Lab 4

Reza Shisheie

2708062

November 13, 2018

1. Data set and goal:

The goal of this lab is finding hot spot of a particular data set. Target data set is NIJ data set which holds the crime type, date, and location of all crimes in a particular period of time.

I am picking crime type (burglary, street crime etc.) and location (x and y) as target attributes and try to find the hot spot of each crime.

Data size is 18K which is too much and I am not going to focus on all of the crimes. I am just focusing on burglary data which has 190 data

2. Data processing methods

Two methods of DBSCAN and K-means are used to evaluate results. As far as data preprocessing there would be no need to standardize data as Euclidean distance is used. I am using the actual x and y data in the original coordination system (planar coordination system in feet) to get Euclidean distance.

<u>DBSCAN</u>:

DBSCAN method is a density-based clustering algorithm: given a set of points in some space, it groups together points that are closely packed together (points with many nearby neighbors), marking as outliers points that lie alone in low-density regions (whose nearest neighbors are too far away).

DBSCAN requires two parameters: ε (eps) and the minimum number of points required to form a dense region (minPts). It starts with an arbitrary starting point that has not been visited. This point's ε-neighborhood is retrieved, and if it contains sufficiently many points, a cluster is started. Otherwise, the point is labeled as noise. Note that this point might later be found in a sufficiently sized ε-environment of a different point and hence be made part of a cluster.

If a point is found to be a dense part of a cluster, its ε-neighborhood is also part of that cluster. Hence, all points that are found within the ε-neighborhood are added, as is their own ε-neighborhood when they are also dense. This process continues until the density-connected cluster is completely found. Then, a new unvisited point is retrieved and processed, leading to the discovery of a further cluster or noise.

DBSCAN can be used with any distance function. The distance function (dist) can therefore be seen as an additional parameter. This parameter is set to euclidean distance for this project.

<u>K-means:</u>

k-means clustering is a method of vector quantization, originally from signal processing, that is popular for cluster analysis in data mining. k-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster.

The KMeans algorithm clusters data by trying to separate samples in n groups of equal variance, minimizing a criterion known as the inertia or within-cluster sum-of-squares.

K-means is probably not the best method to be used for lab 4 as it only divides data into groups of clusters with closest mean point for each cluster. It does not provide any feedback on where data has concentrated.

3. Evaluation methods

K-means:

For K-means prediction Accuracy Index (PAI) is used which is:

$$\frac{\frac{n}{N}}{\frac{a}{A}}$$

where n is the number of tuples in each cluster, N is the whole number of tuples, a is the area of that cluster and A is the area of the whole data set.

The larger the number for each number of clustering, the more data is concentrated in that area.

DBSCAN:

DBSCAN method is a density-based clustering algorithm. Give the radius from center any point among all the data and the number of minimum points it finds appropriate clusters.

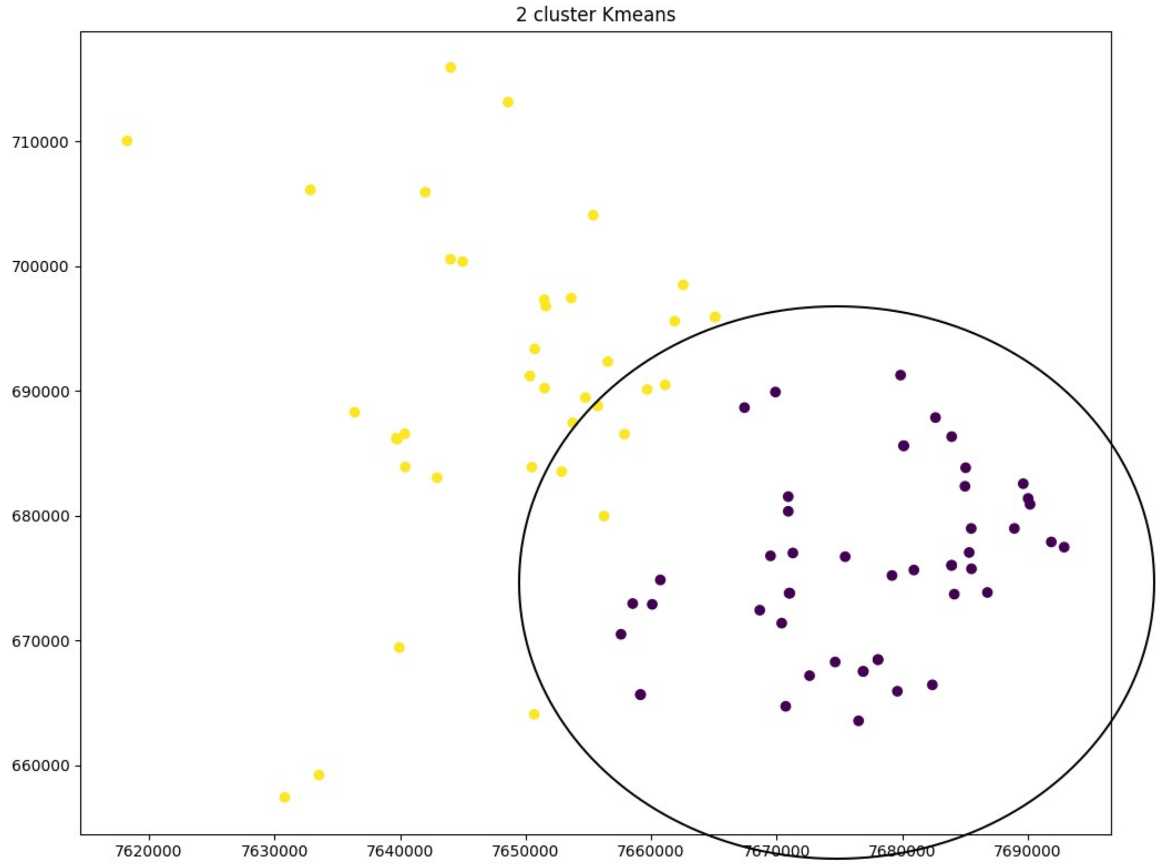
The metric to evaluate DBSCAN algorithm is a little bit trickier than K- means as it needs some knowledge of data set like how large the radius should be or how many incidents can be called as a hot spot.

4. Results

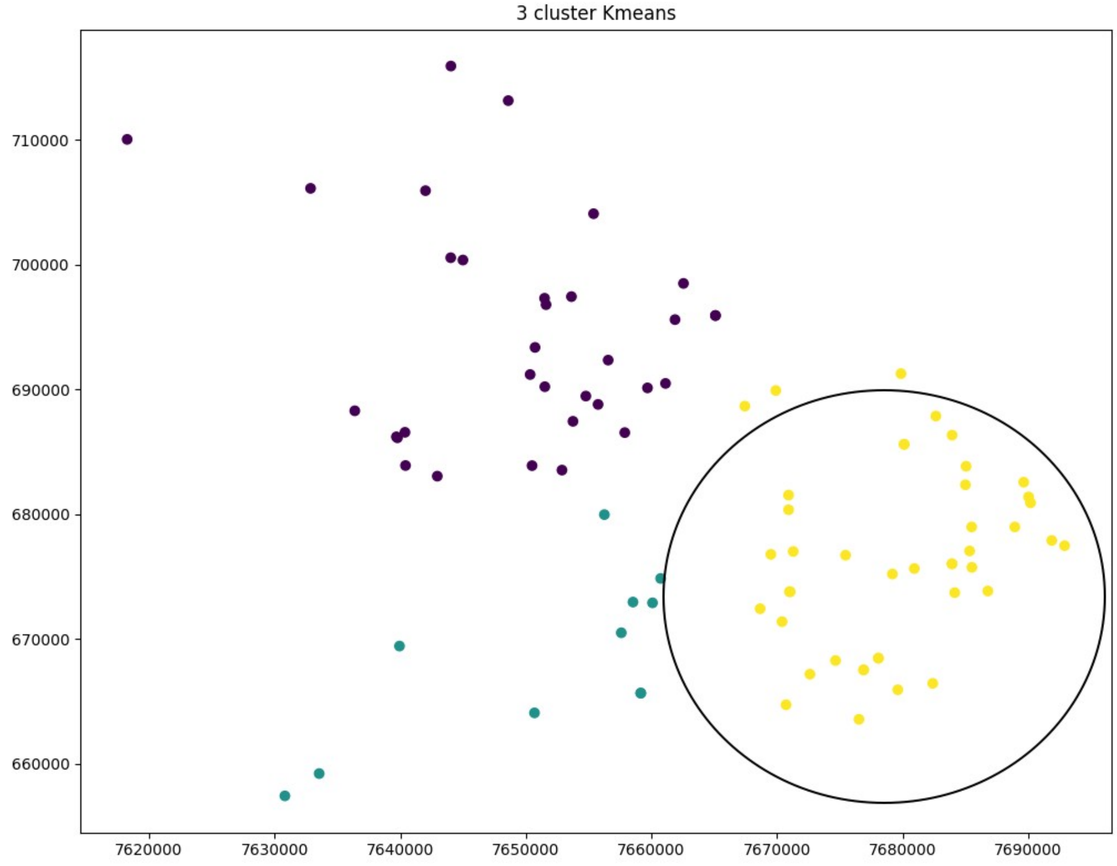
4.1. K-means:

K-means was run for number of clusters from 2 to 7. Here are the results. Best results for each cluster is marked in red and is circled in the plot:

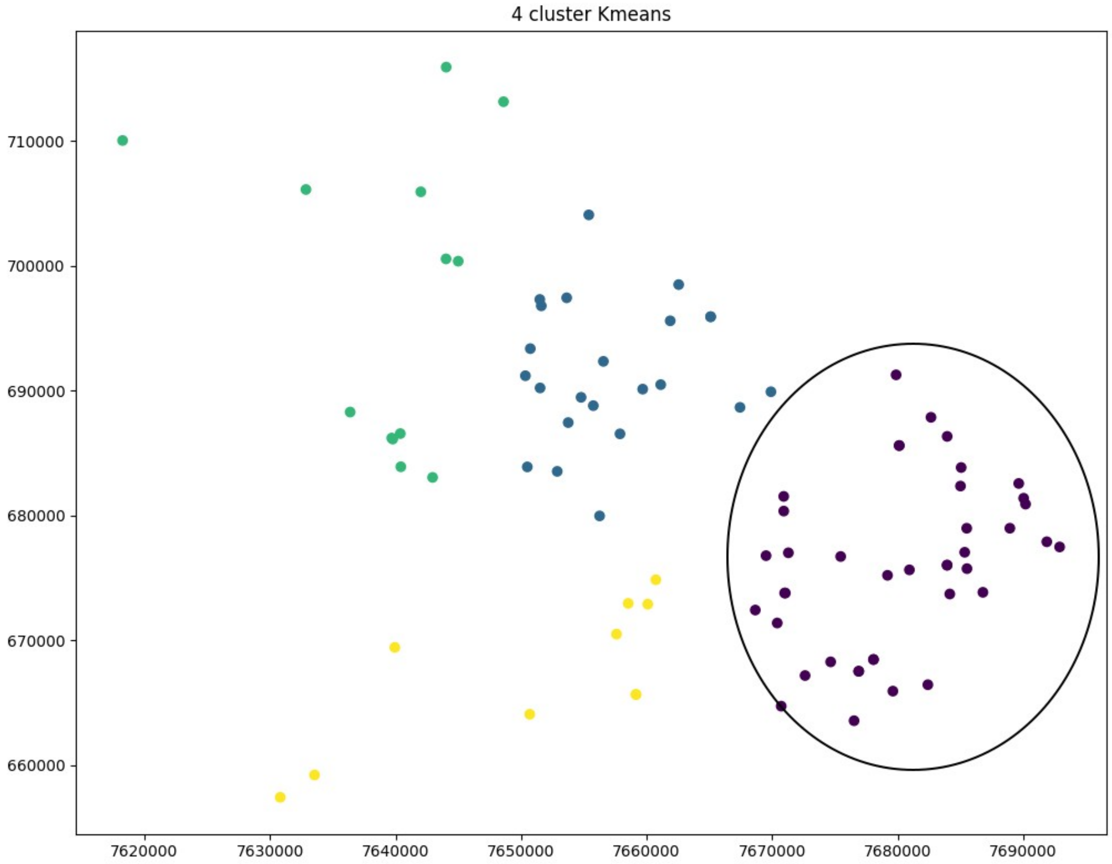
number of clusters: 2
Count: 49 -- current area: 977.490118 -- PAI: 2.51439364106
Count: 38 -- current area: 2738.466009 -- PAI: 0.696026551668



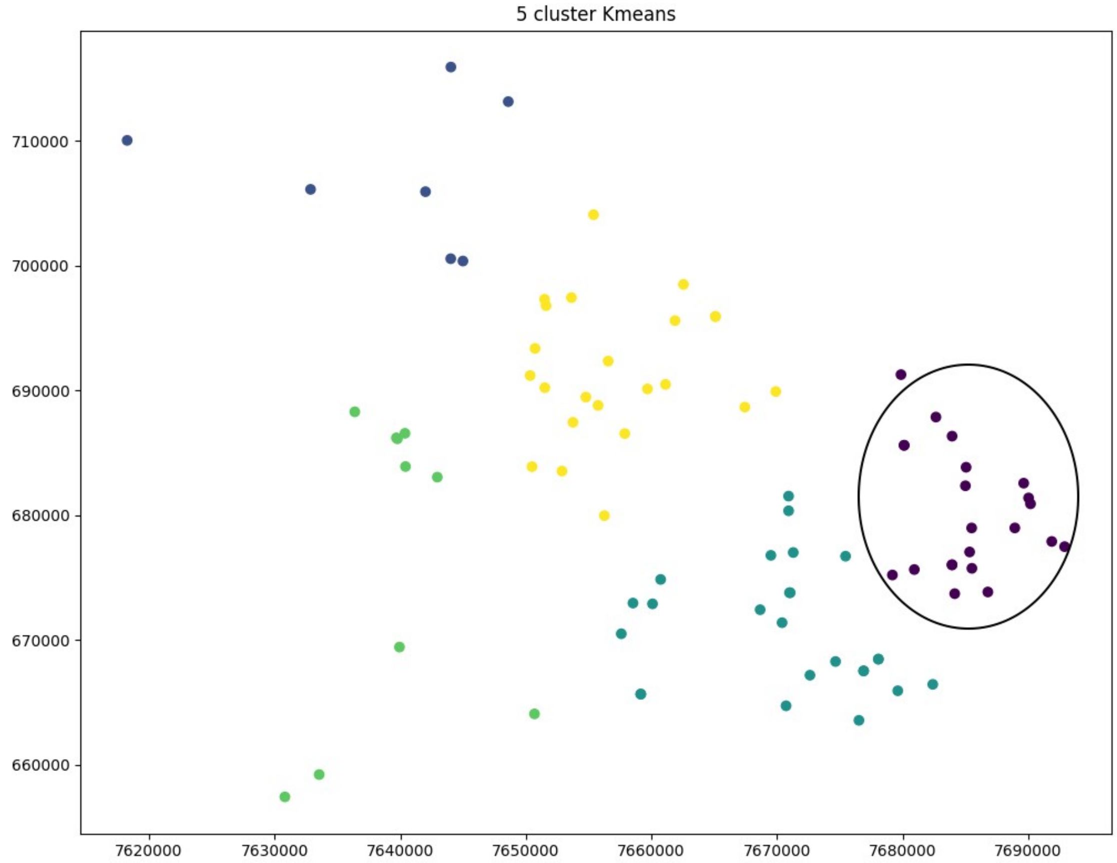
```
number of clusters:  3
Count: 33  -- current area:  1538.791881  -- PAI: 1.07568130026
Count: 11  -- current area:  674.052162  -- PAI: 0.818556656144
Count: 43  -- current area:  705.107986  -- PAI: 3.05887962929
```



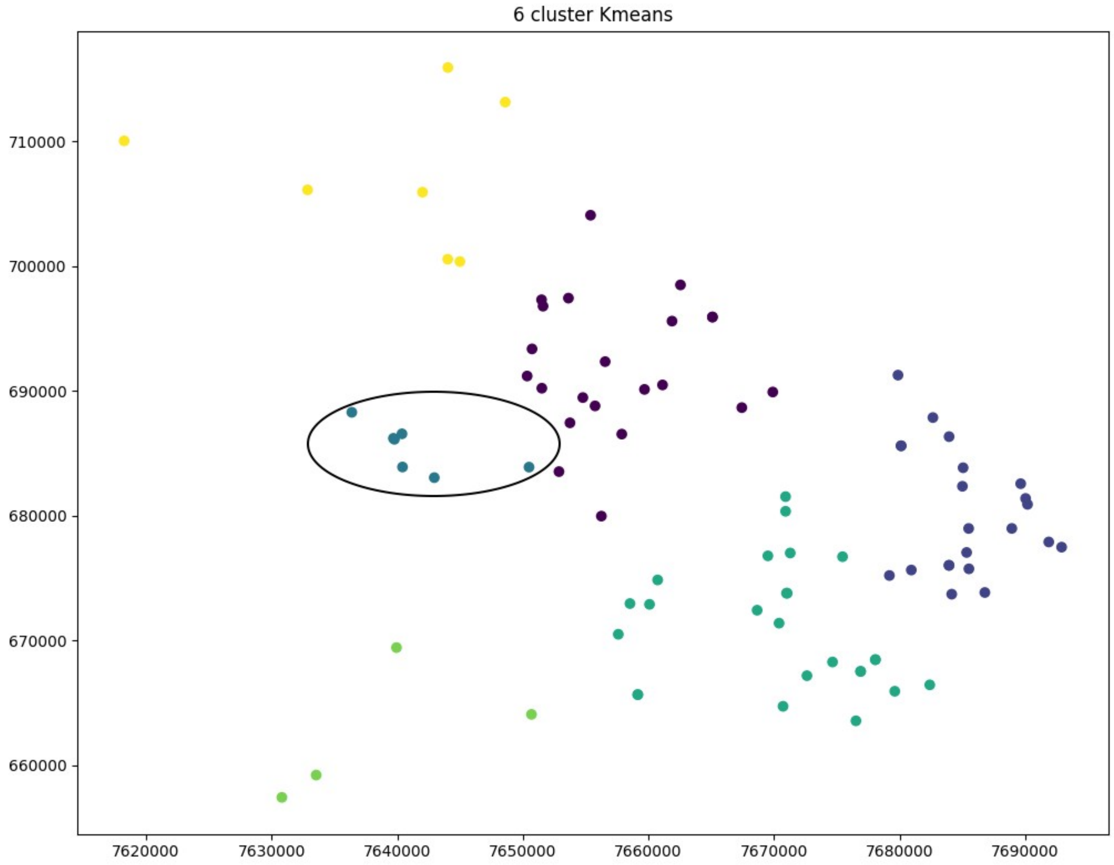
```
number of clusters:  4
Count: 41  -- current area:  671.371822  -- PAI: 3.06316444442
Count: 23  -- current area:  471.456048  -- PAI: 2.44701251119
Count: 13  -- current area:  996.88138  -- PAI: 0.654107958645
Count: 10  -- current area:  521.131794  -- PAI: 0.962502785712
```



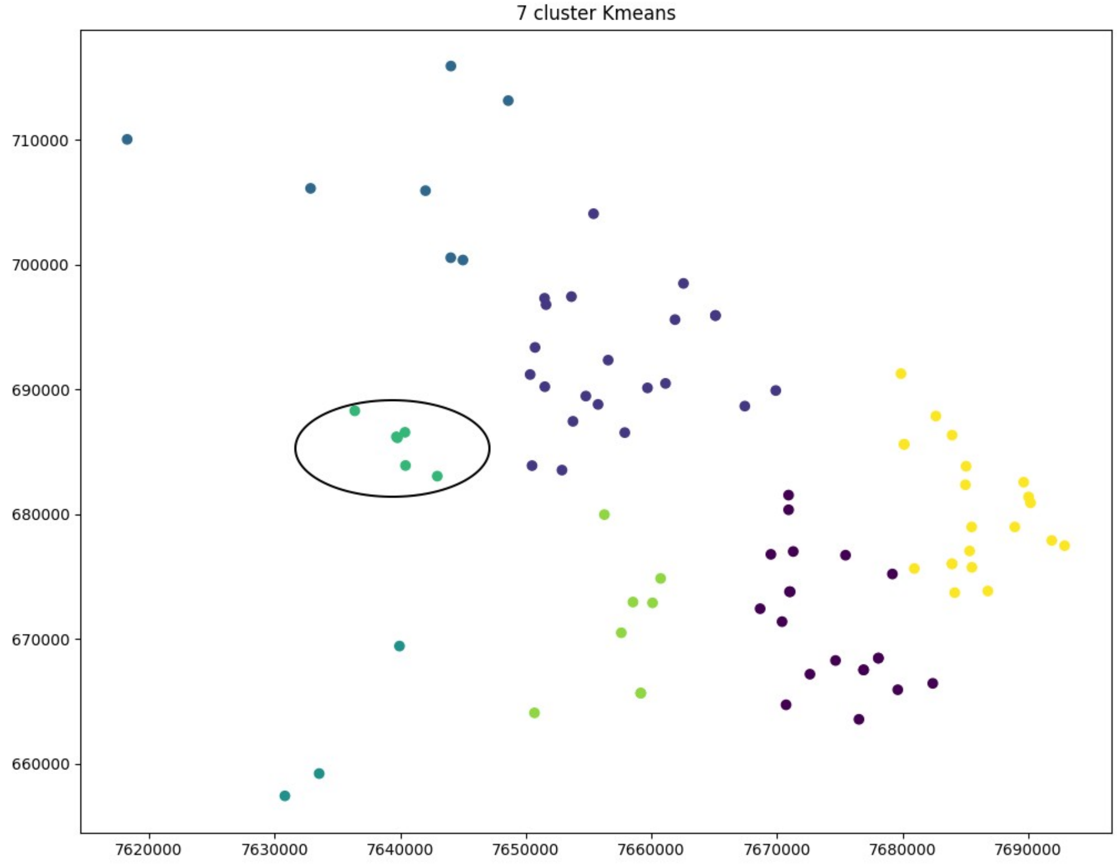
```
number of clusters:  5
Count: 22  -- current area:  240.59295  -- PAI: 4.58658396926
Count: 7  -- current area:  471.51394  -- PAI: 0.74465149941
Count: 25  -- current area:  445.41738  -- PAI: 2.81528531424
Count: 10  -- current area:  613.040221  -- PAI: 0.818202111813
Count: 23  -- current area:  471.456048  -- PAI: 2.44701251119
```



```
number of clusters:  6
Count: 22  -- current area:  471.456048  -- PAI: 2.34062066288
Count: 22  -- current area:  240.59295  -- PAI: 4.58658396926
Count: 7  -- current area:  73.860615  -- PAI: 4.75373190995
Count: 25  -- current area:  445.41738  -- PAI: 2.81528531424
Count: 4  -- current area:  238.491742  -- PAI: 0.841271566457
Count: 7  -- current area:  471.51394  -- PAI: 0.74465149941
```



```
number of clusters:  7
Count: 20  -- current area:  246.879252  -- PAI: 4.06345044701
Count: 22  -- current area:  401.699796  -- PAI: 2.74707574804
Count: 7  -- current area:  471.51394  -- PAI: 0.74465149941
Count: 3  -- current area:  109.494246  -- PAI: 1.37429359561
Count: 6  -- current area:  34.378245  -- PAI: 8.75421308066
Count: 8  -- current area:  159.523012  -- PAI: 2.51545302291
Count: 21  -- current area:  228.6063  -- PAI: 4.60766255016
```

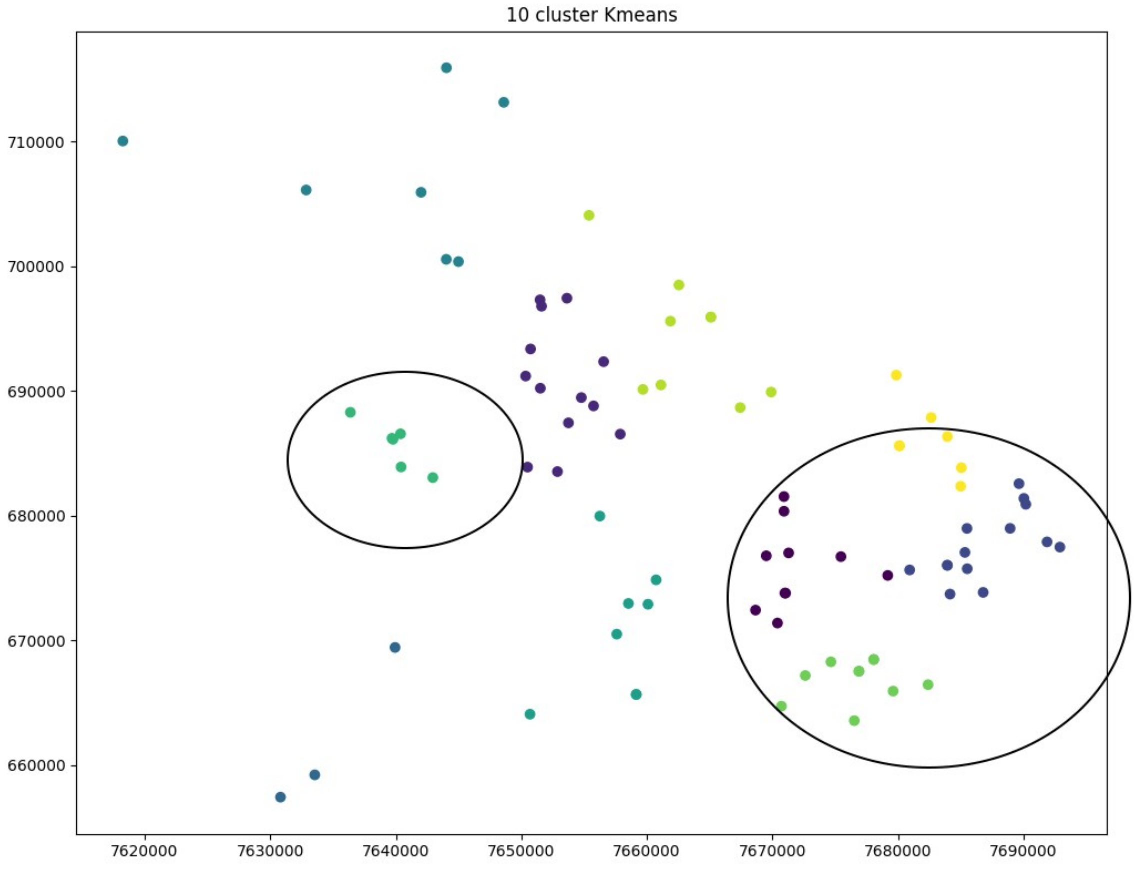


If you review all plots, 4 out of 6 clusters point the right bottom area as hot spot while 2 out of 6 point the middle area as hot spot.

The more clusters are assigned the higher the API number probably would be which measn more concentration. However, There should be a limit assigned to it.

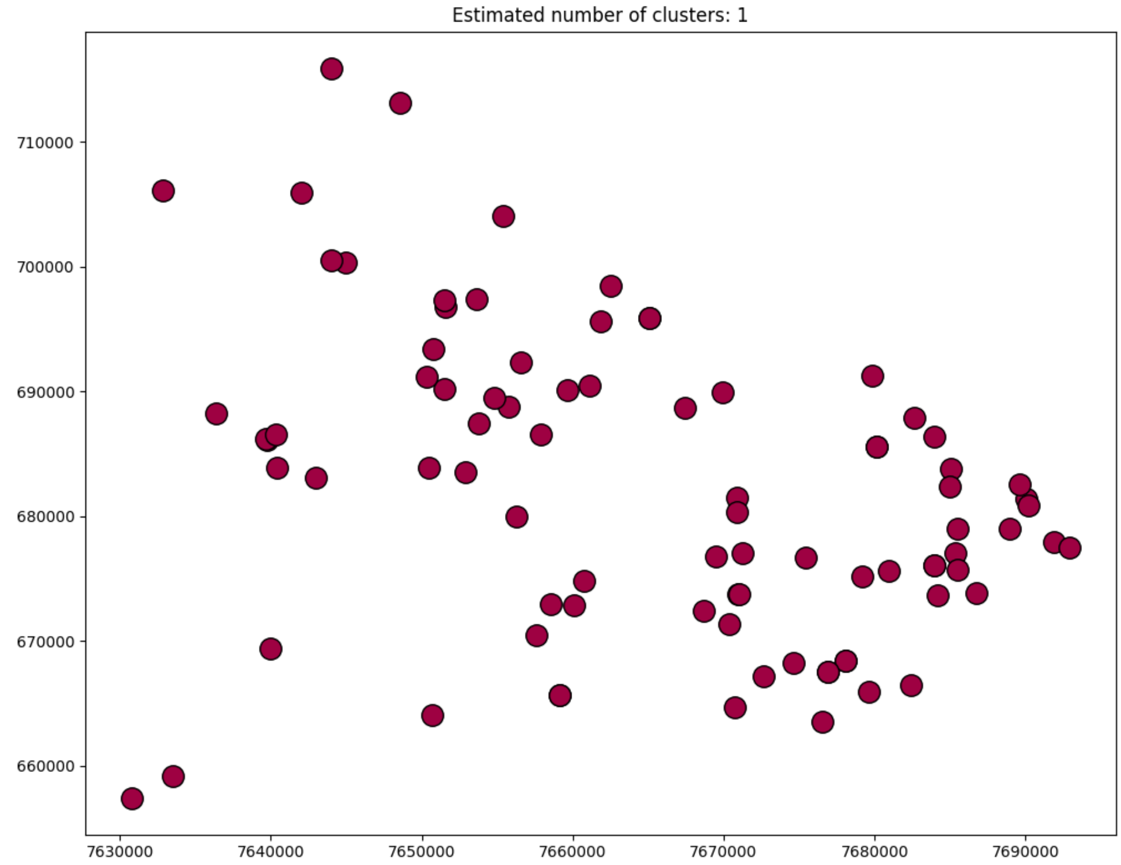
By comparing clustering with 2 to 10 clusters I found 7 clusters the most optimum as the API remained the same in 8,8, and 10 clusters and the one with sever clusters gives the highest API value which is the area at the middle. However,by just taking votes the area on the right corner is selected as the hot spot. Which one is better? I am not sure! That can be debatable.

Here is the plot with 10 clusters and the two areas of concentration.



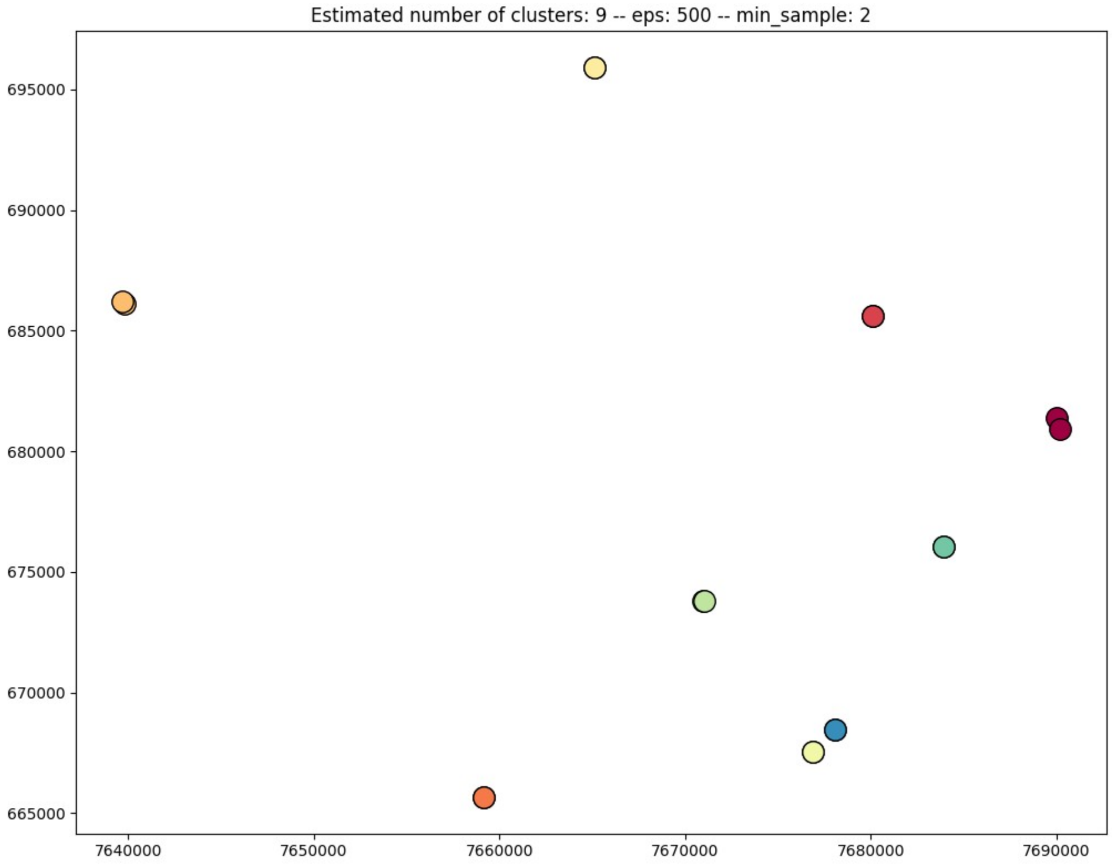
4.2. DBSCAN:

For DBSCAN I ran a number of test. Here is the whole data set of burglary:

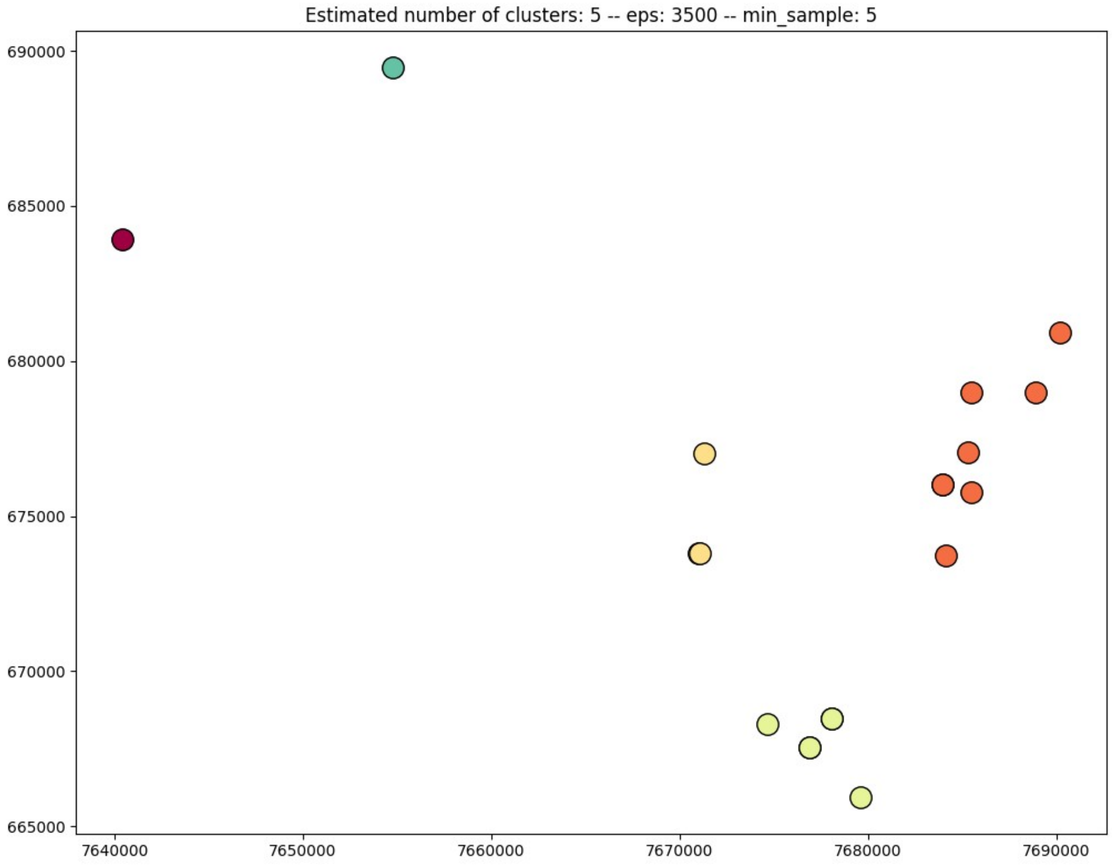


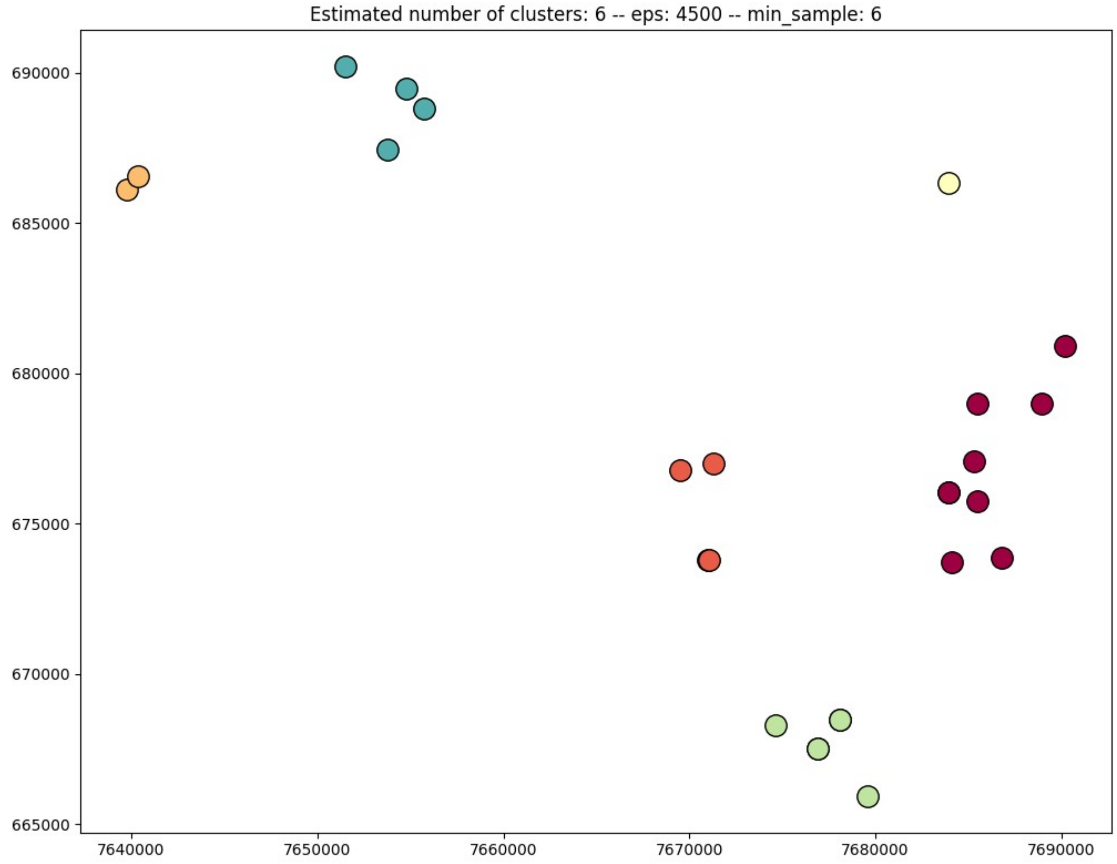
Now playing with the Epsilon value which is the radius of points and minimum data numbers hotspots can be detected:

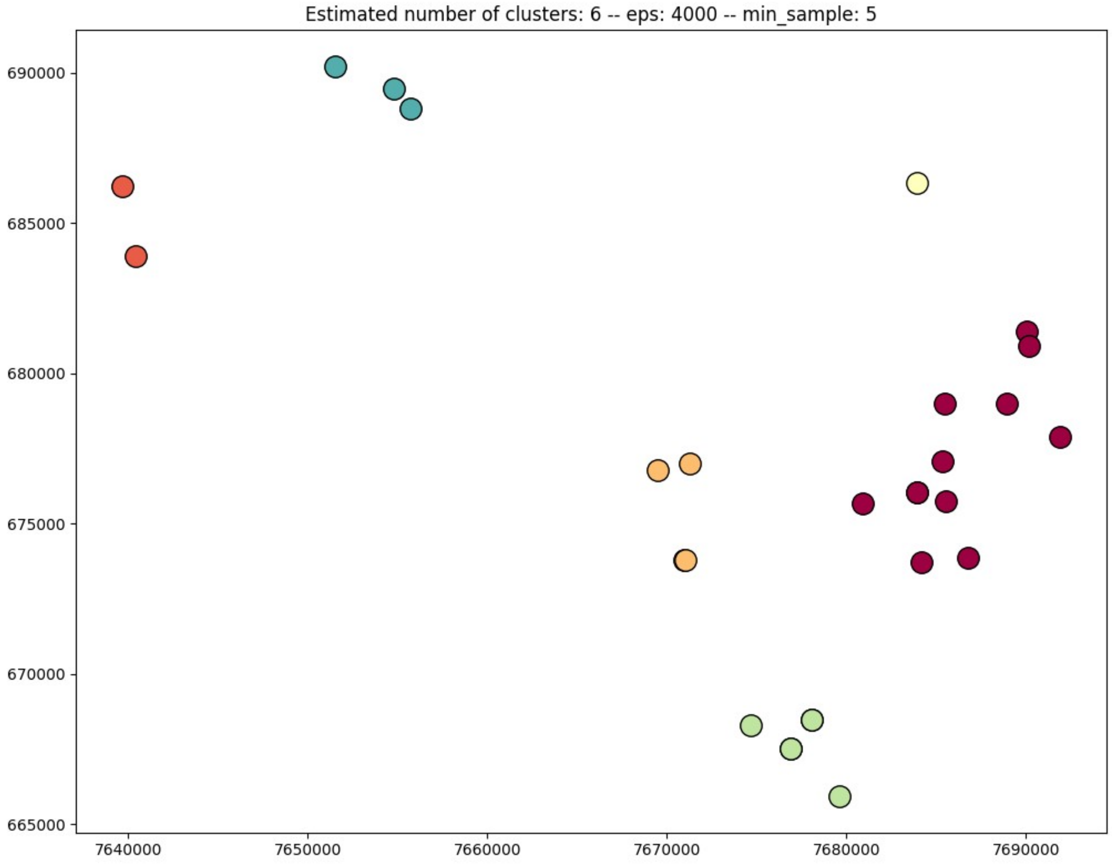
Here is results from Epsilon of 500 feet and 2 data. In other words, any number of points (at least 2) which are 500 or less far from each other can be considered a hot spot. The outcome is 9 clusters which might not be realistic as the minimum number of data is set very low. Increasing that number might help focusing on fewer areas instead of many.



Here is data for 5 clusters and 3500 feet distance. A few set of clusters on each corner is detected.

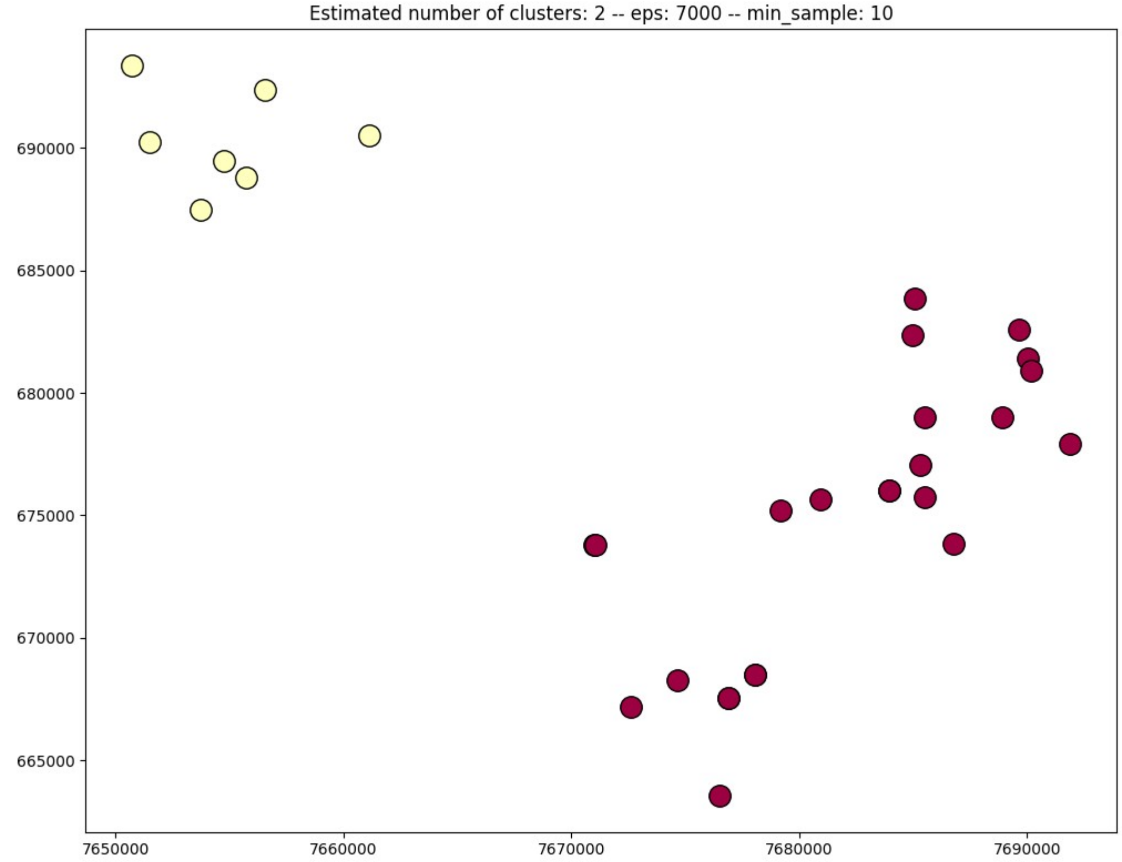



Estimated number of clusters: 6 -- eps: 4500 -- min_sample: 6
690000
685000
680000
675000
670000
665000
7640000
7650000
7660000
7670000
7680000
7690000


Estimated number of clusters: 6 -- eps: 4000 -- min_sample: 5
690000
685000
680000
675000
670000
665000
7640000
7650000
7660000
7670000
7680000
7690000

The best results has gotten by min 10 samples and 7000 feet distance. The best metric to decide which clustering is closer to actual value depends directly to how a hot spot is defined by user.

All can be concluded is that the same pattern of hot spots were discovered in all tests. Clusters were found on the top left and right bottom of almost all graphs. That can be the only metric to be used as comparison.


Estimated number of clusters: 2 -- eps: 7000 -- min_sample: 10