# Coursera Capstone – REPORT

## Content

# 1. Introduction Section :

## Discussion of the business problem and the audience who would be interested in this project.

## 1.1 Scenario and Background

I am a data scientist currently residing in Downtown Singapore. I currently live within walking distance to Downtown "Tolko Ayer MRT metro station" therefore I have access to good public transportation to work. Likewise, I enjoy many amenities in the neighborhood, such as international cuisine restaurants, cafes, food shops and entertainment. I have been offered a great opportunity to work in Manhattan, NY. Although, I am very excited about it, I am a bit stress toward the process to secure a comparable place to live in Manhattan. Therefore, I decided to apply the learned skills during the Coursera course to explore ways to make sure my decision is factual and rewarding. Of course, there are alternatives to achieve the answer using available Google and Social media tools, but it rewarding doing it myself with learned tools.

## 1.2 Problem to be resolved:

The challenge to resolve is being able to find a rental apartment unit in Manhattan NY that offers similar characteristics and benefits to my current situation. Therefore, in order to set a basis for comparison, I want to find a rent a unit subject to the following conditions:

- Apartment with min 2 bedrooms with monthly rent not to exceed US$7000/month
- Unit located within walking distance (<=1.0 mile, 1.6 km) from a subway metro station in Manhattan
- Area with amenities and venues similar to the ones described for current location ( See item 2.1)

## 1.3 Interested Audience

I believe this is a relevant project for a person or entity considering moving to a major city in Europe, US or Asia, since the approach and methodologies used here are applicable in all cases. The use of FourSquare data and mapping techniques combined with data analysis will help resolve the key questions arisen. Lastly, this project is a good practical case toward the development of Data Science skills.

**Upload Libraries Required**

```
import numpy as np # library to handle data in a vectorized manner
import time
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
import requests # library to handle requests
from pandas.io.json import json_normalize # tranform JSON file into a pandas dataframe


!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

!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
import folium # map rendering library
from folium import plugins

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

import seaborn as sns

print('Libraries imported.')
Solving environment: done
```

```
## Package Plan ##

  environment location: /Users/Gerardo/ANACONDA2018/anaconda3

  added / updated specs:
    - geopy


The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    conda-4.5.12               |        py37_1000         652 KB  conda-forge

The following packages will be UPDATED:

    conda: 4.5.11-py37_1000 conda-forge --> 4.5.12-py37_1000 conda-forge


Downloading and Extracting Packages
conda-4.5.12         | 652 KB    | ######################################## | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
Solving environment: done

# All requested packages already installed.

Libraries imported.
```

## 2. Data Section:

# Description of the data and its sources that will be used to solve the problem

## 2.1 Data of Current Situation

I Currently reside in the neighborhood of 'Mccallum Street' in Downtonw Singapore. I use Foursquare to identify the venues around the area of residence which are then shown in the Singapore map shown in methodology and execution in section 3.0. It serves as a reference for comparison with the desired future location in Manhattan NY

## 2.2 Data Required to resolve the problem

In order to make a good choice of a similar apartment in Manhattan NY, the following data is required: List/Information on neighborhoods form Manhattan with their Geodata ( latitude and longitude. List/Information about the subway metro stations in Manhattan with geodata. Listed apartments for rent in Manhattan area with descriptions ( how many beds, price, location, address) Venues and amenities in the Manhattan neighborhoods (e.g. top 10) 2.3 sources and manipulation The list of Manhattan neighborhoods is worked out during Lab exercise during the course. A csv file was created which will be read in order to create a data frame and its mapping. The csv file 'mh_neigh_data.csv' has the following below data structure. The file will be directly read to the Jupiter Notebook for convenience and space savings. The clustering of neighborhoods and mapping will be shown however. An algorithm was used to determine the geodata from Nominate. The actual algorithm coding may be shown in 'markdown' mode because it takes time to run.

mh_neigh_data.tail():

```
    Borough    Neighborhood    Latitude    Longitude

35    Manhattan    Turtle Bay    40.752042    -73.967708

36    Manhattan    Tudor City    40.746917    -73.971219

37    Manhattan    Stuyvesant Town    40.731000    -73.974052
38    Manhattan    Flatiron    40.739673    -73.990947

39    Manhattan    Hudson Yards    40.756658    -74.000111
```

A list of Manhattan subway metro stops was complied in Numbers (Apple excel) and it was complemeted with Wikipedia data. ( https://en.wikipedia.org/wiki/List_of_New_York_City_Subway_stations_in_Manhattan) and information from NY Transit authority and Google maps (https://www.google.com/maps/search/manhattan+subway+metro+stations/@40.7837297,-74.1033043,11z/data=!3m1!4b1)

for a final consolidated list of subway stops names and their address. The geolocation was obtained via an algorithm using Nominate. Details will be shown in the execution of methodology in section 3.0. The subway csv file is "MH_subway.csv'" and the data structure is: mhsub.tail(): sub_station sub_address lat long

A list of places for rent was collected by web-browsing real estate companies in Manhattan : http://www.rentmanhattan.com/index.cfm?page=search&state=results https://www.nestpick.com/search?city=new-york&page=1&order=relevance&district=manhattan&gclid=CjwKCAiAjNjgBRAgEiwAGLlf2hkP3A-cPxjZYkURqQEswQK2jKQEpv_MvKcrIhRWRzNkc_r-

fGi0lxoCA7cQAvD_BwE&type=apartment&display=list https://www.realtor.com/apartments/Manhattan_NY

A csv file was compiled with the rental place that indicated: areas of Manhattan, address, number of beds, area and monthly rental price. The csv file "nnnn.csv" had the following below structure. An algorythm was used to create all the geodata using Nominatim, as shown in section 3.0. The actual algorythm coding may be shown in 'markdown' mode becasues it takes time to run. With the use of geolocator = Nominatim() , it was possible to determine the latitude and longiude for the subway metro locations as well as for the geodata for each rental place listed. The loop algorythms used are shown in the execution of data in section 3.0 "Great_circle" function from geolocator was used to calculate distances between two points , as in the case to calculate average rent price for units around each subway station and at 1.6 km radius. Foursquare is used to find the avenues at Manhattan neighborhoods in general and a cluster is created to later be able to search for the venues depending of the location shown.

## 2.4 How the data will be used to solve the problem

The data will be used as follows: Use Foursquare and geopy data to map top 10 venues for all Manhattan neighborhoods and clustered in groups ( as per Course LAB) Use foursquare and geopy data to map the location of subway metro stations , separately and on top of the above clustered map in order to be able to identify the venues and ammenities near each metro station, or explore each subway location separately Use Foursquare and geopy data to map the location of rental places, in some form, linked to the subway locations. create a map that depicts, for instance, the average rental price per square ft, around a radious of 1.0 mile (1.6 km) around each subway station - or a similar metrics. I will be able to quickly point to the popups to know the relative price per subway area. Addresses from rental locations will be converted to geodata( lat, long) using Geopy-distance and Nominatim. Data will be searched in open data sources if available, from real estate sites if open to reading, libraries or other government agencies such as Metro New York MTA, etc.

## 2.5 Mapping of Data

The following maps were created to facilitate the analysis and the choice of the palace to live. Manhattan map of Neighborhoods manhattan subway metro locations Manhattan map of places for rent Manhattan map of clustered venues and neighborhoods Combined maps of Manhattan rent places with subway locations Combined maps of Manhattan rent places with subway locations and venues clusters

## 3. Methodology section:

This section represents the main component of the report where the data is gathered, prepared for analysis. The tools described are used here and the Notebook cells indicates the execution of steps.

**MAPPING DATA**

## Singapore Map - Current residence and venues in neighborhood

for comparison to future Manhattan renting place

In [3]:

```
# Shenton Way, District 01, Singapore
address = 'Mccallum Street, Singapore'
geolocator = Nominatim()
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print('The geograpical coordinate of Singapore home are {}, {}.'.format(latitude, lo
ngitude))
```

```
/Users/Gerardo/ANACONDA2018/anaconda3/lib/python3.7/site-packages/ipykernel_launcher
.py:3: DeprecationWarning: Using Nominatim with the default "geopy/1.18.1" `user_age
nt` is strongly discouraged, as it violates Nominatim's ToS https://operations.osmfo
undation.org/policies/nominatim/ and may possibly cause 403 and 429 HTTP errors. Ple
ase specify a custom `user_agent` with `Nominatim(user_agent="my-application")` or b
y overriding the default `user_agent`: `geopy.geocoders.options.default_user_agent =
 "my-application"`. In geopy 2.0 this will become an exception.
  This is separate from the ipykernel package so we can avoid doing imports until
```

```
The geograpical coordinate of Singapore home are 1.2792655, 103.8480938.
```

In [4]:

```
neighborhood_latitude=1.2792655
neighborhood_longitude=103.8480938
```

## Dial FourSquare to find venues around current residence in Singapore

```
# @hidden_cell
CLIENT_ID = 'DVCxxxxxxxxxxxxxxxxxxxxC0CFLF1T' # your Foursquare ID
CLIENT_SECRET = '5NWAGyyyyyyyyyyyyyyyyyyyyyyyyLFWL1' # your Foursquare Secret
VERSION = '2xxxxxxxxxxxxxxxxxx5' # Foursquare API version

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

In [6]:

```
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
```

Out[6]:

```
'https://api.foursquare.com/v2/venues/explore?&client_id=DVCTZDPDYXTS0BRJFPLMHM323AP
GXNWZI5PLRQ1VC0CFLF1T&client_secret=5NWAGXRLXIXAV0L3DNYY1EPIHNMAAAIZFDFELYSYXL5LFWL1
&v=20180605&ll=1.2792655,103.8480938&radius=500&limit=100'
```

In [7]:

```
# results display is hidden for report simplification
results = requests.get(url).json()
```

***function that extracts the category of the venue - borrow from the Foursquare lab***

```
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']
```

In [9]:

```
venues = results['response']['groups'][0]['items']
SGnearby_venues = json_normalize(venues) # flatten JSON
# filter columns
filtered_columns = ['venue.name', 'venue.categories', 'venue.location.lat', 'venue.location.lng']
SGnearby_venues =SGnearby_venues.loc[:, filtered_columns]
# filter the category for each row
SGnearby_venues['venue.categories'] = SGnearby_venues.apply(get_category_type, axis=1)
# clean columns
SGnearby_venues.columns = [col.split(".")[-1] for col in SGnearby_venues.columns]

SGnearby_venues.shape
```

Out[9]:

```
(100, 4)
```

## The analysis and the strategy:

The strategy is based on mapping the above described data in section 2.0, in order to facilitate the choice of at least two candidate places for rent. The choice is made based on the demands imposed: location near a subway, rental price and similar venues to Singapore. This visual approach and maps with popup labels allow quick identification of location, price and feature, thus making the selection very easy.

The processing of these DATA and its mapping will allow answering the key questions to make a decision:

- What is the cost of available rental places that meet the demands?

- What is the cost of rent around a mile radius from each subway metro station?
- What is the area of Manhattan with best rental pricing that meets criteria established?
- What is the distance from work place (Park Ave and 53 rd St) and the tentative future rental home?
- What are the venues of the two best places to live? How the prices compare?
- How venues distribute among Manhattan neighborhoods and around metro stations?
- Are there tradeoffs between size and price and location?
- Any other interesting statistical data findings of the real estate and overall data.

## Algorithm to find latitude and longitude for each subway metro station and add them to data frame. This coding has been 'Markdown' just to simplify the file report, and the .csv file will be read in cell below.

```
for n in range

(len(mh)): address= mh['sub_address'][n] geolocator = Nominatim() location =
geolocator.geocode(address) latitude = location.latitude longitude = location.longitude
mh['lat'][n]=latitude mh['long'][n]=longitude
    #print(n,latitude,longitude)
    time.sleep(2)

print('Geodata completed')
```

## 4.0 Results

## ONE CONSOLIDATE MAP

**Let's consolidate all the required information to make the apartment selection in one map**

**Map of Manhattan with rental places, subway locations and cluster of venues**

**Red dots are Subway stations, Blue dots are apartments available for rent, Bubbles are the clusters of venues**

```
# create map of Manhattan using latitude and longitude values from Nominatim
latitude= 40.7308619
longitude= -73.9871558

map_mh_one = folium.Map(location=[latitude, longitude], zoom_start=13.3)

# add markers to map
for lat, lng, label in zip(mh_rent['Lat'], mh_rent['Long'],'$ ' + mh_rent['Rent_Price'].astype(str)+ ', '+mh_rent['Address']):
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=6,
        popup=label,
        color='blue',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.7,
        parse_html=False).add_to(map_mh_one)

    # add markers of subway locations to map
for lat, lng, label in zip(mhsub1['lat'], mhsub1['long'],  mhsub1['sub_station'].astype(str) ):
    label = folium.Popup(label, parse_html=True)
    folium.RegularPolygonMarker(
        [lat, lng],
        number_of_sides=6,
        radius=6,
        popup=label,
        color='red',
        fill_color='red',
        fill_opacity=2.5,
    ).add_to(map_mh_one)


# set color scheme for the clusters
kclusters=5
x = np.arange(kclusters)
ys = [i+x+(i*x)**2 for i in range(kclusters)]
colors_array = cm.rainbow(np.linspace(0, 1, len(ys)))
rainbow = [colors.rgb2hex(i) for i in colors_array]

# add markers to the map
markers_colors = []
```

```
for lat, lon, poi, cluster in zip(manhattan_merged['Latitude'], manhattan_merged['Longitude'], manhattan_merged['Neighborhood'], manhattan_merged['Cluster Labels']):
    label = folium.Popup(str(poi) + ' Cluster ' + str(cluster), parse_html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=15,
        popup=label,
        color=rainbow[cluster-1],
        fill=True,
        fill_color=rainbow[cluster-1],
        fill_opacity=0.7).add_to(map_mh_one)

    # Adds tool to the top right
from folium.plugins import MeasureControl
map_mh_one.add_child(MeasureControl())

# Measurement ruler icon tool to measure distances in map
from folium.plugins import FloatImage
url = ('https://media.licdn.com/mpr/mpr/shrinknp_100_100/AAEAAQAAAAAAAAlgAAAAJGE3OTA4YTdlLTkzZjUtNDFjYy1iZThlLWQ5OTNkYzlhNzM4OQ.jpg')
FloatImage(url, bottom=5, left=85).add_to(map_mh_one)

map_mh_one
```

## Apartment Selection

Using the "one map" above, I was able to explore all possibilities since the popups provide the information needed for a good decision. Apartment 1 rent cost is US7500 slightly above the US7000 budget. Apt 1 is located 400 meters from subway station at 59th Street and work place ( Park Ave and 53rd) is another 600 meters way. I can walk to work place and use subway for other places around. Venues for this apt are as of Cluster 2 and it is located in a fine district in the East side of Manhattan. Apartment 2 rent cost is US6935, just under the US7000 budget. Apt 2 is located 60 meters from subway station at Fulton Street, but I will have to ride the subway daily to work , possibly 40-60 min ride. Venues for this apt are as of Cluster 3. Based on current Singapore venues, I feel that Cluster 2 type of venues is a closer resemblance to my current place. That means that APARTMENT 1 is a better choice since the extra monthly rent is worth the conveniences it provides.

## 5.0 DISCUSSION

In general, I am positively impressed with the overall organization, content and lab works presented during the Coursera IBM Certification Course I feel this Capstone project presented me a great opportunity to practice and apply the Data Science tools and methodologies learned. I have created a good project that I can present as an example to show my potential. I feel I have acquired a good starting point to become a professional Data Scientist and I will continue exploring to creating examples of practical cases.

## 6.0 CONCLUSIONS

I feel rewarded with the efforts, time and money spent. I believe this course with all the topics covered is well worth of appreciation. This project has shown me a practical application to resolve a real situation that has impacting personal and financial impact using Data Science tools. The mapping with Folium is a very powerful technique to consolidate information and make the analysis and decision thoroughly and with confidence. I would recommend for use in similar situations. One must keep abreast of new tools for DS that continue to appear for application in several business fields.