



# Applied Data Science

---

COURSERA – CAPSTONE PROJECT

# Goal

---

- Explore a geographical location using FOUR SQUARE API
- Using pandas to explore the available datasets
- Use data visualization techniques to show the result

# Problem Statement

---

- Customer is planning to open a new restaurant in United States. They don't have a clear picture of where to open and how to choose a city for the new restaurant.
- Goal of the customer is to open a new restaurant in a “Good populated” environment.
- In order to be safe, it is expected to have a low crime rate in the chosen location.
- Explore the chosen city for current available restaurants.

# Data Sets

---

- Crime Rate records as per FBI Report
  - <https://ucr.fbi.gov/crime-in-the-u.s/2015/crime-in-the-u.s.-2015>
- Four Square API – Explore by Restaurants
  - <https://api.foursquare.com/v2/venues/search?>
  - search\_query = 'Restaurant'
  - radius = 1000

# Analysis Method

---

- Clean the data by filtering out unwanted columns and filling up “NaN” values.
- Normalize the values so that it can be visually represented in the similar scale.
- Arrange the columns based on “Population” (Descending order)
- Map the top cities with population against the crime rate per 1000
- Map the crime rate with respect to the cities.
- Choose the top 10 cities with highest population
- Select the right city with low crime rate and good enough population.

# Crime Data – Raw Output

---

[192]:

	City	Population	PropertyCrime	Burglary	VehicleTheft
0	Abbeville	2610	76.0	24.0	2.0
1	Adamsville	4414	330.0	38.0	14.0
2	Addison	748	28.0	11.0	2.0
3	Alabaster	31821	534.0	64.0	29.0
4	Albertville	21534	723.0	197.0	64.0
...	...	...	...	...	...
9390	Sundance	1252	15.0	8.0	1.0
9391	Thermopolis	3024	12.0	1.0	4.0
9392	Torrington	6762	116.0	24.0	4.0
9393	Wheatland	3666	94.0	31.0	7.0
9394	Worland	5335	57.0	11.0	0.0

9395 rows × 5 columns

➤ *Displaying only the columns of interest*

# Consolidated Crime Rate Output

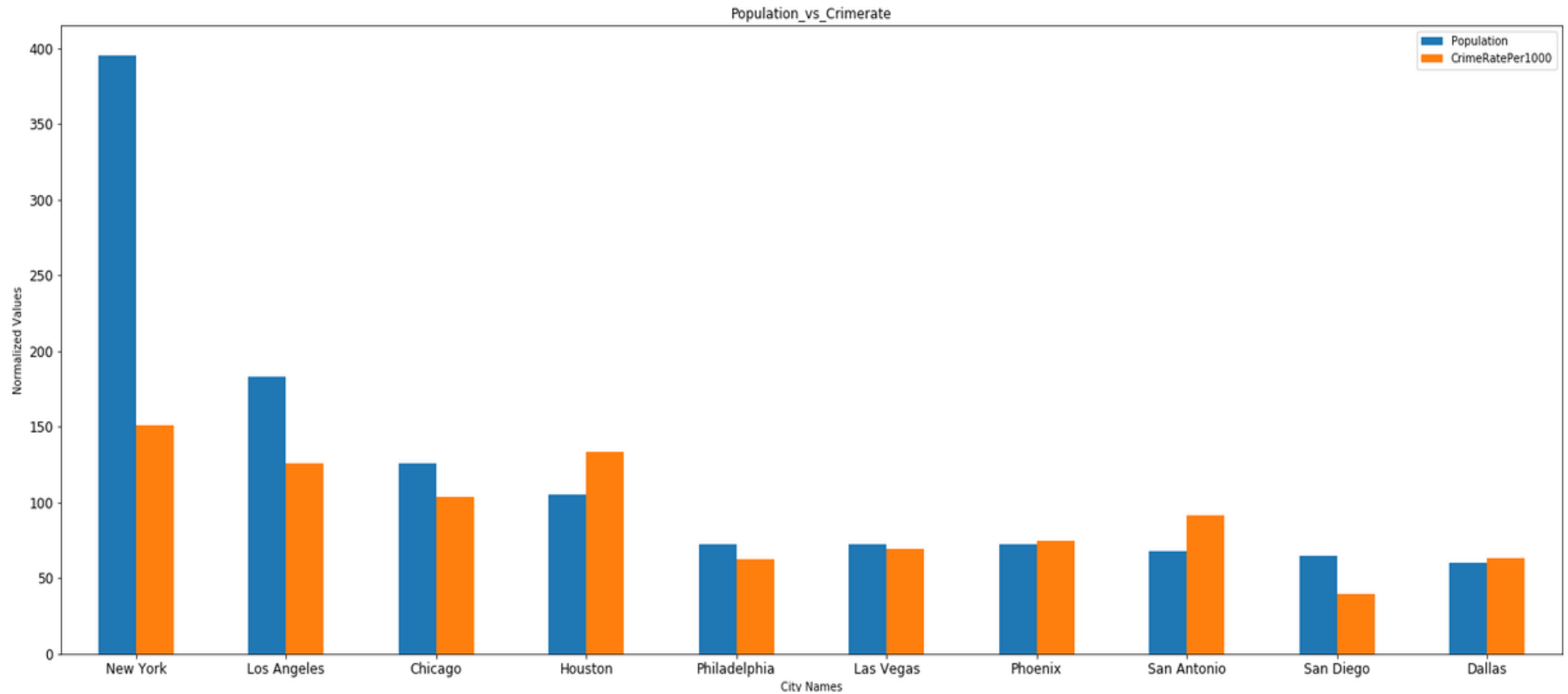
---

[195]:

	City	Population	PropertyCrime	Burglary	VehicleTheft	TotalCrime	CrimeRatePer1000
5595	New York	8550861	129860.0	14098.0	7386.0	151344.0	151.344
697	Los Angeles	3962726	93503.0	16160.0	16152.0	125815.0	125.815
1854	Chicago	2728695	80395.0	13151.0	10222.0	103768.0	103.768
8188	Houston	2275221	100053.0	19859.0	13560.0	133472.0	133.472
7167	Philadelphia	1567810	49345.0	8083.0	5034.0	62462.0	62.462

- We will use the Population & CrimeRatePer1000 columns for visualizing the output.
- Before visualizing, lets normalize the 'Population' column with its "mean" value, for better visual output.

# Data Visualization





# Explore the chosen place

---

- Using Four Square API explore the chosen city with nearby restaurants.
- Visualize the restaurant locations.

# Exploring a specific location

---

```
[238]: address = 'New York City'
geolocator = Nominatim(user_agent="foursquare_agent")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print(latitude, longitude)
#40.7128° N, 74.0060° W (New York City)

40.7127281 -74.0060152
```

*We will analyze the restaurants with respect to this location.*

# Restaurants in 1000m radius

```
[246]: # assign relevant part of JSON to venues
venues = results['response']['venues']
# tranform venues into a dataframe
dataframe = json_normalize(venues)
dataframe.head()
```

[246]:	id	name	categories	referralId	hasPerk	location.address	location.crossStreet	location.lat	location.lng	location.labeledLatLngs
0	3fd66200f964a520d5e31ee3	Jing Fong Restaurant 金豐大酒樓	{'id': '4bf58dd8d48988d1f5931735', 'name': 'D...'	v-1583199256	False	20 Elizabeth St	btwn Bayard & Canal St	40.715881	-73.997209	{'label': 'display', 'lat': 40.7158812029412, ...}
1	3fd66200f964a520ece31ee3	Golden Unicorn Restaurant 麒麟金閣	{'id': '4bf58dd8d48988d1f5931735', 'name': 'D...'	v-1583199256	False	18 E Broadway	at Catherine St	40.713629	-73.997230	{'label': 'display', 'lat': 40.71362850464683, ...}
2	49f125dcf964a52091691fe3	O'Hara's Restaurant & Pub	{'id': '4bf58dd8d48988d11b941735', 'name': 'P...'	v-1583199256	False	120 Cedar St	at Greenwich St.	40.709894	-74.012836	{'label': 'display', 'lat': 40.70989378141622, ...}
3	45e5c256f964a52046431fe3	Mudville Restaurant & Tap House	{'id': '4bf58dd8d48988d14c941735', 'name': 'W...'	v-1583199256	False	126 Chambers St	btwn W Broadway & Church St	40.715336	-74.008881	{'label': 'display', 'lat': 40.71533575723845, ...}
4	4a00df67f964a520ba701fe3	Bo Ky Restaurant 波記潮州小食	{'id': '4bf58dd8d48988d145941735', 'name': 'C...'	v-1583199256	False	80 Bayard St	at Mott St	40.715696	-73.998667	{'label': 'display', 'lat': 40.71569636637641, ...}

# Clean the data

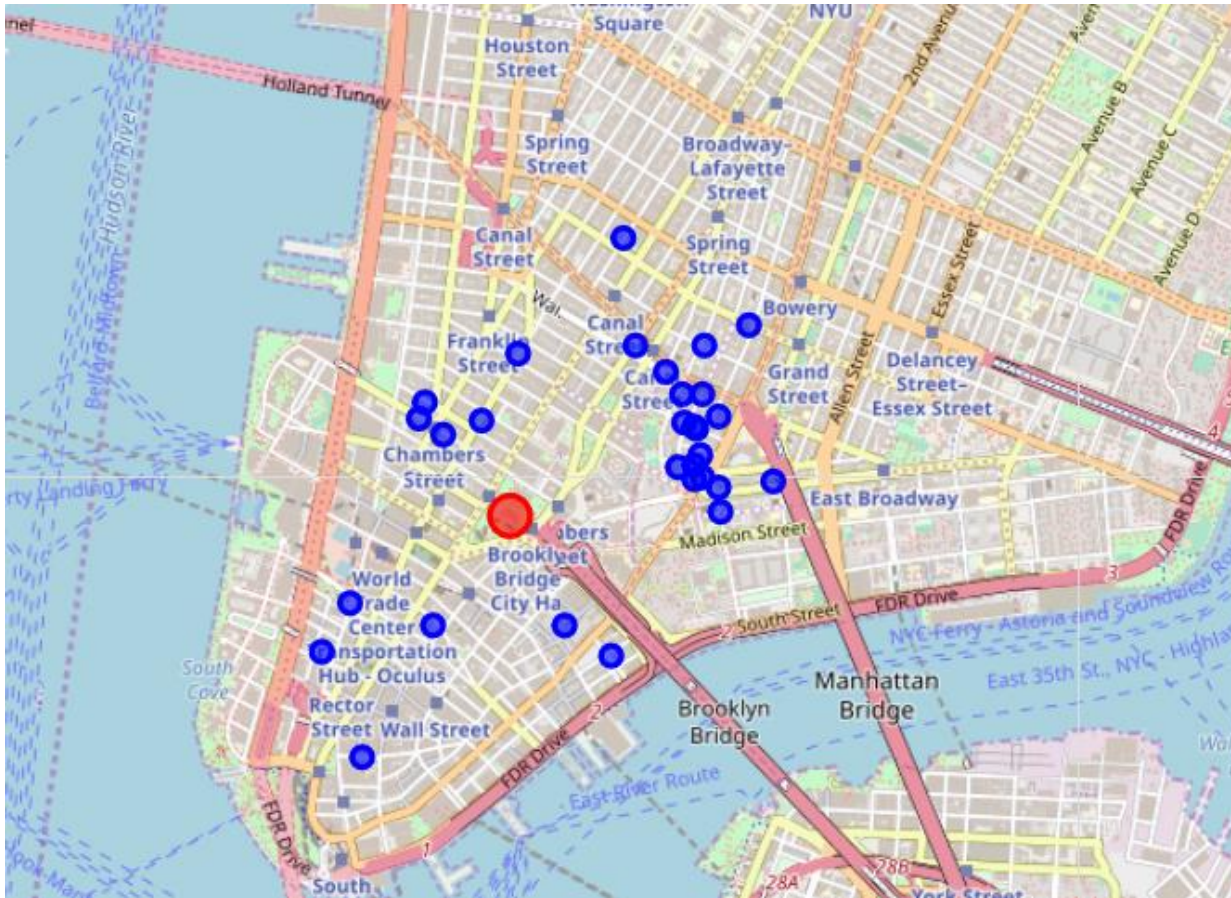
---

- Choose the specific columns of the JSON data
  - Id
  - Name
  - Category

Create a map around the chosen “Latitude , Longitude” and place the “Restaurant Results” on map

# Visualizing the cleaned-up data

---



- Total of 30 restaurants are located near the chosen location.
- Customer can visualize the output and then select their strategic location.

# Conclusion

---

- Based on the analysis it looks like “New York City” population tops the list.
- With respect to the “No of reported crimes” in “New York City” and its “Population” crime rate seems to be lesser compared to other cities.
- Customer can choose “New York City” as their new venue location.
- We have identified the “Densely populated” restaurant locations in “New York City” via map.
- Customer can use this map to locate their strategic location.

# Future Scope

---

- Collect more info on the different categories of restaurant in the “New York Location”
- Identify the most famous / trending restaurant category in the “New York Location”
- Based on the “Visual Output” identify a location to open a new restaurant where it doesn’t exist now near the “Trending Places”