# DO YOU WANT TO BE RICH? FOLLOW THE MONEY!

*(Coursera Data Science Capstone)*

*1st January 2021*

# 1. Introduction

## 1.1. Background

It is always believed that if you want to get rich you should associate with the rich, spend more money, get more! If you want to get lucky you should always live in an expensive city where there are lots of opportunities including getting high paying jobs and getting married to the rich (Ohh, yes!). If you live in an expensive neighbourhood, you are likely to bump into diplomats, successful business men and that's how your network will grow richer. New York City is one of those cities that will elevate your dreams of getting rich. New York is an expensive and very famous city that most people dream to live in. New York City comprises 5 boroughs namely The Bronx, Manhattan, Brooklyn, Queens, Staten Island. To make it better, more than 800 languages are spoken in New York City (WorldStrides, 2017), making it the most linguistically diverse city in the world and for this matter attract not only americans but most foreigners too.

## 1.2. Problem

Relocating to a new place is a challenge especially if you have never been there before. Therefore, this project intends to find an expensive borough in the New York city.

## 1.3. Target Audience

This project targets people who believe that richness is happiness and wants to follow where the money is by living in an expensive neighbourhood in the New York city.

# 2. Data

## 2.1. Data Sources

This project uses a foursquare dataset for the location boroughs in the New York city. In addition to that the project uses New York city housing price data which is obtained from Kaggle which contains apartments sold in the past 12 months.

Housing sales data obtained from kaggle
https://www.kaggle.com/new-york-city/nyc-property-sales

## 2.2. Data Cleaning

The data was downloaded from two sources, foursquare database and Kaggle.

Part I

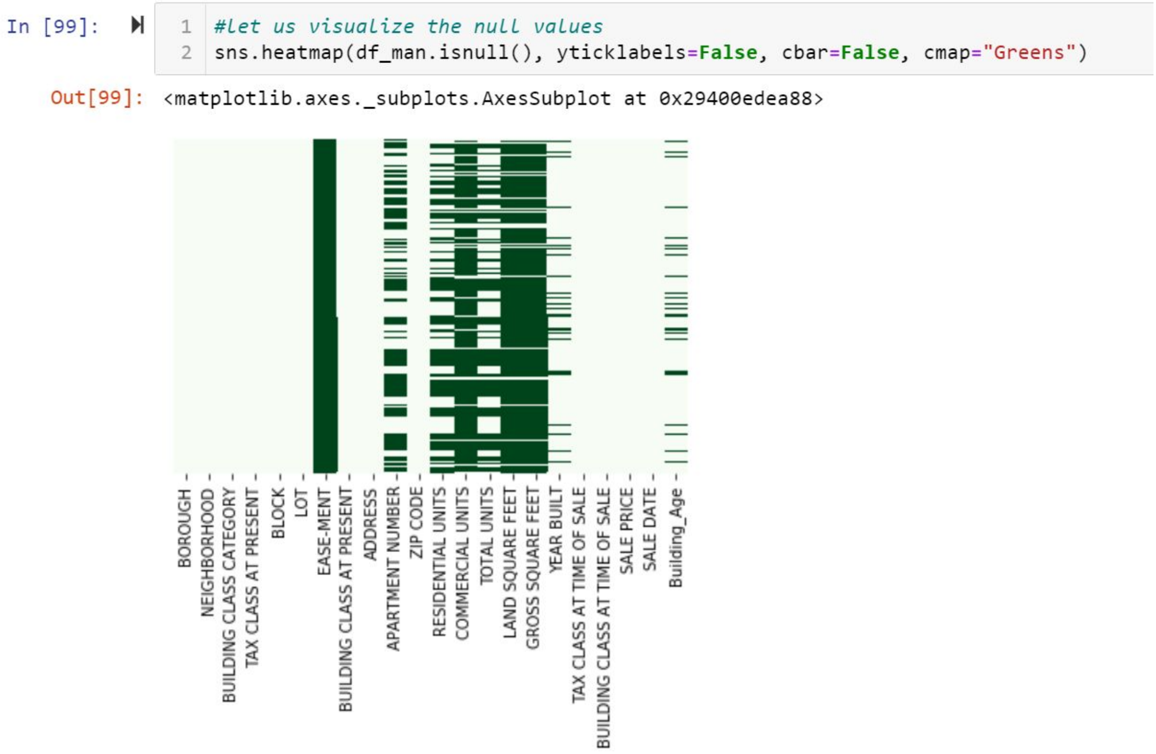
The dataset of New York city from kaggle was cleaned as follows;

- Duplicate column for index field which was dropped
- Removed rows with sale price between 0 to $10,000 as they dont make sense. An assumption was made to be an error in the data entry. This assumption is made based on observation of housing prices in the New York city from 2012.
- The Borough codes were renamed to appropriate names so that it could make sense during the visualisation for instance Borough “1” was named to Manhattan and so on.

Part II

- Foursquare data was clean, we used it together with New York city data to draw choropleth map

Part III



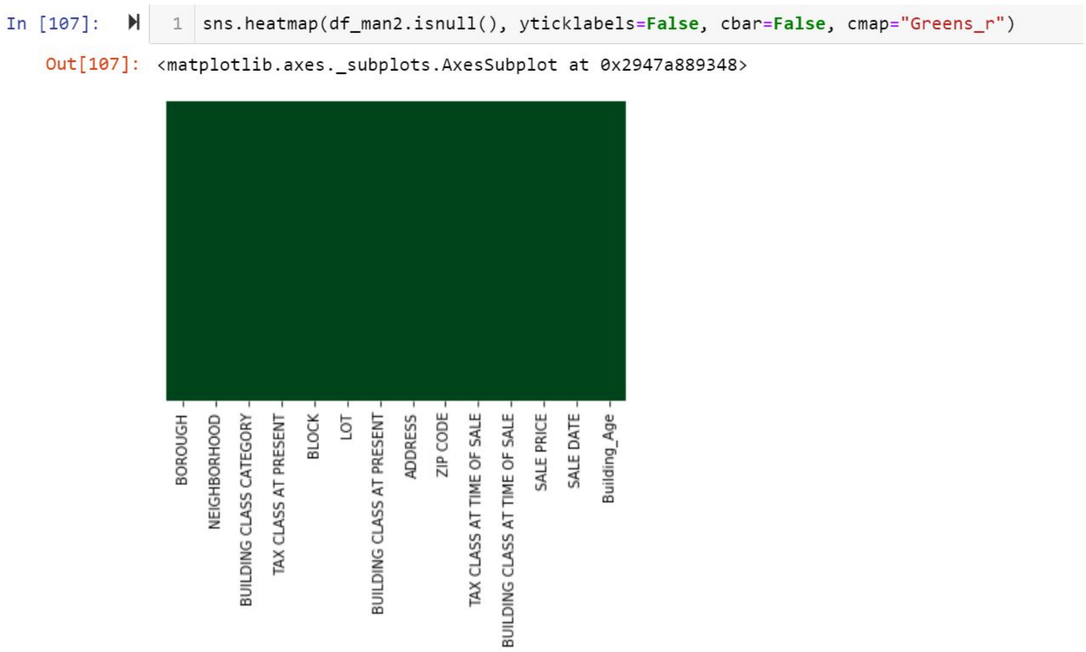
```
In [99]:  #let us visualize the null values
           sns.heatmap(df_man.isnull(), yticklabels=False, cbar=False, cmap="Greens")

Out[99]: <matplotlib.axes._subplots.AxesSubplot at 0x29400edea88>
```

**Figure 1: The dataset prior to data cleaning( with null values)**

The dataset of Manhattan from Kaggle was cleaned as follows

- The first four rows were dropped as they contained information about the dataset that was not needed in the working database. The preceding information blocked the title column to be recognised.

- Removed rows with sale price between 0 to $10,000 as they dont make sense. An assumption was made to be an error in the data entry. This assumption is made based on observation of housing prices in the New York city from 2012.
- The columns with null values more than 50% of the data were dropped so as to avoid generating inaccurate data.
- The columns with fewer null values were filled in by mode for categorical variables and mean for numerical values.



```
In [107]:  sns.heatmap(df_man2.isnull(), yticklabels=False, cbar=False, cmap="Greens_r")
Out[107]: <matplotlib.axes._subplots.AxesSubplot at 0x2947a889348>
```

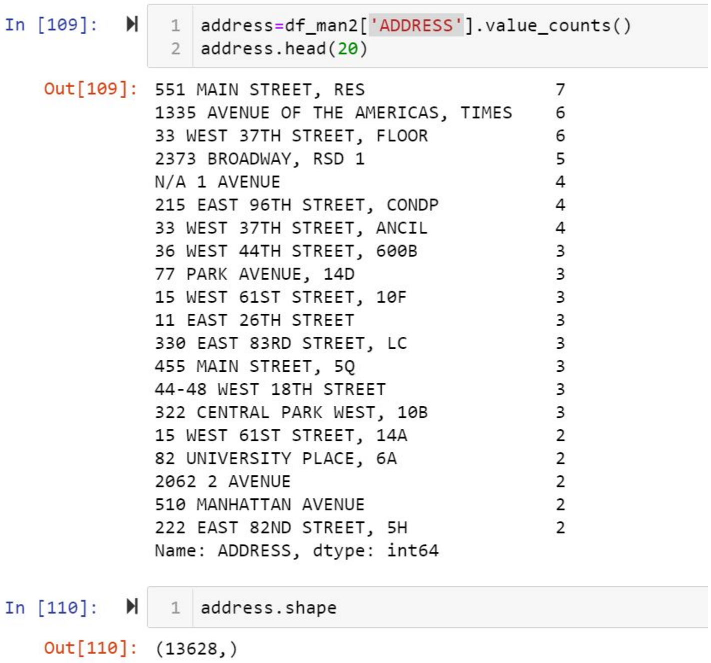
**Figure 2: Dataset after removing null values**

# 3. Methodology

## 3.1. Exploratory Analysis

For the Part III of the task which was predicting housing sales in Manhattan, exploratory analysis performed as follows;

- Address column was dropped as it contained very specific information that could rather make our model heavy less accurate



```
In [109]:  address=df_man2['ADDRESS'].value_counts()
           address.head(20)

Out[109]:  551 MAIN STREET, RES                  7
           1335 AVENUE OF THE AMERICAS, TIMES    6
           33 WEST 37TH STREET, FLOOR            6
           2373 BROADWAY, RSD 1                  5
           N/A 1 AVENUE                          4
           215 EAST 96TH STREET, CONDP           4
           33 WEST 37TH STREET, ANCIL            4
           36 WEST 44TH STREET, 600B             3
           77 PARK AVENUE, 14D                   3
           15 WEST 61ST STREET, 10F              3
           11 EAST 26TH STREET                   3
           330 EAST 83RD STREET, LC              3
           455 MAIN STREET, 5Q                   3
           44-48 WEST 18TH STREET                3
           322 CENTRAL PARK WEST, 10B            3
           15 WEST 61ST STREET, 14A              2
           82 UNIVERSITY PLACE, 6A               2
           2062 2 AVENUE                         2
           510 MANHATTAN AVENUE                  2
           222 EAST 82ND STREET, 5H              2
           Name: ADDRESS, dtype: int64

In [110]:  address.shape

Out[110]:  (13628,)
```

Figure 3: Address column with less groupings and a shape of 13628 rows which means the feature is very specific considering the whole dataset has 13916 rows.

- Year Built was dropped and instead replaced by a computed feature which is building age by subtracting current year and the Year Built
- Sale Date was replaced by Month of Sale because it is more general and we could group more data with it

## 3.2. Feature Engineering

A column of categorical features was created and then passed into a function (category_onehot_multcol) to convert the numerical values. Predicting housing sales is a regression problem thus we need to convert the categorical variables to numerical values to simplify the prediction process and improve accuracy. Afterwards duplicates were removed.

## 3.3. Model Selection

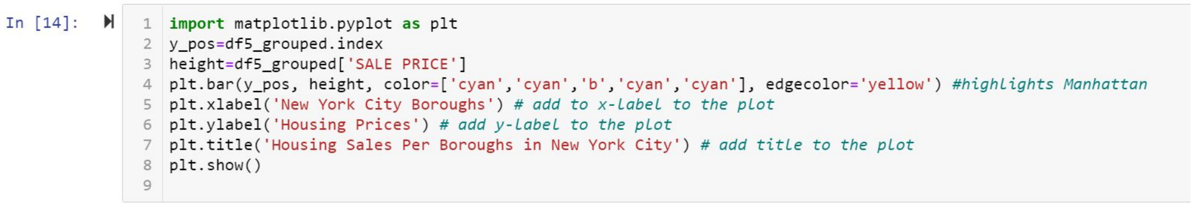
Housing sales prediction is a common regression machine learning problem since our target variables are numerical values. As we wanted to predict a housing price based on a time series data, therefore we decided to try out two models which are linear regression and random forest. Thereafter results were compared to find out which model performed better. The evaluation methods used were mean squared error and Coefficient Determination. Linear regression resulted in 0.27 Coefficient Determination while Random Forest resulted in 0.66 Coefficient Determination. Thus, a random forest model was selected.

# 4. Results and Discussion

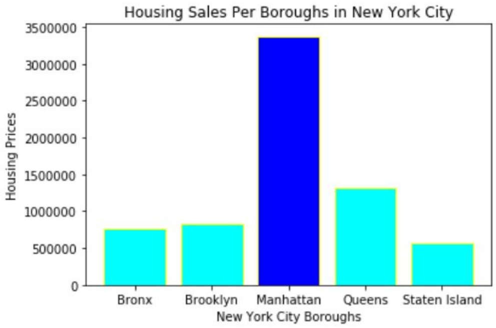
### Part I

The first part of the project was trying to identify which borough has a high living standard using housing sales price as a determinant . It should be noted in the real world scenario more determinants could be used. Based on that we found out that Manhattan was the most expensive borough in the New York city.

**visualize the housing sales prices in a bar graph**

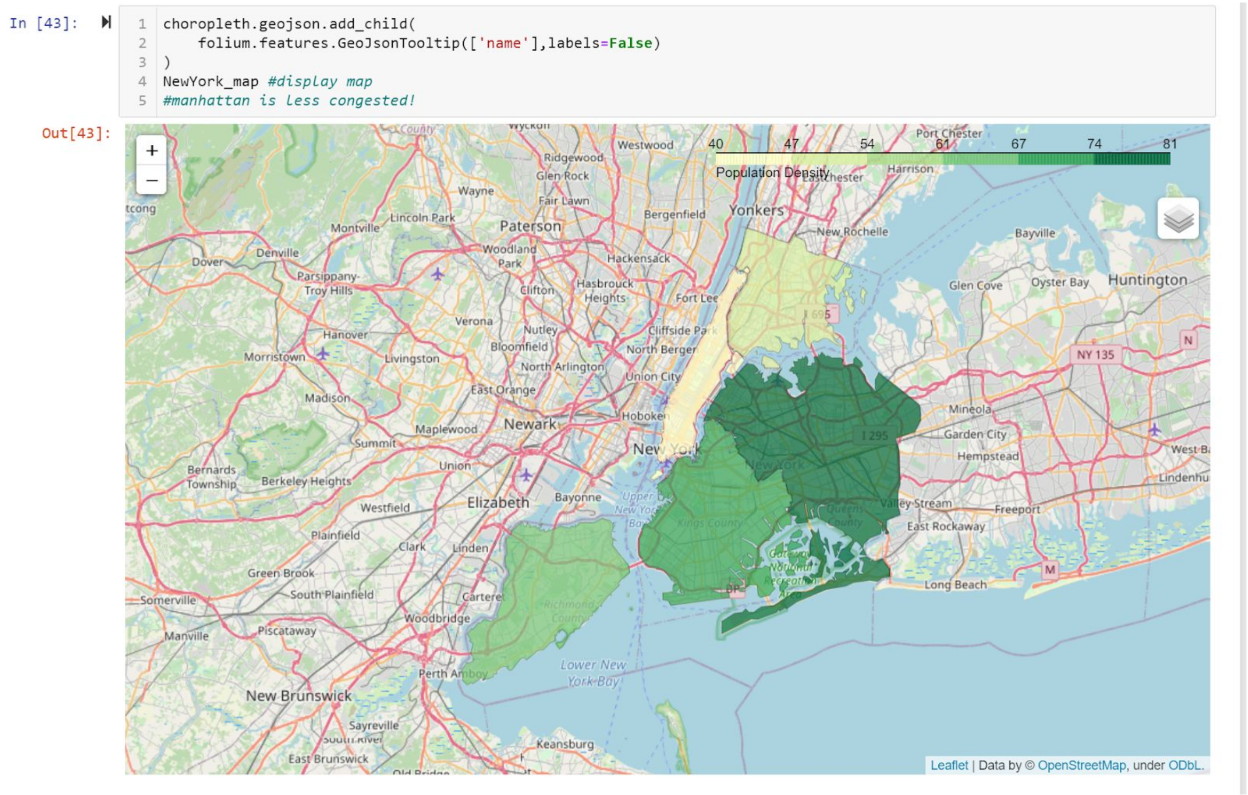


```python
import matplotlib.pyplot as plt
y_pos=df5_grouped.index
height=df5_grouped['SALE PRICE']
plt.bar(y_pos, height, color=['cyan','cyan','b','cyan','cyan'], edgecolor='yellow') #highlights Manhattan
plt.xlabel('New York City Boroughs') # add to x-label to the plot
plt.ylabel('Housing Prices') # add y-label to the plot
plt.title('Housing Sales Per Boroughs in New York City') # add title to the plot
plt.show()

```



### Part II

Part II of the project we used foursquare geospatial data in combination with the New York city data to visualise neighbourhood in the New York city and we found out that Manhattan has less neighbourhood and is bordered by ocean on the east, west and south part. For those who want to live by the beach they could locate an apartment on the east, west or south part of the city.



```
choropleth.geojson.add_child(
    folium.features.GeoJsonTooltip(['name'],labels=False)
)
NewYork_map #display map
#manhattan is less congested!
```

**Part III**

In part three we used a machine learning model to predict housing prices and found out that random forest performed better than linear regression. Random forest has also less mean squared error which makes it a better model in this case. Random forest had 0.66 coefficient determination while linear regression has 0.27 coefficient determination. That means the Random Forest model will produce results with better accuracy compared to linear regression. However, in a real world environment, feature optimization will need to be performed and thorough data analysis to be able to get a better score.

# 5. Conclusion

In this project I developed a model to predict housing sales in Manhattan after discovering that Manhattan was the most expensive city in New York. The objective was to identify boroughs with high living standards and I used housing price as a determinant. I understand there are other determinants that could be used, but for the sake of this assignment and time limitation I decided to limit into inme factor. Machine learning model ws therefore developed to assist our target audience to determine the price of houses based on different factors such as building age, building class, neighbourhood etc. However, I believe that if I could have enough data on land square feet I could have built a better model.

# 6. References

WorldStrides. *WorldStrides*, 2017, https://worldstrides.com/blog/2017/10/12-interesting-facts-about-new-york-city/. Accessed 31st December 2020.