad and untarred. The buildings are most often not planned or distorted because of corruption of the officials in charge for ennforcing the town planning laws.

The difficulty of sourcing neighborhood data during analysis, is quite a big problem, not only in Africa but in some other conutries and is an issue of considerable discussion as obtained from sites such as this: https://wpgeodirectory.com/support/topic/setting-up-a-proper-neighborhood-map-url/

However, after considerable search and without getting the 'ideal' dataset I sought, I settled for this source for my data, ng.csv an open source data from: https://www.npmjs.com/package/ng-csv.

Let's download all the dependencies that we will need.

[40]:

```
import numpy as np # library to handle data in a vectorized manner

#!pip install beautifulsoup4

#from bs4 import BeautifulSoup
#import requests

import pandas as pd # library for data analsysis
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

import json # library to handle JSON files

!conda install -c conda-forge geopy --yes # uncomment this line if you haven't completed the
  ↪Foursquare API lab
from geopy.geocoders import Nominatim # convert an address into latitude and longitude values

import requests # library to handle requests
from pandas.io.json import json_normalize # tranform JSON file into a pandas dataframe

# Matplotlib and associated plotting modules
import matplotlib.cm as cm
import matplotlib.colors as colors

# import k-means from clustering stage
from sklearn.cluster import KMeans

#!conda install -c conda-forge folium=0.5.0 --yes # uncomment this line if you haven't
  ↪completed the Foursquare API lab
import folium # map rendering library

!pip install lxml

!pip install geopy

print('Libraries imported.')
```

```
Solving environment: done


==> WARNING: A newer version of conda exists. <==
  current version: 4.5.11
  latest version: 4.7.11
```

```
Please update conda by running

    $ conda update -n base -c defaults conda



# All requested packages already installed.

Requirement already satisfied: lxml in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (4.4.1)
Requirement already satisfied: geopy in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (1.20.0)
Requirement already satisfied: geographiclib<2,>=1.49 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from geopy)
(1.49)
Libraries imported.
```

[41]:

```
# Print and read data
url = '/resources/data/ng.csv'
print(url)
```

```
/resources/data/ng.csv
```

[42]:

```
# Use pandas to read the csv file into a table and look at the first five entries
ng_df = pd.read_csv('/resources/data/ng.csv')
ng_df.head()
```

[42]:

```
     city        lat       lng  country iso2                     admin  \
0   Lagos   6.454066  3.394673  Nigeria   NG                     Lagos
1    Kano  12.002381  8.513160  Nigeria   NG                      Kano
2  Ibadan   7.377562  3.905907  Nigeria   NG                       Oyo
3   Abuja   9.083333  7.533333  Nigeria   NG  Federal Capital Territory
4  Kaduna  10.526413  7.438795  Nigeria   NG                    Kaduna

   capital  population  population_proper
0    minor   9466000.0             1536.0
1    admin   3140000.0          3140000.0
2    admin   2628000.0          1814570.0
3  primary   1576000.0           162135.0
4    admin   1442000.0           940593.0
```

[43]:

```
# Check the last five entries to see if all is fine
ng_df.tail()
```

[43]:

```
       city        lat       lng  country iso2     admin capital  \
745   Ikire   7.372414  4.187394  Nigeria   NG      Osun   minor
746   Kalgo  12.326664  4.200400  Nigeria   NG     Kebbi   minor
747   Wamba   8.941532  8.603154  Nigeria   NG  Nasarawa   minor
```

```
748    Wamako  13.030541  5.104326  Nigeria   NG    Sokoto   minor
749  Minjibir  12.177652  8.657818  Nigeria   NG      Kano   minor

     population  population_proper
745         NaN                NaN
746         NaN                NaN
747         NaN                NaN
748         NaN                NaN
749         NaN                NaN
```

There are NaN values and has to be removed latter

[44]:
```
# Look at the statistics of the dataframe
ng_df.describe()
```

[44]:
```
              lat         lng    population  population_proper
count  750.000000  750.000000  6.800000e+01       6.800000e+01
mean     8.714484    7.401900  5.903178e+05       3.220102e+05
std      2.661091    2.467611  1.227963e+06       4.652362e+05
min      4.312306    2.842465  9.351000e+03       1.536000e+03
25%      6.463062    5.559045  1.039538e+05       6.858225e+04
50%      8.092312    7.332433  2.608335e+05       1.853210e+05
75%     11.401723    8.660263  6.996910e+05       4.332632e+05
max     13.754303   14.465521  9.466000e+06       3.140000e+06
```

[45]:
```
# Let's check the city with the maximal population
ng_df[ng_df['population'] == 9.466000e+06]
```

[45]:
```
    city       lat       lng  country iso2  admin capital  population  \
0  Lagos  6.454066  3.394673  Nigeria   NG  Lagos   minor   9466000.0

   population_proper
0             1536.0
```

Lagos city has the highest population

[46]:
```
# Let's check the city with the minimal population
ng_df[ng_df['population'] == 9.351000e+03]
```

[46]:
```
    city       lat       lng  country iso2 admin capital  population  \
67  Orlu  5.795645  7.035126  Nigeria   NG   Imo   minor      9351.0

    population_proper
67             9351.0
```

Orlu city has the least population amongst the cities
Let's drop the NaN values

[47]:
```
# Let's drop NaNs and check the tail where it occoured
ng_df2 = ng_df.dropna()
ng_df2.tail()
```

[47]:
```
      city        lat        lng  country iso2   admin capital  population  \
62  Lokoja   7.796882   6.740481  Nigeria   NG    Kogi   admin     60579.0
63    Kumo  10.048067  11.210555  Nigeria   NG   Gombe   minor     35712.0
64   Opobo   4.513882   7.537941  Nigeria   NG  Rivers   minor     34911.0
66   Dutse  11.756180   9.338959  Nigeria   NG  Jigawa   admin     17129.0
67    Orlu   5.795645   7.035126  Nigeria   NG     Imo   minor      9351.0

    population_proper
62            44722.0
63             2786.0
64            34911.0
66            17129.0
67             9351.0
```

b. Preprocessing and data cleaning.

The data frame was preprocessed and cleaned by removing NaN values, columns that of no use for the analysis were removed. Incomplete rows were equally deleted

I will drop the columns that are of no importance to my analysis.

[48]:
```
ng_df3 = ng_df2.drop(columns = ['country', 'iso2', 'admin', 'capital', 'population_proper'],
↪axis=0)
```

Check the new dataframe

[49]:
```
ng_df3.head()
```

[49]:
```
     city        lat       lng  population
0   Lagos   6.454066  3.394673   9466000.0
1    Kano  12.002381  8.513160   3140000.0
2  Ibadan   7.377562  3.905907   2628000.0
3   Abuja   9.083333  7.533333   1576000.0
4  Kaduna  10.526413  7.438795   1442000.0
```

[50]:
```
ng_df3.tail()
```

[50]:
```
      city        lat        lng  population
62  Lokoja   7.796882   6.740481     60579.0
63    Kumo  10.048067  11.210555     35712.0
64   Opobo   4.513882   7.537941     34911.0
66   Dutse  11.756180   9.338959     17129.0
67    Orlu   5.795645   7.035126      9351.0
```

Reindex the data frame to make the population the second index

[51]:
```
ng_df4 = ng_df3[['city', 'population', 'lat', 'lng']]

ng_df4.head()
```

[51]:
```
     city  population        lat       lng
0   Lagos   9466000.0   6.454066  3.394673
1    Kano   3140000.0  12.002381  8.513160
2  Ibadan   2628000.0   7.377562  3.905907
```

```
3   Abuja   1576000.0   9.083333  7.533333
4  Kaduna   1442000.0  10.526413  7.438795
```

[52]:
```
ng_df4.tail()
```

[52]:
```
      city  population        lat        lng
62  Lokoja     60579.0   7.796882   6.740481
63    Kumo     35712.0  10.048067  11.210555
64   Opobo     34911.0   4.513882   7.537941
66   Dutse     17129.0  11.756180   9.338959
67    Orlu      9351.0   5.795645   7.035126
```

Rename the last two columns for them to be aquate for folium to access

[53]:
```
ng_df4.rename(columns = {'lat':'latitude', 'lng':'longitude'}, inplace = True)
ng_df4.head()
```

[53]:
```
     city  population   latitude  longitude
0   Lagos   9466000.0   6.454066   3.394673
1    Kano   3140000.0  12.002381   8.513160
2  Ibadan   2628000.0   7.377562   3.905907
3   Abuja   1576000.0   9.083333   7.533333
4  Kaduna   1442000.0  10.526413   7.438795
```

2.1 C. Data Visualization

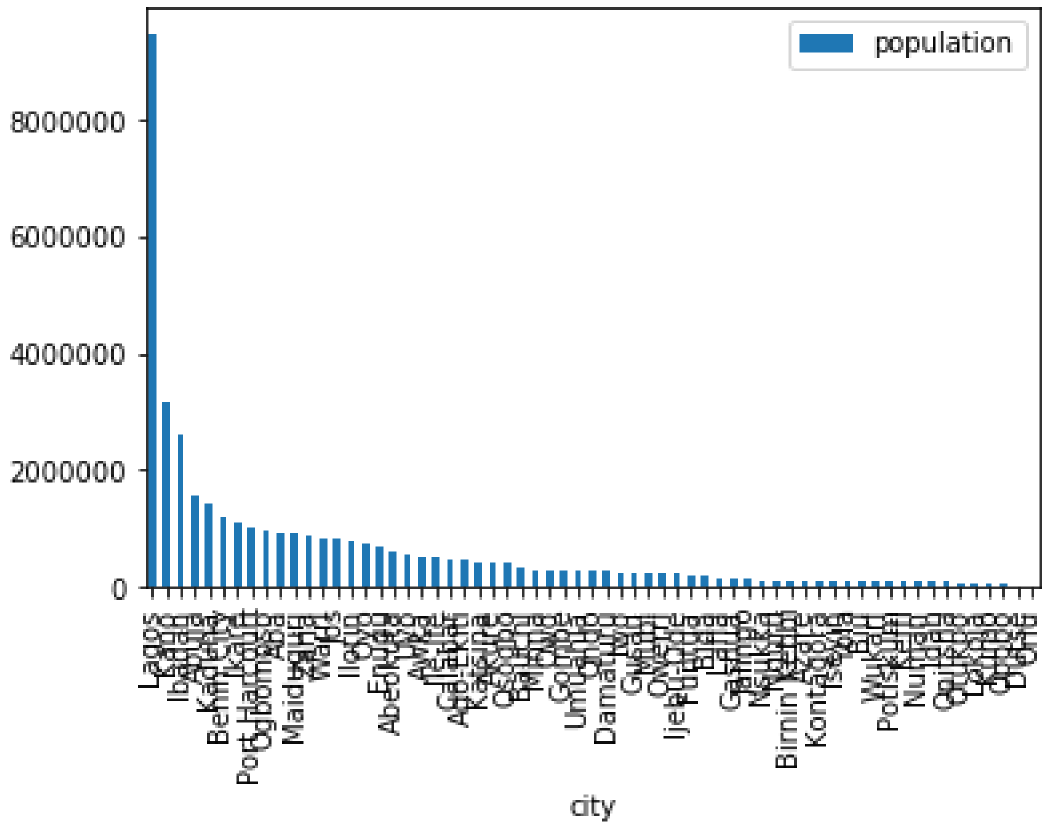
A preview of the emerged data frame was visualized with seaborn. The obtained graph will is as seen below.

A histogram of some Nigerian cities and their population

[54]:
```
import seaborn as sns

ng_df4.plot( x='city', y='population', kind='bar')
```

[54]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4ee034c240>
```



3 D. Geomapping and map creation using folium

The data frame was subjected to geo mapping after which the map of th two major cities of interest were created as below.

Geolocate the towns of interest and create a folium map

[55]:
```
address = 'Lagos, LG'

geolocator = Nominatim(user_agent="LG_explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
#print('The geographical coordinate of Lagos are {}, {}.'.format(latitude, longitude)

lagos_map = folium.Map(location=[latitude, longitude], zoom_start=11)

# add markers to map
for latitude, longitude, label in zip(ng_df4['latitude'], ng_df4['longitude'], ng_df4['city']):
    label = folium.Popup(label)
    folium.CircleMarker(
        [latitude, longitude],
        radius=5,
        popup=label,
        color='blue',
        fill_color='#3186cc',
        fill_opacity=0.7).add_to(lagos_map)

lagos_map
```

[55]:
```
<folium.folium.Map at 0x7f4ee1546f60>
```

Abuja is the capital of Nigeria and as such, should be of interest too, but, from the population figures, Lagos seems to be the business capital of Nigeria

[56]:
```
address = 'Abuja, AB'

geolocator = Nominatim(user_agent="NG_explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
#print('The geograpical coordinate of Lagos are {}, {}.'.format(latitude, longitude)

abuja_map = folium.Map(location=[latitude, longitude], zoom_start=11)

# add markers to map
for lat, lng, label in zip(ng_df4['latitude'], ng_df4['longitude'], ng_df4['city']):
    label = folium.Popup(label)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='blue',
        fill_color='#3186cc',
        fill_opacity=0.7).add_to(abuja_map)

abuja_map
```

[56]:
```
<folium.folium.Map at 0x7f4ee01cbe10>
```

Get the Foursquare credentials and print them

[57]:
```
CLIENT_ID = 'LV0Q3M0KCX402UG0VUHKHH02ZYFZM1MJ0S3IUIIXCSMOLEYU' #␣
  ↪your Foursquare ID
CLIENT_SECRET = 'QCSKDJ2QILQ01T1PIVHLCN0I1HAVDUGRHJ2XV5VYUYFQSYLS'␣
  ↪# your Foursquare Secret
VERSION = '20180605' # Foursquare API version

print('Your credentails:')
#print('CLIENT_ID: ' + CLIENT_ID)
#print('CLIENT_SECRET:' + CLIENT_SECRET)
```

Your credentails:

I will focus on the city of Lagos because of its large population as revealed by my data and plot

[58]:
```
# Get the lagos data
lagos_df = ng_df4.loc[ng_df4['city'] == 'Lagos']
lagos_df
```

[58]:
```
    city  population  latitude  longitude
0  Lagos   9466000.0  6.454066   3.394673
```

There is no need to reset index as Lagos already is at the top of the index

[59]:
```
neighborhood_latitude = lagos_df.loc[0]['latitude']
neighborhood_longitude = lagos_df.loc[0]['longitude']
```

Use my four square credentials to get the Lagos venues url

[60]:
```
LIMIT = 100 # limit of number of venues returned by Foursquare API

radius = 500 # define radius

 # create URL
url = 'https://api.foursquare.com/v2/venues/explore?
  ↪&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'.format(
    CLIENT_ID,
    CLIENT_SECRET,
    VERSION,
    neighborhood_latitude,
    neighborhood_longitude,
    radius,
    LIMIT)
#url # display URL
```

Get th json data

[61]:
```
results = requests.get(url).json()
#results
```

Extract the category of venues

[62]:
```
# function that extracts the category of the venue
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']
```

[63]:
```
venues = results['response']['groups'][0]['items']

#venues
```

[64]:
```
nearby_venues = json_normalize(venues) # flatten JSON
nearby_venues
```

[64]:
```
                  referralId  reasons.count  \
0  e-0-50ba3f00e4b0c78358420c87-0              0
1  e-0-4d1744d9816af04df50f4ec2-1              0
2  e-0-51463888e4b0ec732f11ee5b-2              0
3  e-0-500f3818e4b0bc9a81d55c76-3              0

                                      reasons.items  \
0  [{'summary': 'This spot is popular', 'type': '...
1  [{'summary': 'This spot is popular', 'type': '...
2  [{'summary': 'This spot is popular', 'type': '...
3  [{'summary': 'This spot is popular', 'type': '...

                   venue.id               venue.name venue.location.address  \
0  50ba3f00e4b0c78358420c87           Campos Stadium          Campos Street
1  4d1744d9816af04df50f4ec2           Sweet Sensation           27 Kakawa St
2  51463888e4b0ec732f11ee5b  The place where it began                    NaN
3  500f3818e4b0bc9a81d55c76              Geez Lounge          Lekki phase 1

   venue.location.lat  venue.location.lng  \
0            6.451098            3.394655
1            6.453180            3.391035
2            6.453686            3.394275
3            6.453056            3.395833

                         venue.location.labeledLatLngs  venue.location.distance  \
0  [{'label': 'display', 'lat': 6.451097614372146...                      330
1  [{'label': 'display', 'lat': 6.45318023620573,...                      414
2  [{'label': 'display', 'lat': 6.453685937293428...                       61
3  [{'label': 'display', 'lat': 6.4530556, 'lng':...                      170

  venue.location.cc venue.location.city venue.location.state  \
0                NG        Lagos Island                Lagos
1                NG               Lagos                Lagos
2                NG               Lagos                Lagos
3                NG               Lagos                Lagos

  venue.location.country                  venue.location.formattedAddress  \
0                Nigeria      [Campos Street, Lagos Island, Lagos, Nigeria]
1                Nigeria             [27 Kakawa St, Lagos, Lagos, Nigeria]
2                Nigeria                          [Lagos, Lagos, Nigeria]
3                Nigeria  [Lekki phase 1 (Opp pavilion lounge), Lagos, L...
```

```
                                   venue.categories  venue.photos.count  \
0  [{'id': '4cce455aebf7b749d5e191f5', 'name': 'S...                   0
1  [{'id': '4bf58dd8d48988d16e941735', 'name': 'F...                   0
2  [{'id': '4deefb944765f83613cdba6e', 'name': 'H...                   0
3  [{'id': '4bf58dd8d48988d121941735', 'name': 'L...                   0

  venue.photos.groups venue.location.crossStreet
0                  []                        NaN
1                  []                        NaN
2                  []                        NaN
3                  []        Opp pavilion lounge
```

Clean the json and structure it into a pandas dataframe.

[65]:
```
# filter columns
filtered_columns = ['venue.name', 'venue.categories', 'venue.location.lat', 'venue.location.lng']
nearby_venues = nearby_venues.loc[:, filtered_columns]

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

# clean columns
nearby_venues.columns = [col.split(".")[-1] for col in nearby_venues.columns]

nearby_venues
```

[65]:
```
                      name            categories       lat       lng
0           Campos Stadium          Soccer Field  6.451098  3.394655
1          Sweet Sensation  Fast Food Restaurant  6.453180  3.391035
2  The place where it began         Historic Site  6.453686  3.394275
3              Geez Lounge                Lounge  6.453056  3.395833
```

[66]:
```
print('{} venues were returned by Foursquare.'.format(nearby_venues.shape[0]))
```

```
4 venues were returned by Foursquare.
```

[67]:
```
def getNearbyVenues(names, latitudes, longitudes, radius=500):

    venues_list=[]
    for name, lat, lng in zip(names, latitudes, longitudes):
        print(name)

        # create the API request URL
        url = 'https://api.foursquare.com/v2/venues/explore?
 ↪&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'.format(
            CLIENT_ID,
            CLIENT_SECRET,
            VERSION,
            lat,
            lng,
            radius,
            LIMIT)

        # make the GET request
        results = requests.get(url).json()["response"]['groups'][0]['items']


         # return only relevant information for each nearby venue
        venues_list.append([(
            name,
            lat,
            lng,
            v['venue']['name'],
            v['venue']['location']['lat'],
            v['venue']['location']['lng'],
            v['venue']['categories'][0]['name']) for v in results])
       nearby_venues = pd.DataFrame([item for venue_list in venues_list for item in venue_list])
    nearby_venues.columns = ['Neighborhood',
                  'Neighborhood Latitude',
                  'Neighborhood Longitude',
                  'Venue',
                  'Venue Latitude',
                  'Venue Longitude',
                  'Venue Category']

    return(nearby_venues)
```

Foursquare returned four venues.

[68]:
```
lagos_df
```

[68]:
```
    city  population  latitude  longitude
0  Lagos   9466000.0  6.454066   3.394673
```

I will stop my analysis here because, as you can see, the url I used to generate my data did not list any neighborhoods of the cities. This sems to be a common problem. Please, go to the url below and you will understand my dilemma. https://wpgeodirectory.com/support/topic/setting-up-a-proper-neighborhood-map-url/

3.0.1 E. Results / Discussions

1. For someone looking to open a restaurant I will recommend Lagos city, around Campos standium.

Reasons:
Lagos has a large population and implicitly, a large population translates to higher clientele.

Lagos is situated by the sea and has seaports and airports and will have a lot of people in transit and tourism.

2. For a contractor trying to start his own business

I will still recommend lagos city for the same reasons proffered above, population. From the data and plot, Lagos city’s population is not comparable to any other city in Nigeria, in size and people will like to build either for renting our for personl use or both.

3.1 F. Conclusions

From the results of my analysis, I will recommend the city of Lagos in Nigeria as the most fertile business ground for someone planning to open restaurant as well as for an aspiring contractor. Its rich road network, seaports as well as airports and dense population is a plus for businesses.

Thank you.

[ ]:

```
```