

# 'THE BEST WALKABLE SPOTS IN NEW YORK'

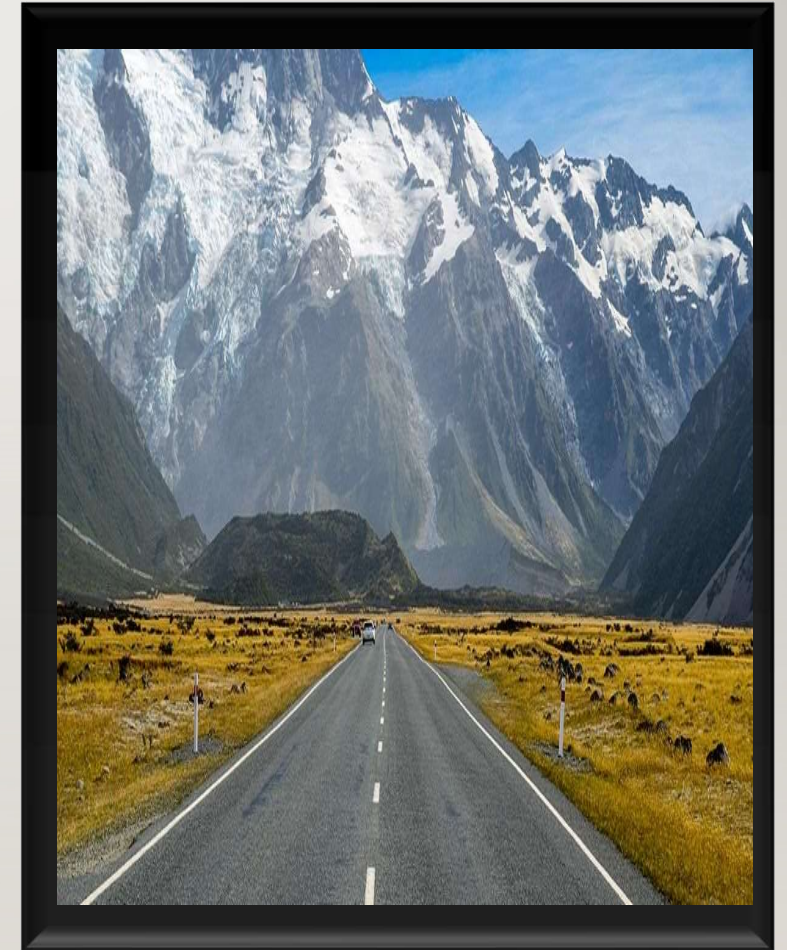
---

CAPSTONE PROJECT

FINAL REPORT

# PLAN

- ▶ **Business Problem Definition**
  - ▶ **Data Sources**
  - ▶ **Data Processing**
  - ▶ **Methodology**
  - ▶ **Analysis - Exploration**
  - ▶ **Visual Exploration**
  - ▶ **Clustering**
  - ▶ **Results**
  - ▶ **Conclusions & Perspectives**
- 



# BUSINESS PROBLEM DEFINITION

**Problem** : « I want to live in New York City ! »

## ► Requirements

- Everything is nearby
- No need for a vehicle
- Workplaces are numerous
- Apartment Rents are low

## ► Audience : who may be Interested ?

- Students that would like to find a job close to the place of residence
- People who want to live without cars
- Recommendation tools writers
- Etc.

➤ **Question** :

***Can you facilitate my quest ?***





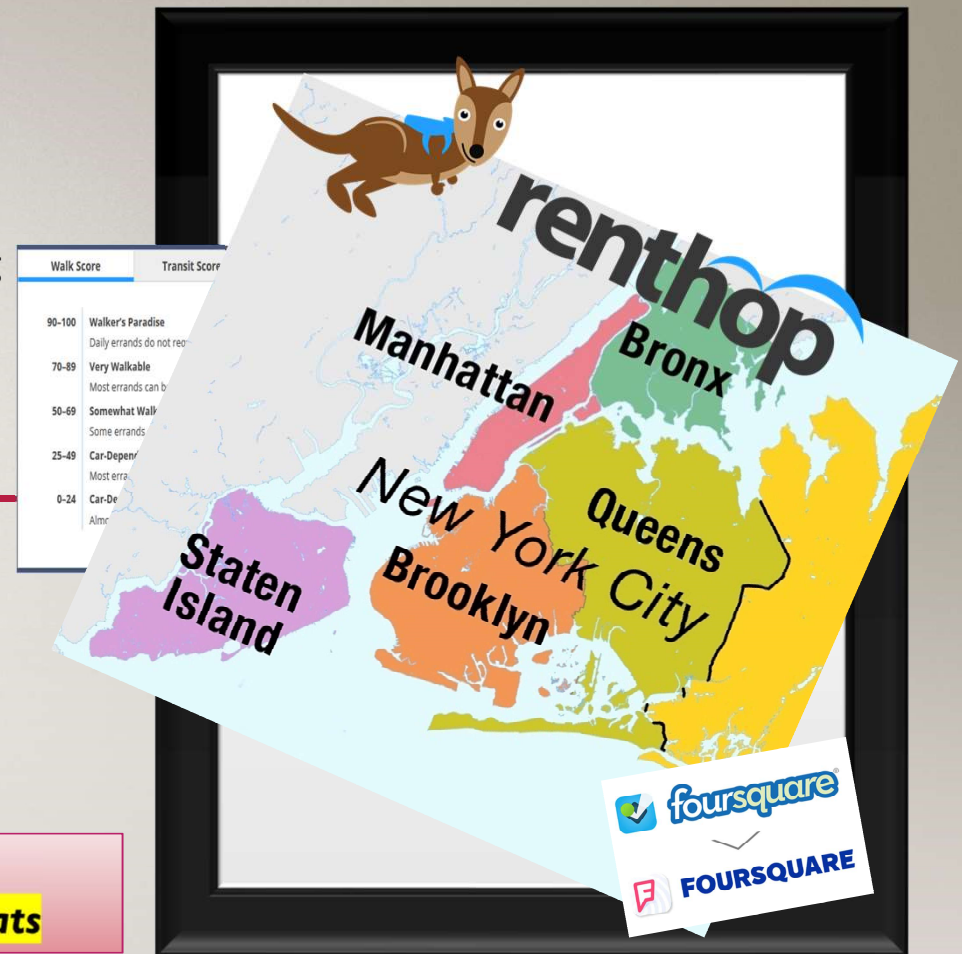
# DATA SOURCES

## New York City Information Retrieval

- ▶ **Compositon : 5 Boroughs and 306 Neighborhoods**
  - ▶ [https://geo.nyu.edu/catalog/nyu\\_2451\\_34572](https://geo.nyu.edu/catalog/nyu_2451_34572) :Json data parsing
- ▶ **Venues : 11 311 retrieved initially**
  - ▶ Foursquare :APIs
- ▶ **Walkability Scores : 5 levels [car-dependants, Walker's Paradise]**
  - ▶ Web Site "Walk Score" (<https://www.walkscore.com/>) :APIs
- ▶ **Average Rental Price : for Studio, 1BR by Neighborhoods**
  - ▶ Web Site "renthop" (<https://www.renthop.com/>) : HTML parsing

### Data Profiles:

4 data sets, 2 APIs, 2 files, JSON and HTML formats



# DATA PROCESSING

## Input : Raw Data

### ► Merging

- All data : 306 neighborhoods, 11300 venues, Walk Scores, Rentals

### ► Cleaning

- Dropping or replacing unknown or no consistent values

### ► Rewriting

- Correct data formatting : '\$4,542 -> 4542'

### ► Completing

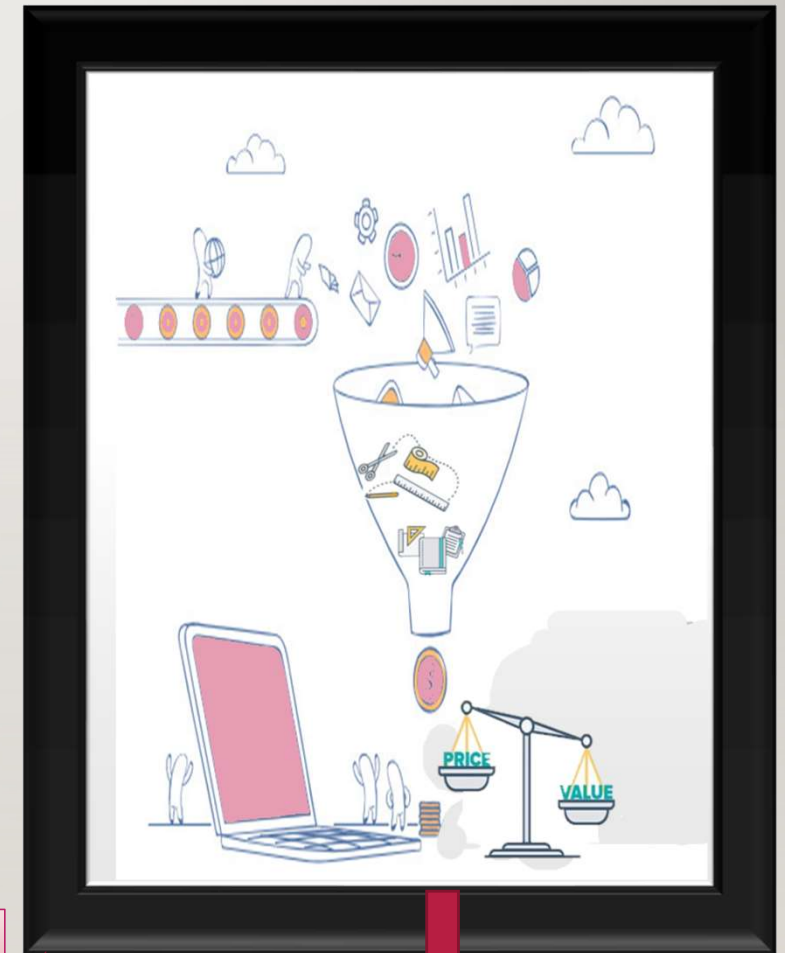
- Complete missing fields by calculation if possible (IBR budget with mean or ratio etc.)
- Get data from other sources (missing neighborhoods ranked by RentHop), etc.

### ► Renaming

- Avoid confusion : rename if it makes sense or drop

### ► Remapping

- Consistency : rescale categorical (i.e. Budget)



## Output :

**81 NeighBorhoods, 3508 venues, Walk Scores and Average Prices**

# METHODOLOGY

Follow the way !

## ► Step 1

- Data Set build and mapped in *pandas* Data Frames ready for analysis

## ► Step 2

- Exploration and Analysis

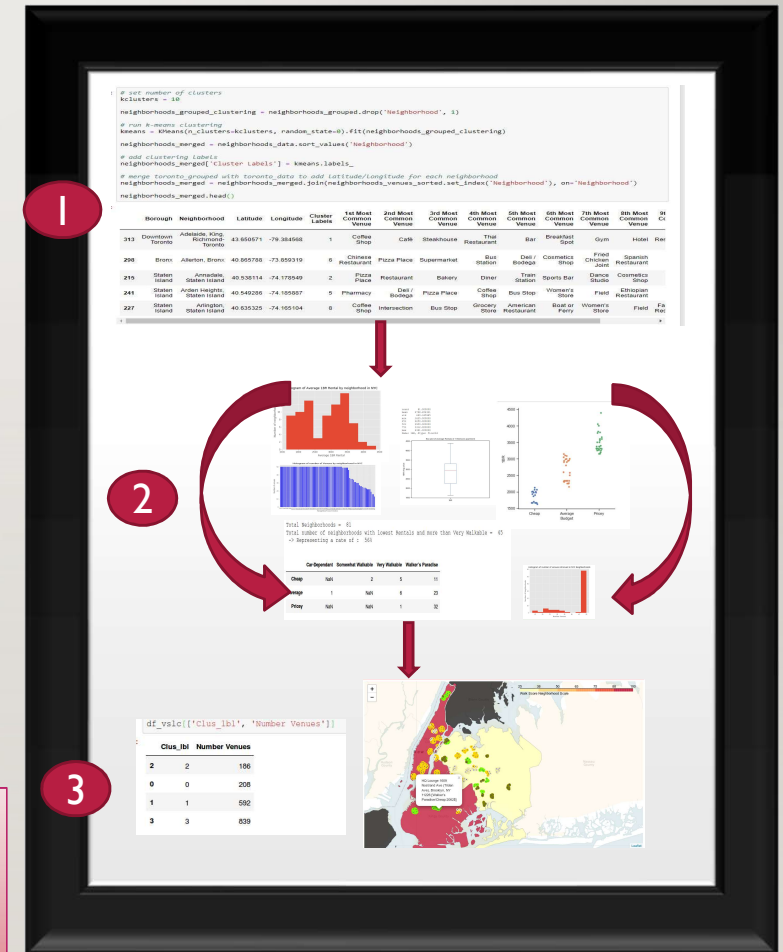
- Calculation : means, densities etc.
- Data visualization (histograms, bar plot, maps, heatmaps etc.)

## ► Step 3

- Selection of Areas
- Modeling Locations Clustering

## Results

Display Maps and Interactive Heatmaps indicating locations :  
**'Best Walkable Spots in New York City within a 500m radius'**





# ANALYSIS - EXPLORATION

Extracted Data : 81 Neighborhoods, 3508 Venues

► NYC Neighborhoods and Venues are 'Highly Walkable' > 97%

► Very Walkable or Walker's Paradise ~ 97%

► NYC Neighborhoods : High Density of Potential Workplaces

► Maximum Venues : 50 venues in 87% of Neighborhoods

► NYC 'IBR' Prices represent a High Budget

► Min = 1625, Max = Mean = 2798, Median = 2950

## Results

Easy to find Areas in NYC with High Walkability and Potential Workplace but Expensive !



# VISUAL EXPLORATION

## Maps

- ▶ **NYC Neighborhoods Map Localisation**
  - ▶ Neighborhoods are displayed with colors according to category
- ▶ **NYC Venues Map Localisation**
  - ▶ Venues are also displayed with same colors as Neighborhoods

## Heatmaps

- ▶ **NYC Venues on Heatmap**
  - ▶ NYC Venues are displayed in areas of NYC classified by colors according to WS Neighborhood scale

### Results

***Easy way to identify and navigate across NYC sections to find interesting places in neighborhoods.***





# CLUSTERING

## Classify Automatically Neighborhoods and Venues

### ► Data and Features Sets Definition

- Keep only number typed features, drop categorical or string.
- Ex. 'IBR' prices, 'Walk Score number, Venue number and drop others (Budget, etc.)

### ► Normalization/ Standardization over the Standard deviation

- Setting values on the same scale to avoid biasing clustering measure
- Ex: prices in [1000-4000] ws\_scores in [0-100], venue number in [0-50]

### ► Modeling

- Generating Cluster Labels to **4 clusters with 'K-Means'**
- Cluster Centroids definition

### ► Labels Aggregation in DataFrames (Neighborhoods and Venues)

- Resulting Data set can be filtered/sorted to be printed out or displayed on Maps/Heatmaps

## Results

**Labeled Data are grouped in differentiated sets according to « similarity » measure computed by K-Means**

```
Data ready for visualisation reloaded from : NYC_Information_Viz.csv
Neighborhood      object
IBR                float64
Budget             category
Neighborhood Latitude float64
Neighborhood Longitude float64
Number Venues      int64
WS_mean            int64
WS_descr           category
dtype: object 81
```

	Neighborhood	IBR	Budget	Neighborhood Latitude	Neighborhood Longitude	Number Venues	WS_mean	WS_descr
8	Central Brooklyn	2062	Cheap	40.650104	-73.949682	50	97	Walker's Paradise
20	Elmhurst	1900	Cheap	40.736580	-73.878393	49	97	Walker's Paradise

	IBR	Number Venues	WS_mean
0	2900	50	100
1	2995	50	100
2	2999	50	100

Clus_lbl	Number Venues
2	186
0	208
1	592

```
array([[ 0.76282144,  1.05036027,
         0.76282144,  1.05036027,
         0.76282144,  1.05036027,
         0.76282144,  1.05036027,
         0.76282144,  0.91721597],
       [ 0.76282144,  0.91721597],
       [ 0.76282144,  0.91721597],
       [ 0.76282144,  0.91721597],
       [ 0.76282144,  0.78407171],
       [ 0.76282144,  0.78407171],
       [ 0.76282144,  0.78407171],
       [ 0.76282144,  0.78407171]])
```

Neighborhood Clusters Centroids

```
df.groupby('Clus_lbl').mean()
```

	IBR	Number Venues	WS_mean	Index	Clus_lbl
0	2900	50	100	42	1
1	2995	50	100	34	1
2	2999	50	100	35	1

Clus_lbl	IBR	Number Venues	WS_mean
0	2216.916667	23.75	87.000000
1	2547.541667	49.75	97.166667
2	1958.500000	49.75	75.250000
3	2147.400000	33.40	93.600000

	Neighborhood	IBR	Budget	Neighborhood Latitude	Neighborhood Longitude	Number Venues	WS_mean	WS_descr	Index	Clus_lbl
0	Corona	2331	Average	40.746959	-73.860146	34	96	Walker's Paradise	62	0
1	Fort George	1700	Cheap	40.858413	-73.926507	35	95	Walker's Paradise	61	0
2	Ridgewood	2331	Average	40.708056	-73.914167	31	95	Walker's Paradise	64	0

# RESULTS

## Output of Clustering Process

► **Filtering, Sorting and Grouping Data Automatically**

- By 'IBR' prices, 'Walk Score' number, Venue number

► **Printing Selection of Areas Addresses**

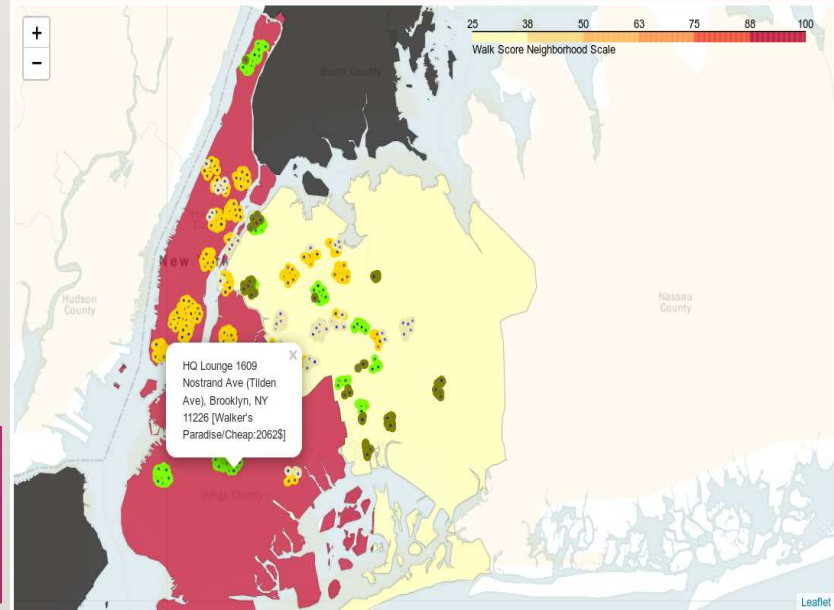
- Choice of criteria
- based on neighborhoods and venues localization

► **Displaying Areas**

- Interactive Maps
- Interactive Heatmaps

I

	Neighborhood	IBR	Budget	Neighborhood Latitude	Neighborhood Longitude	Number Venues	WS_mean	WS_descr	Index	Clus_lbl
0	Corona	2331	Average	40.746959	-73.860146	34	96	Walker's Paradise	62	0
1	Fort George	1700	Cheap	40.858413	-73.926507	35	95	Walker's Paradise	61	0
2	Ridgewood	2331	Average	40.708056	-73.914167	31	95	Walker's Paradise	64	0



➤ **Results**

**Labelled Data are grouped in differentiated sets.**  
**Navigation across different targeted groups is facilitated.**

# CONCLUSIONS & PERSPECTIVES

## ► Study of Data and Features to classify automatically Areas of NYC

- Print textual list of NYC Areas according to required criteria
- Print Interactive Maps/Heatmaps to facilitate the search of interesting places

## ► K-Means Modeling

- Good performance for this use-case with these data
- **Constraints:**
  - Fixing Clusters number
  - Using a limited number of parameters (Walk Score, Number of Venues, Rent Price)
  - Rescaling of parameters at the same magnitude order (equivalent weighting)

## ► Questions and Future

- How to compute automatically the number of clusters ?
- What about the behavior if adding more features ?
- And what if defining different importance to the different features (differentiated weighting) ?
- Another better Clustering method ?

**- END -**

