

# Introduction

starting any business now is very hard because of the highly competitive markets and this applies to every kind of business and OfCourse it is a big problem especially for startups.

I am trying to solve this problem for a specific scenario which is:

someone wants to open a restaurant in crowded city like Tokyo, Cairo, Newyork, ...etc.

OfCourse it would be hard to find the spot that people will go to him in it as there are many restaurants in the area

and areas without restaurants doesn't have restaurants for a reason (like: people don't afford restaurants, not safe, ..etc.)

I will use Forsquare data to solve this problem by searching for an area that will make a new restaurant POPULAR.

## Data

I will only use the Forsquare data, I will search for restaurants and I will get their ratings and delivery id and location so, I would be able to analyze their data and apply ML models to find the best place.

## Methodology

I firstly got the data as a whole from the Forsquare API for the search term “restaurant”

```
In [37]: pd.set_option("display.max_rows", None, "display.max_columns", None)
df3
```

```
Out[37]:
```

	id	name	categories	referralId	hasPerk	location.address	location.crossStreet	location.lat	location.lng
0	49f125dcf964a52091691fe3	O'Hara's Restaurant & Pub	['id': '4bf58dd8d48988d11b941735', 'name': 'P...	1594745128	False	120 Cedar St	at Greenwich St.	40.709894	-74.01283
1	45e5c256f964a52046431fe3	Mudville Restaurant & Tap House	['id': '4bf58dd8d48988d14c941735', 'name': 'W...	1594745128	False	126 Chambers St	btwn W Broadway & Church St	40.715336	-74.00888
2	4bc238adf8219c744286b410	Amore's Pizza Restaurant	['id': '4bf58dd8d48988d1ca941735', 'name': 'P...	1594745128	False	147 Chambers St	Hudson Street	40.715860	-74.00988
3	4c4890da1879c9b6cce6e143	New Shezan Restaurant	['id': '4bf58dd8d48988d115941735', 'name': 'M...	1594745128	False	183 Church St	btwn Duane & Reade St.	40.715789	-74.00722
4	4b3b9355f964a5202e7625e3	Yips Restaurant	['id': '4bf58dd8d48988d145941735', 'name': 'C...	1594745128	False	18 Beaver St	at New St.	40.704950	-74.01236

```
In [124]: df4 = df3[["id", "location.lat", "location.lng", "delivery.id", "name"]]
df4.head()
```

Then I extracted the data that I need by the column's names

```
In [124]: df4 = df3[["id", "location.lat", "location.lng", "delivery.id", "name"]]
df4.head()

Out[124]:
```

	id	location.lat	location.lng	delivery.id	name
0	49f125dcf964a52091691fe3	40.709894	-74.012836	NaN	O'Hara's Restaurant & Pub
1	45e5c256f964a52046431fe3	40.715336	-74.008881	299726	Mudville Restaurant & Tap House
2	4bc238adf8219c744286b410	40.715860	-74.009888	1431324	Amore's Pizza Restaurant
3	4c4890da1879c9b6cce6e143	40.715789	-74.007227	NaN	New Shezan Restaurant
4	4b3b9355f964a5202e7625e3	40.704950	-74.012360	NaN	Yips Restaurant

cleaning the data

Then I called the API for every id in the data frame to get its rating

```
In [126]: venues_ids= df3['id']
ratings=[]
for venue_id in venues_ids.values.tolist():
    url = 'https://api.foursquare.com/v2/venues/{}?client_id={}&client_secret={}&v={}'.format(venue_id, CLIENT_ID, CLIENT_SECRET,
    result = requests.get(url).json()
    with open('data3.json', 'a+') as file:
        json.dump(result, file)
    try:
        venues_rating=result['response']['venue']['rating']
        ratings=ratings+[venues_rating]
    except:
        print('This venue has not been rated yet.')
        ratings = ratings + ["NaN"]
```

requesting every rating of found restaurants

```
In [99]: df4.insert(1,"ratings",ratings)
```

Then I changed the value of the delivery to 0 and 1 which indicates if this restaurant has delivery or not and also the nan ratings to 0 as unknown rating means less popular

```
In [104]: df4

Out[104]:
```

	id	ratings	location.lat	location.lng	delivery.id	name
0	49f125dcf964a52091691fe3	8.1	40.709894	-74.012836	0	O'Hara's Restaurant & Pub
1	45e5c256f964a52046431fe3	7.2	40.715336	-74.008881	1	Mudville Restaurant & Tap House
2	4bc238adf8219c744286b410	6.3	40.715860	-74.009888	1	Amore's Pizza Restaurant
3	4c4890da1879c9b6cce6e143	0	40.715789	-74.007227	0	New Shezan Restaurant
4	4b3b9355f964a5202e7625e3	5.5	40.704950	-74.012360	0	Yips Restaurant
5	4b4dfc8f964a520801827e3	6.6	40.709233	-74.003747	0	TJ Byrnes Bar and Restaurant
6	575808cf38fa9ce070c7ce1b	0	40.708363	-74.014046	0	Su Hang Restaurant
7	40fc6080f964a520040b1fe3	0	40.707970	-74.013779	0	Romi Restaurant & Lounge
8	4b88199af964a520e6df31e3	0	40.709193	-74.009344	0	Win Won Restaurant
9	525ab25d11d236eef03b8496	0	40.715206	-74.009132	0	Restaurant at Smyth
10	4f0c7c106d866af8d40ae337	7.2	40.715080	-74.015306	0	Atrio New York

changing delivery data to 0 and 1 and also ratings of nan to 0

Extracting the ratings and delivery data to use them in the ML model

```
In [117]: df10 = df4[["ratings", "delivery.id"]]
```

```
In [128]: df10.rename(columns={"delivery.id": "delivery"})
df10.head()
```

```
Out[128]:
```

	ratings	delivery.id
0	8.1	0
1	7.2	1
2	6.3	1
3	0	0
4	5.5	0

taking the data that will be used for clustering

Fitting the Kmeans clustering model and inserting the cluster labels for visualization.

```
In [120]: cluster = KMeans(n_clusters= 4,init="k-means++",n_init=10,random_state=0)
cluster.fit(df10)
```

```
Out[120]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
n_clusters=4, n_init=10, n_jobs=None, precompute_distances='auto',
random_state=0, tol=0.0001, verbose=0)
```

fitting the model

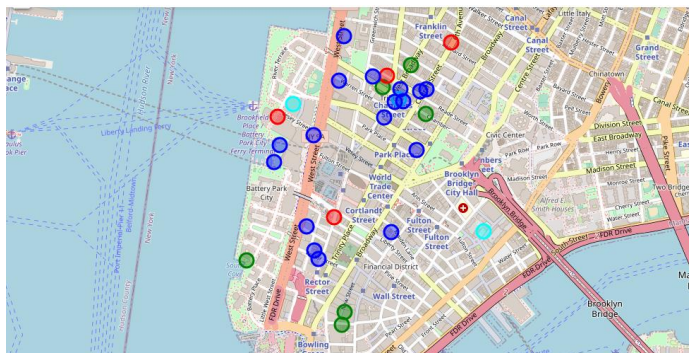
```
In [122]: df4.insert(0, 'cluster labels', cluster.labels_)
```

```
In [132]: df4
```

```
Out[132]:
```

	cluster labels	id	ratings	location.lat	location.lng	delivery.id	name
0	1	49f125dcf964a52091691fe3	8.1	40.709894	-74.012836	0	O'Hara's Restaurant & Pub
1	3	45e5c256f964a52046431fe3	7.2	40.715336	-74.008881	1	Mudville Restaurant & Tap House
2	2	4bc238adf8219c744286b410	6.3	40.715860	-74.009888	1	Amore's Pizza Restaurant
3	0	4c4890da1879c9b6cce6e143	0	40.715789	-74.007227	0	New Shezan Restaurant
4	2	4b3b9355f964a5202e7625e3	5.5	40.704950	-74.012360	0	Yips Restaurant
5	3	4b4dfc8f964a520801827e3	6.6	40.709233	-74.003747	0	TJ Byrnes Bar and Restaurant
6	0	575808cf38fa9ce070c7ce1b	0	40.708363	-74.014046	0	Su Hang Restaurant
7	0	40fc6080f964a520040b1fe3	0	40.707970	-74.013779	0	Romi Restaurant & Lounge

Visualizing the cluster on the map



## Results

After fitting the model with the data (delivery, rating)

There were 4 clusters

0 cluster: unrated, probably no delivery

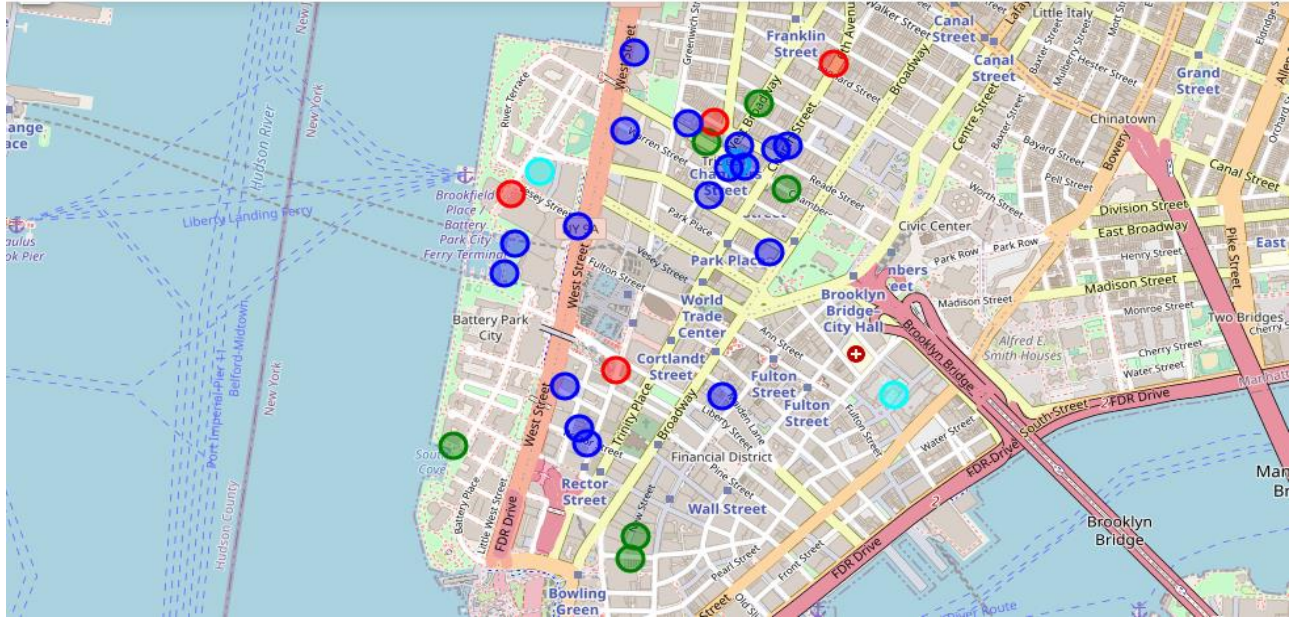
1<sup>st</sup> cluster: high rating, mostly has delivery

2<sup>nd</sup> cluster: low rating, mostly has delivery

3<sup>rd</sup> cluster: medium rating, has no delivery

Based on this data (which is limited and can be mistaken)

The best place to start is away from red dots and close to the green ones or blue ones



## Discussion

In this section I wanted to say that this data is insufficient to make the most precise decision of the restaurant place but I am certain that if there were more data and funds it can be a great metric to decide your spot

## Conclusion

To sum up everything in the report according to this project (if the required data is present) you can open your business in any place if you analyze the surroundings and the weaknesses of your opponents.