CAPSTONE PROJECT – RESTAURANT SEGMENTATION BY CRIME

Introduction and Background

In the Capstone Project, my idea is to show that when driven by venue and location data from FourSquare AP1, backed up with open source crime data, it is possible to present a new/first-time/solo traveller with a list of attractions to visit supplementd with a graphics showing the crime occurance in the region of the venue.

A high level approach is as follows:

1. The travellers decides on a city location
2. Fetch the top venues from FourSquare
3. Augment the list of top venues with additional geographical data
4. Using this additional geographical data the top nearby restaurents are selected
5. The historical crime within a predetermined distance of all venues are obtained
6. A map is presented to the to the traveller showing the selected venues and crime statistics of the area.

Who are the end users?

This solution is targeted at new/first-time/cautious traveller. They want to see all the main sites of a city that they have never visited before but at the same time, for whatever reaons unknown, they want to be able to do all that they can to make sure that they stay clear of trouble i.e. is it safe to visit this venue and this restaurant at 4:00 pm in the afternoon.

There are many data science aspect of this project including:

- Data Acquisition
- Data Cleansing
- Data Analysis
- Machine Learning
- Prediction

Data

In this section, I will describe the data used to solve the problem. We discuss the following 4 scenarios:

1. Query the FourSqaure website for the top sites in Chicago
2. Use the FourSquare API to get supplemental geographical data about the top sites
3. Use the FourSquare API to get top restaurent recommendations closest to each of the top site

4. Use open source Chicago Crime data to provide the user with additional crime data

1. Query the FourSqaure website for the top sites in Chicago

Although FourSquare provides a comprehensive API, one of the things that API does not easily support is a mechanism to directly extract the top N sites / venues in a given city. This data, however, is easily available directly from the FourSquare Website. To do this simply go to www.foursquare.com, enter the city of your choise and select Top Picks from I'm Looking For selection field.

Using BeautifulSoup and Requests the results of the Top Pick for Chicago was retrieved. From this HTML the following data can be extracted:

- Venue Name
- Venue Score
- Venue Category
- Venue HREF
- Venue ID [Extracted from the HREF]

2. Use the FourSquare API to get supplemental geographical data about the top sites

Using the id field extracted from the HTML it is then possible to get further supplemental geographical details about each of the top sites from FourSquare. We extract the following fields:

From this the following attributes are extracted:

- Venue Address
- Venue Postalcode
- Venue City
- Venue Latitude
- Venue Longitude

3. Use the FourSquare API to get top restaurent recommendations closest to each of the top site

Using the the list of all id values in the Top Sites DataFrame and the FourSquare categoryID that represents all food venues we now search for restaurants within a 500 meter radius. From this JSON the following attributes are extraced and added to the Dataframe:

- Restaurant ID
- Restaurant Category Name
- Restaurant Category ID
- Restaurant Nest_name
- Restaurant Address
- Restaurant Postalcode
- Restaurant City
- Restaurant Latitude
- Restaurant Longitude
- Venue Name

- Venue Latitude
- Venue Longitude

The only piece of data that is missing is the Score or Rating of the Restaurant. To get this we need to make another FourSquare API query using the id of the Restaurant.

4. Use open source Chicago Crime data to provide the user with additional crime data

This dataset can be download from the Chicago Data Portal and reflects reported incidents of crime (with the exception of murders where data exists for each victim) that occurred in the City of Chicago in the last year, minus the most recent seven days. A full desription of the data is available on the site.

Data is extracted from the Chicago Police Department's CLEAR (Citizen Law Enforcement Analysis and Reporting) system. In order to protect the privacy of crime victims, addresses are shown at the block level only and specific locations are not identified.

Not all of the attributes are required so on the following data was imported:

1. Date of Occurance
2. Block
3. Primary Description
4. Ward
5. Latitude
6. Longitude

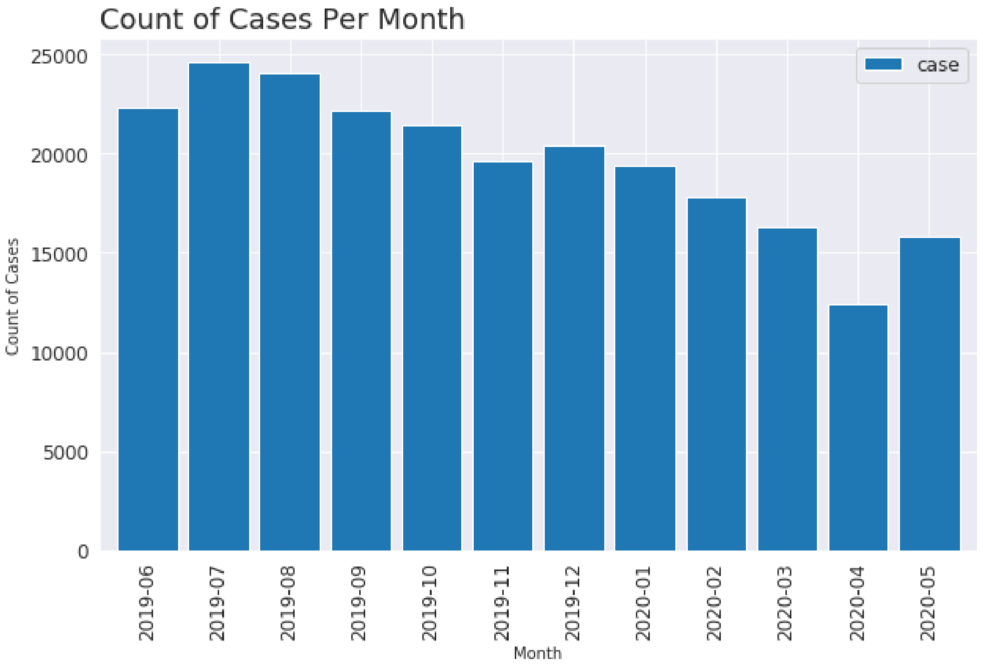
Methodology

In this section, i discuss and describe exploratory data analysis, inferential statistical testing machine learning techniques used.

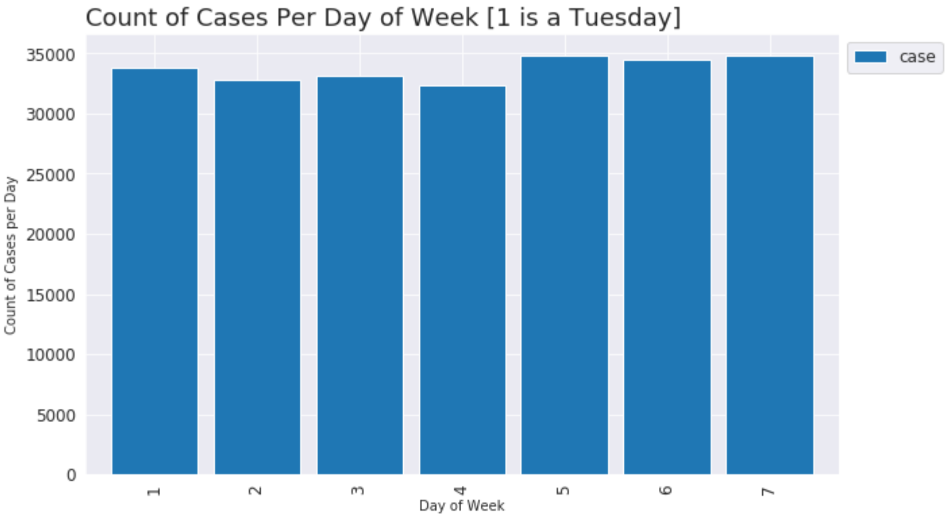
Exploratory Data Analysis and Statistical Testing

We perform the following analysis on the crime data set:

1. Number of Crimes per month

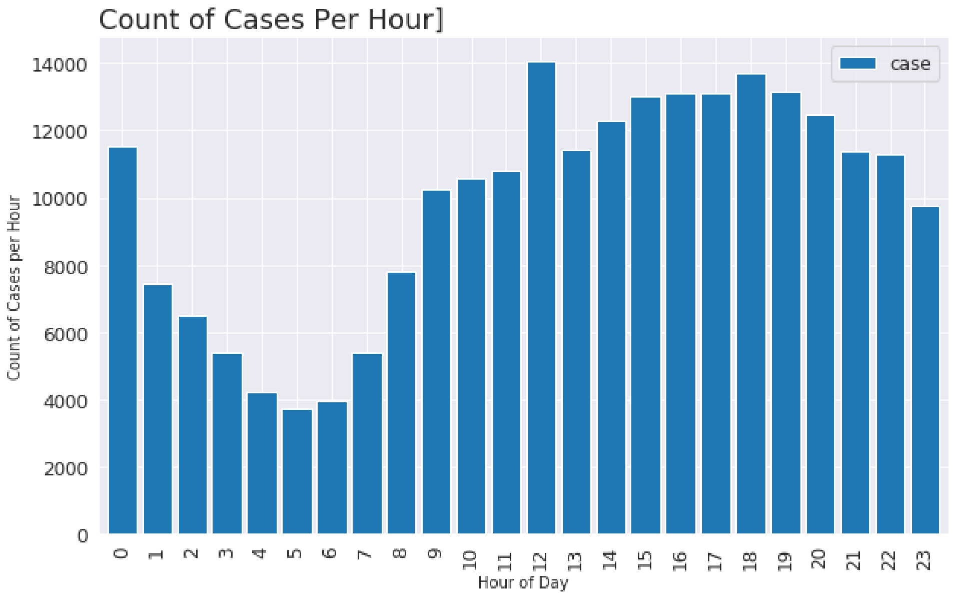


2. Number of crimes occuring on each day



There is a small increase in crime reported at the weekend, Saturday and Sunday, but nothing that couldbe considered significant.

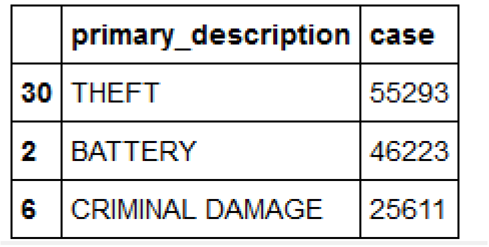
3. Number of crimes occuring in each hour



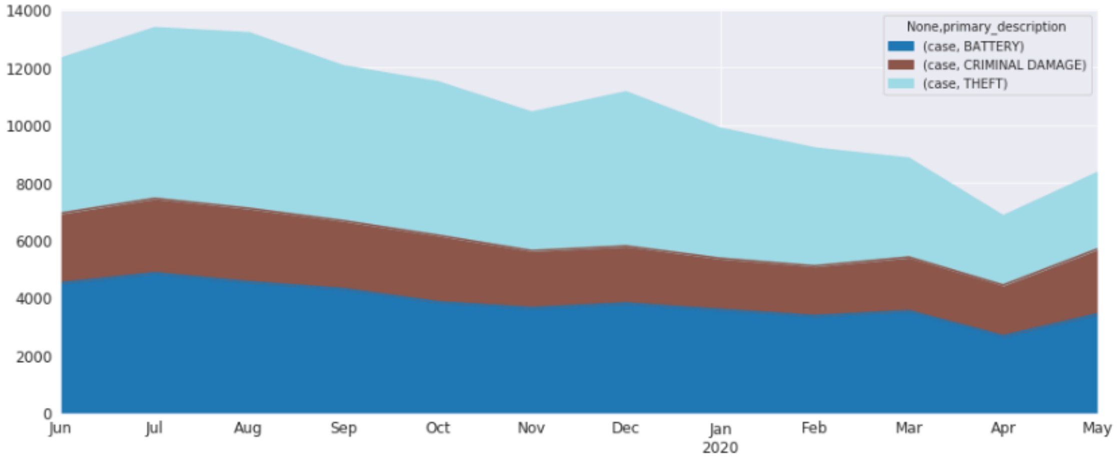
There is an expected fall-off in reported crime rates after midnight before elevating again after eight in the morning.

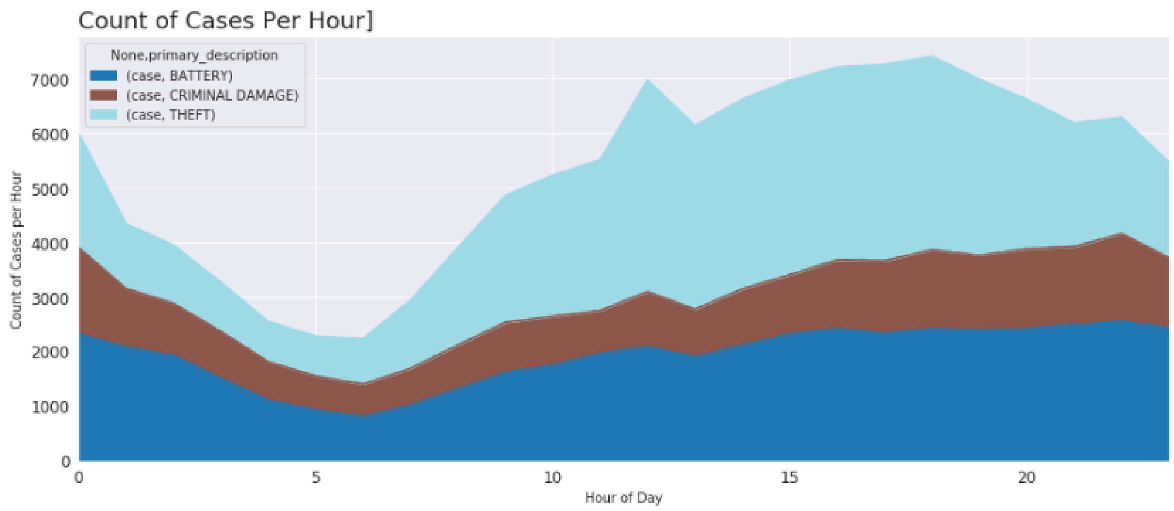
4. Crime Categories

3 most commonly occurring crimes:



	primary_description	case
30	THEFT	55293
2	BATTERY	46223
6	CRIMINAL DAMAGE	25611



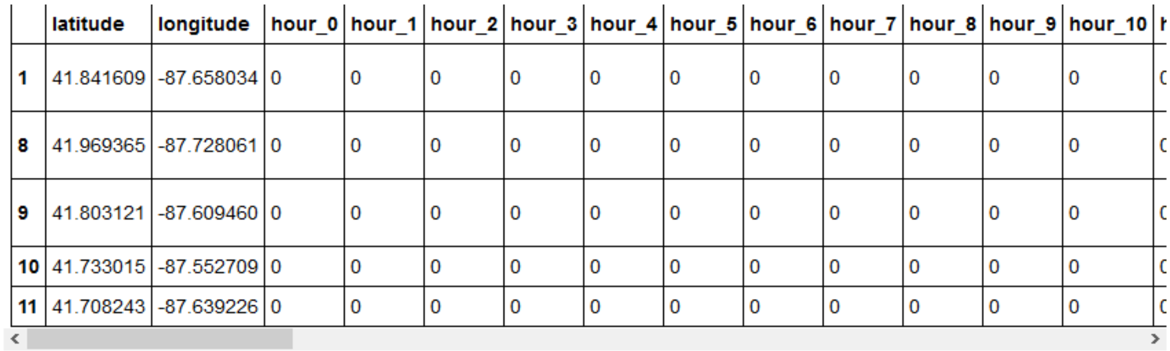


Modelling and Machine Learning

Before we start modelling we need to prepare the data frame to include only numerical data and by removing unneeded columns.

Rather than removing colums from `df_crimes` a new `df_features` DataFrame was created with just the required columns. This `df_features` DataFrame was then processed to remove Categorical Data Types and replace them with One Hot encoding. Finally the Dependant Variables were Normalised.

The Features DataFrame looked like this:



	latitude	longitude	hour_0	hour_1	hour_2	hour_3	hour_4	hour_5	hour_6	hour_7	hour_8	hour_9	hour_10	h
1	41.841609	-87.658034	0	0	0	0	0	0	0	0	0	0	0	0
8	41.969365	-87.728061	0	0	0	0	0	0	0	0	0	0	0	0
9	41.803121	-87.609460	0	0	0	0	0	0	0	0	0	0	0	0
10	41.733015	-87.552709	0	0	0	0	0	0	0	0	0	0	0	0
11	41.708243	-87.639226	0	0	0	0	0	0	0	0	0	0	0	0

There was one significant issue with the crimes data frame as acquired. Although multiclass classification / prediction is possible, the crimes dataset is unbalanced. Modelling algorithms work best when there is approximately an equal number of samples for each class for example The Curse of Class Imbalance and Class imbalance and the curse of minority hubs.

For this reason the modelling task was turned into a simple binary classification task by only modelling based on the top two most occuring crimes. For each model development 10 Fold Cross Validation was used to ensure the best results were achieved and a Grid Search approach was used to determine the best setting for each of the models.

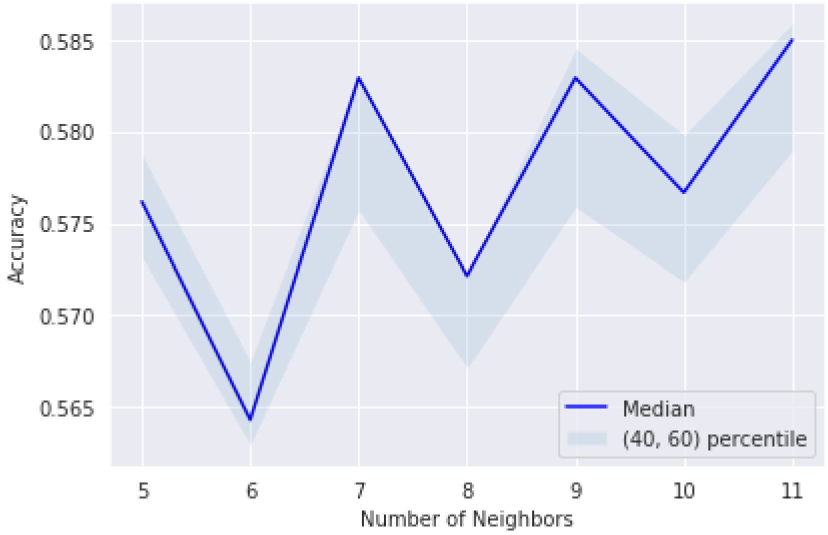
Five model type were then chosen to be evaluated:

1. K Nearest Neighbours
2. Decision Trees
3. Logestic Regression
4. Naive Bayes
5. Decision Forest using a Random Forest

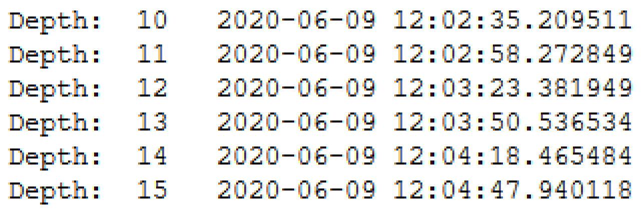
K Nearest Neighbor(KNN)

Find the best k to build the model with the best accuracy:

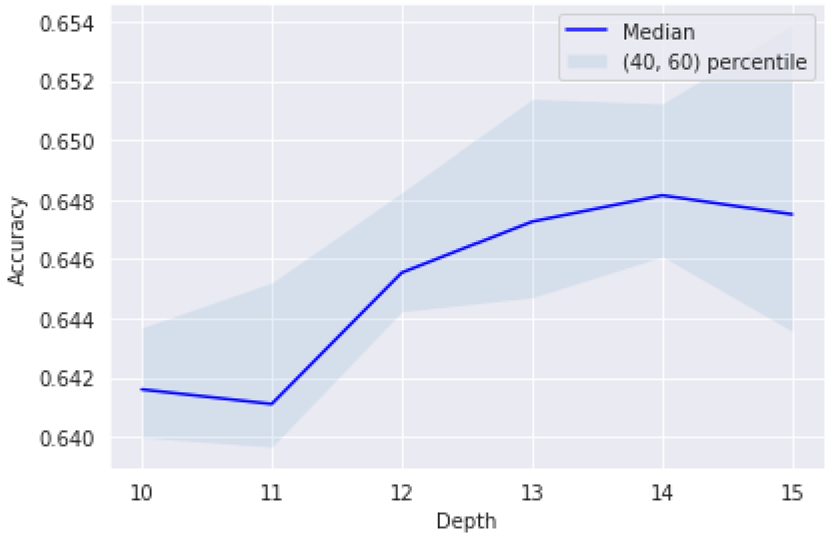
```
Neighbours:  5   2020-06-09 09:53:41.963387
Neighbours:  6   2020-06-09 10:10:46.628468
Neighbours:  7   2020-06-09 10:28:08.873893
Neighbours:  8   2020-06-09 10:46:12.056430
Neighbours:  9   2020-06-09 11:04:39.679689
Neighbours:  10   2020-06-09 11:23:34.661549
Neighbours:  11   2020-06-09 11:42:53.788871
```



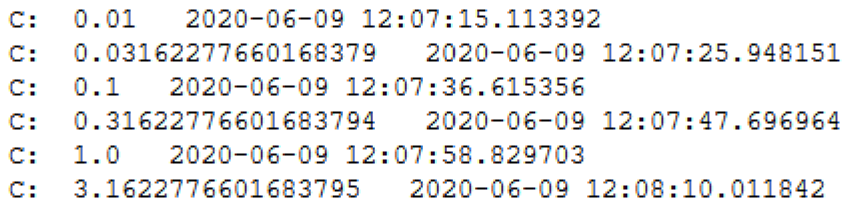
Decision Tree



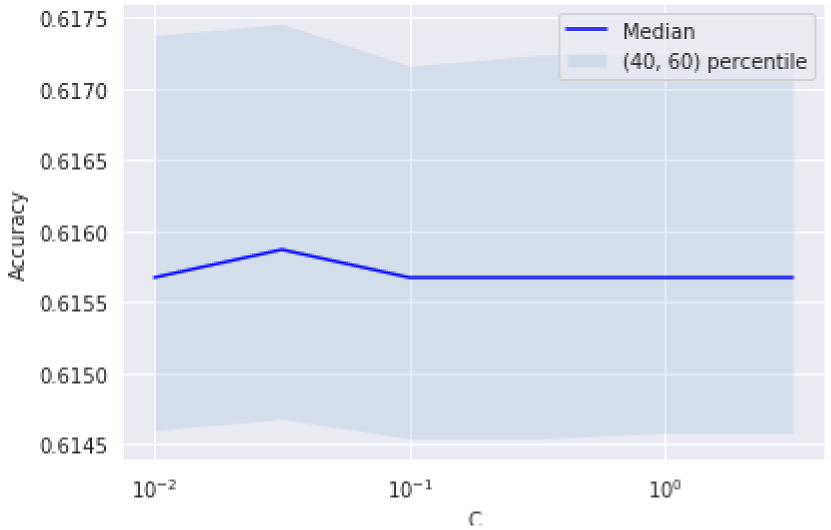
```
Depth:  10   2020-06-09 12:02:35.209511
Depth:  11   2020-06-09 12:02:58.272849
Depth:  12   2020-06-09 12:03:23.381949
Depth:  13   2020-06-09 12:03:50.536534
Depth:  14   2020-06-09 12:04:18.465484
Depth:  15   2020-06-09 12:04:47.940118
```



Logistic Regression

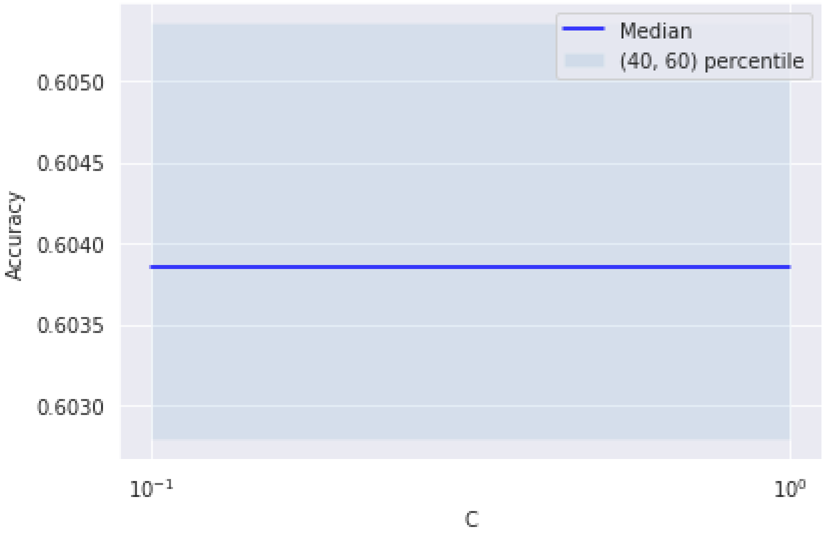


```
C:  0.01   2020-06-09 12:07:15.113392
C:  0.03162277660168379   2020-06-09 12:07:25.948151
C:  0.1   2020-06-09 12:07:36.615356
C:  0.31622776601683794   2020-06-09 12:07:47.696964
C:  1.0   2020-06-09 12:07:58.829703
C:  3.1622776601683795   2020-06-09 12:08:10.011842
```



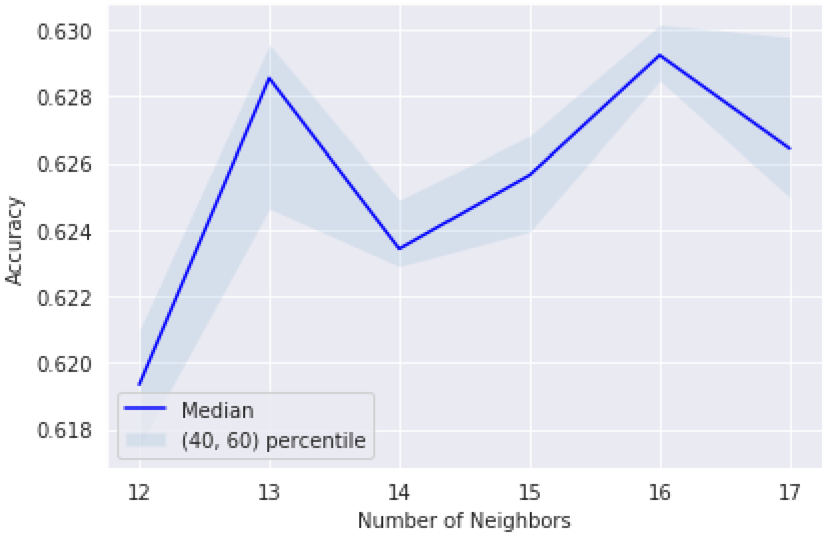
Naive Bayes

```
Alpha:  0.1   2020-06-09 12:08:21.823293
Alpha:  0.2   2020-06-09 12:08:26.475823
Alpha:  0.30000000000000004   2020-06-09 12:08:31.153939
Alpha:  0.4   2020-06-09 12:08:35.878085
Alpha:  0.5   2020-06-09 12:08:40.559441
Alpha:  0.6   2020-06-09 12:08:45.045613
Alpha:  0.7000000000000001   2020-06-09 12:08:49.769498
Alpha:  0.8   2020-06-09 12:08:54.588773
Alpha:  0.9   2020-06-09 12:08:59.233789
Alpha:  1.0   2020-06-09 12:09:03.709130
```



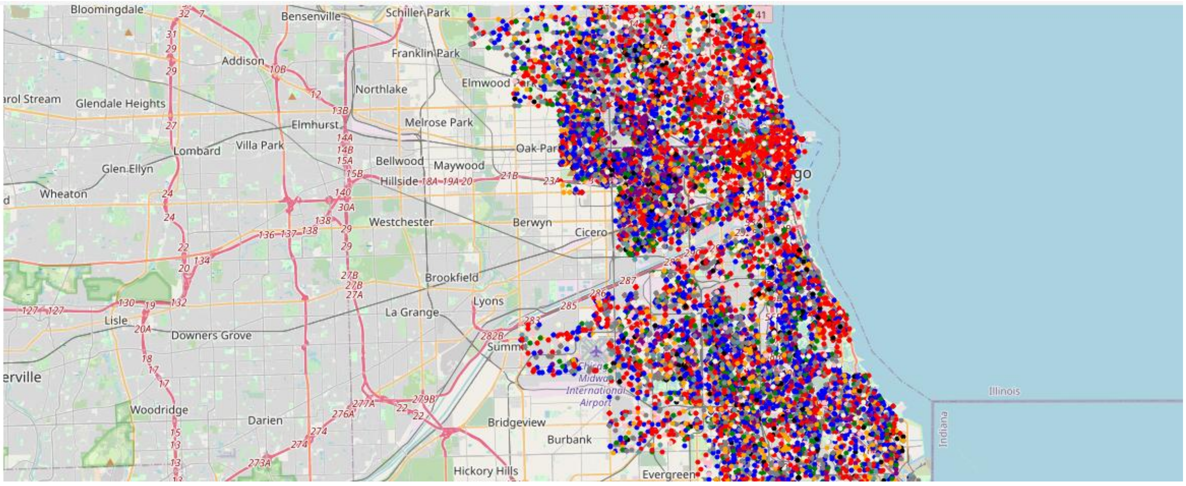
Random Forest

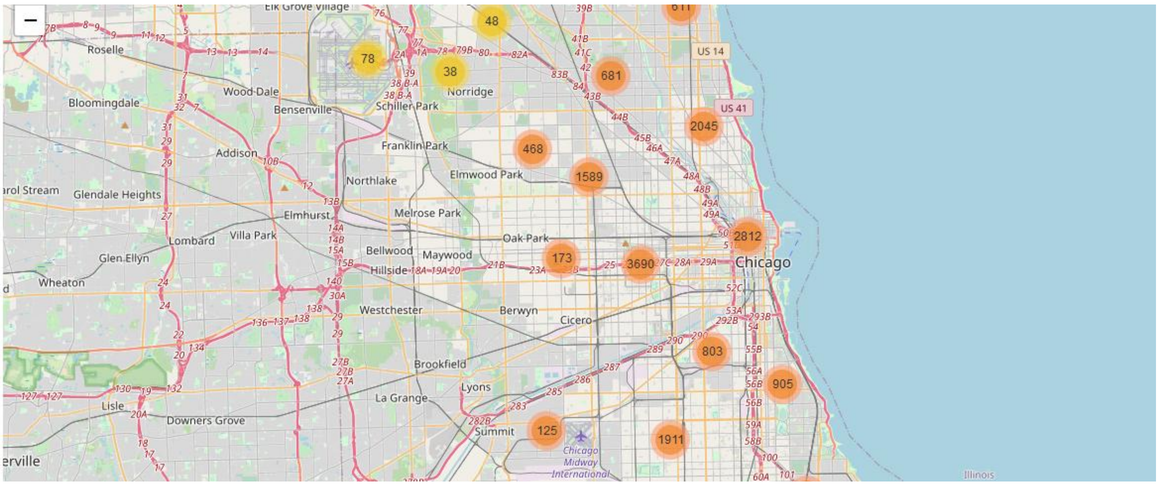
```
Estimator:  12   2020-06-09 12:09:08.773907
Estimator:  13   2020-06-09 12:09:50.456010
Estimator:  14   2020-06-09 12:10:36.761925
Estimator:  15   2020-06-09 12:11:27.762148
Estimator:  16   2020-06-09 12:12:22.985768
Estimator:  17   2020-06-09 12:13:23.949602
```



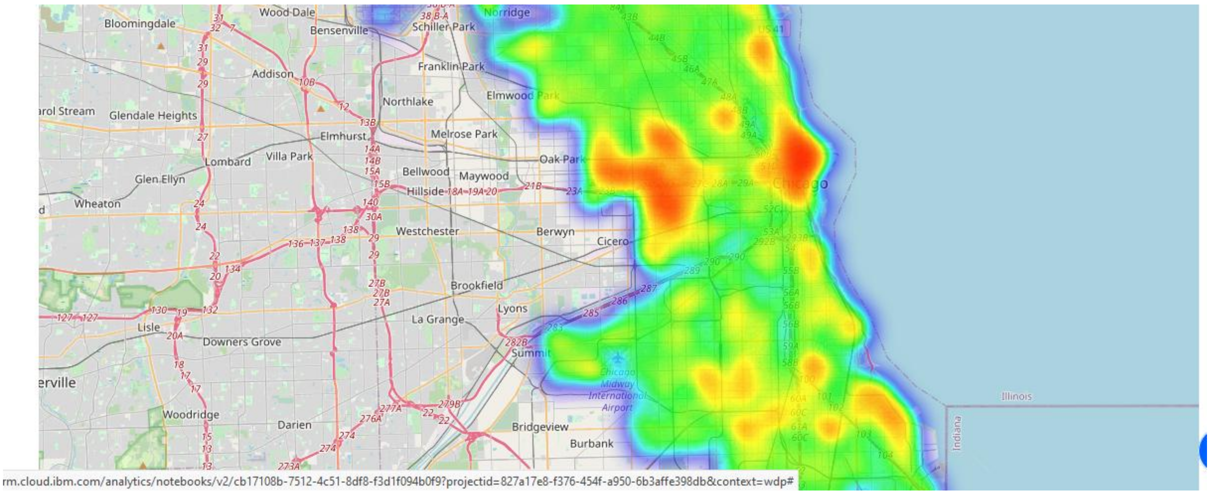
RESULTS:

Create a folium map with a different colour per crime:



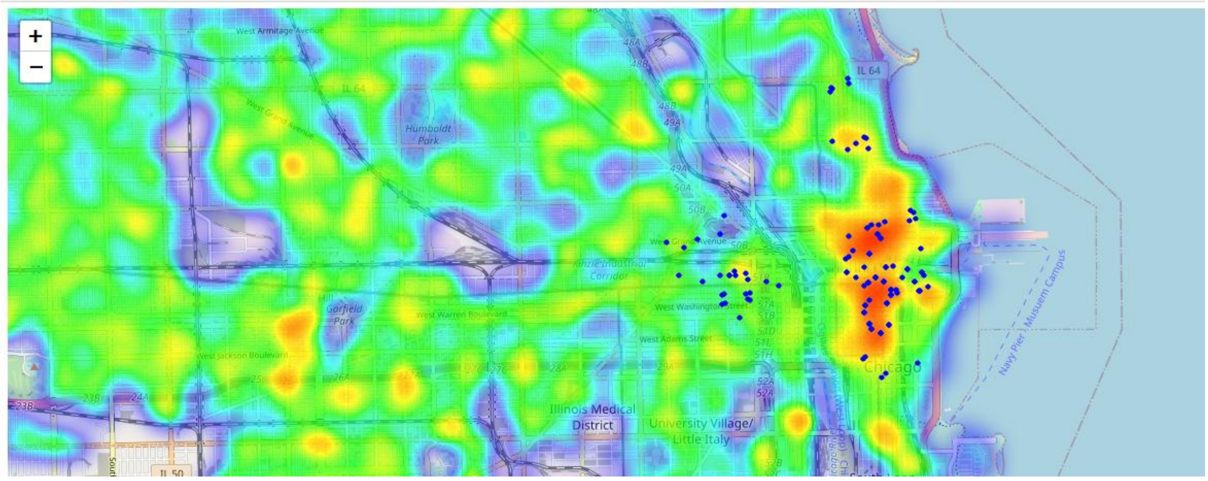


Crime heatmap:



rm.cloud.ibm.com/analytics/notebooks/v2/cb17108b-7512-4c51-8df8-f3d1f094b0f9?projectid=827a17e8-f376-454f-a950-6b3affe398db&context=wdp#

Popular restaurants visualized on heatmap:



This will help tourist decide a restaurant where crime probability is lesser

DISCUSSION

In all we were able to build visualizations that would help the tourists decide on locations to visit. The decision is going to be of the traveller. Whether he/she wants to totally avoid an area that has high crime rate or want to visit a popular attraction despite that. We can clearly make more such maps for other categories of attractions as we have done for restaurants.

Conclusion

Although all of the goals of this project were met there is definitely room for further improvement and development as noted below. However, the goals of the project were met and, with some more work, could easily be devleoped into a fully phledged application that could support the cautious traveller in an unknown location.

Of the contributing data the Chicago Crime data is the one where more data would be good to have. Also not every city in the world makes this data freely available so that is a drawback.

FourSquare proved to be a good source of data but frustrating at times. Despite having a Developer account I regularly exceeded my hourly limit locking me out for the day. This is why Pickle was used to store the captured data.