Data Mining (CS524)

Lab Assignment 3

Clustering

Submitted by:
Paras Kumar
2016CSB1047

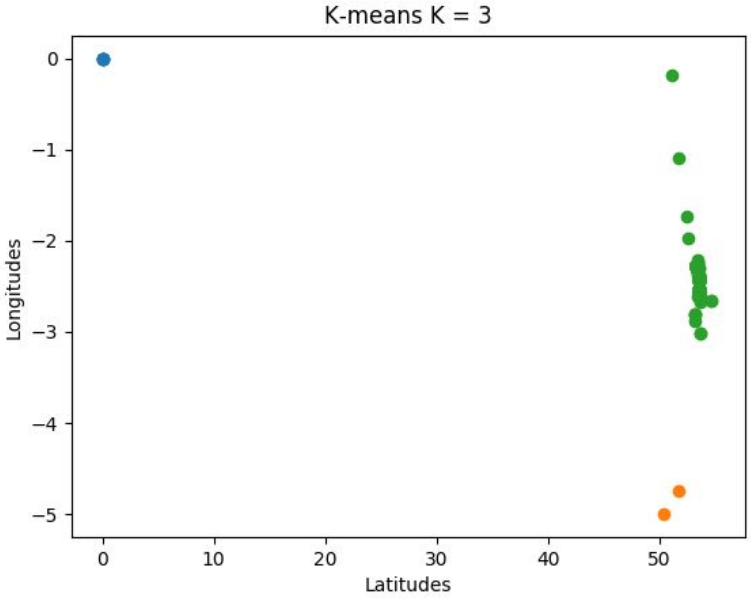
TASK 1

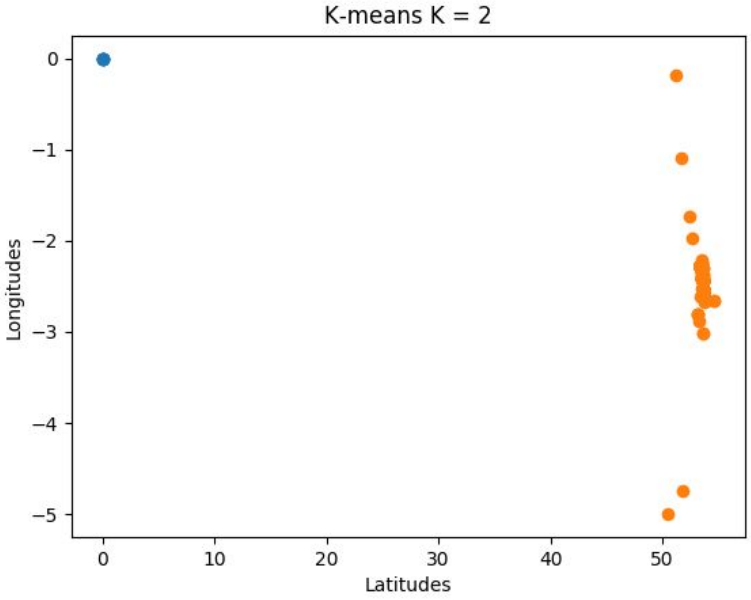
All the four clustering algorithms are implemented in python3 and submitted along with this report. I have followed the algorithms given in the book and in case of Agglomerative clustering, I used distance matrix. The code is modularized and easy to understand.

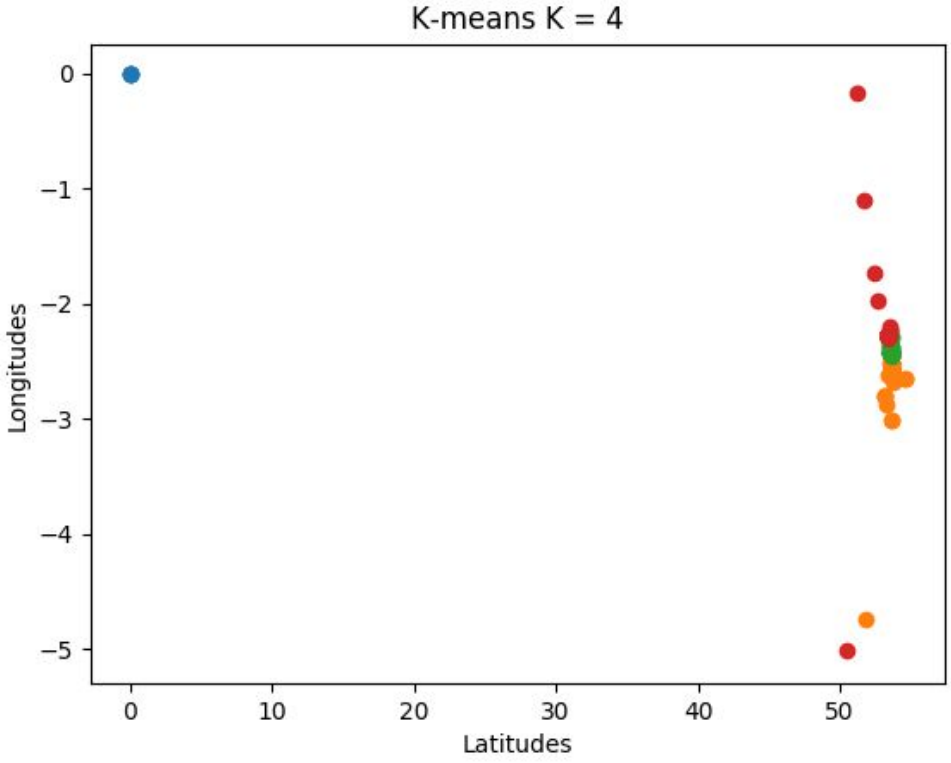
Note 1: I have used Euclidean distance in all the algorithms.

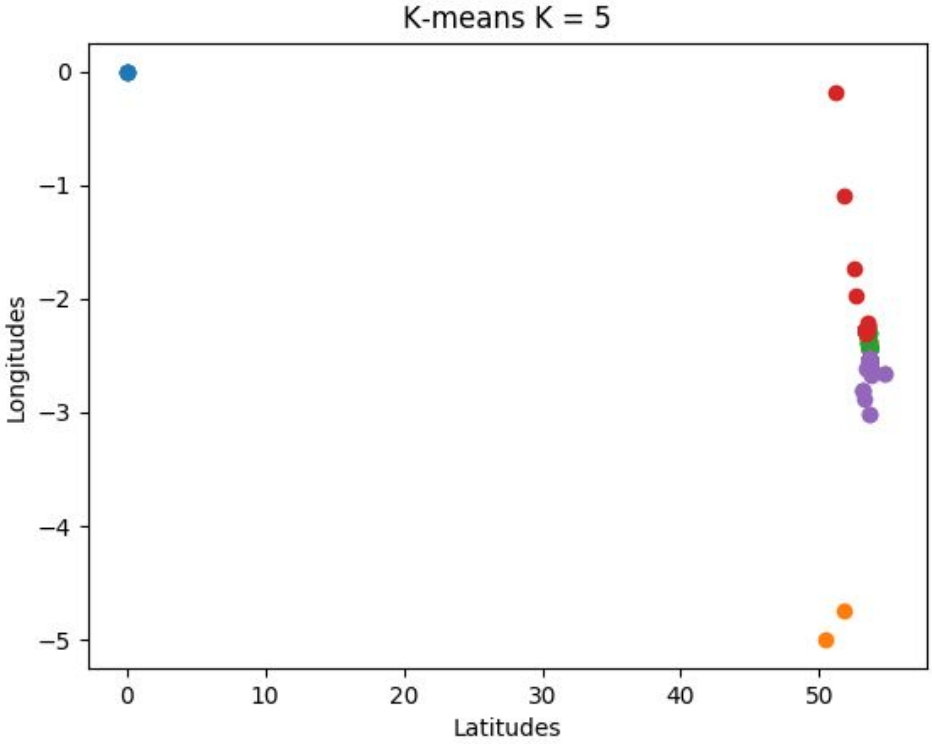
TASK 2

Clusters obtained by K-means algorithm:






K-means K = 4
Longitudes
Latitudes
0
−1
−2
−3
−4
−5
0
10
20
30
40
50

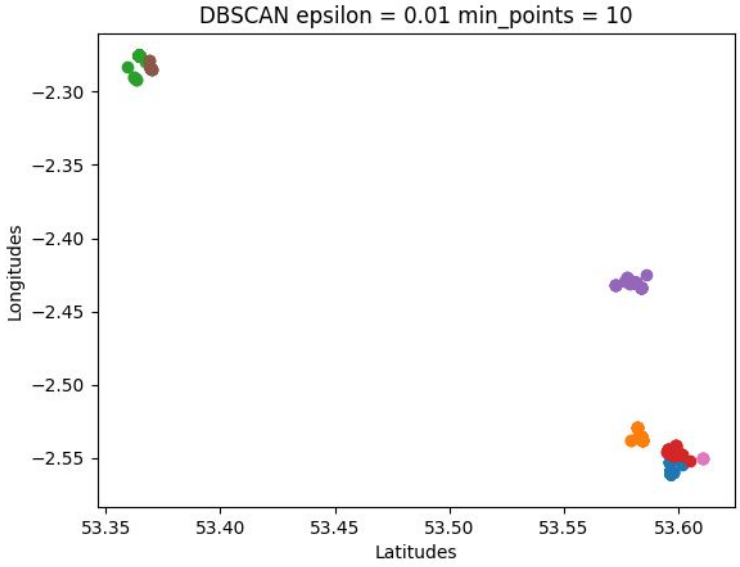

K-means K = 5
Longitudes
Latitudes
0
−1
−2
−3
−4
−5
0
10
20
30
40
50

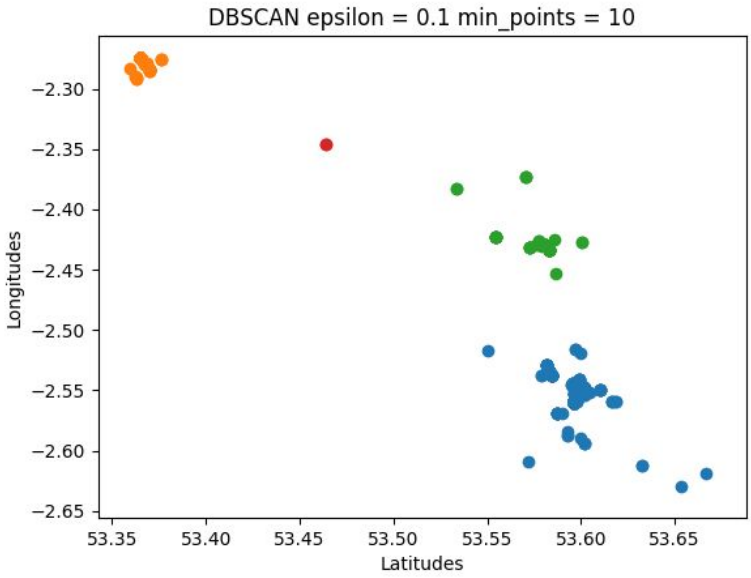
K-means Analysis

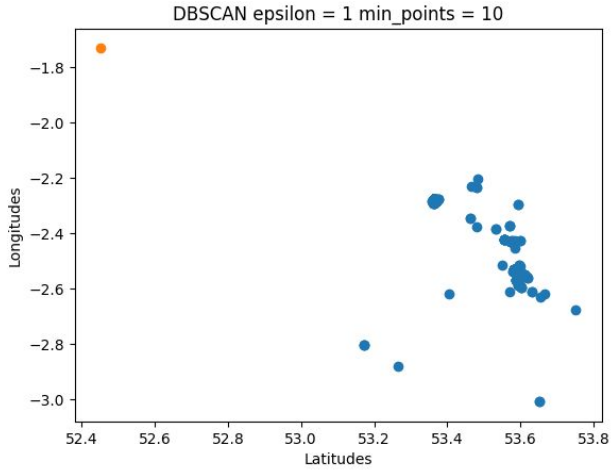
We observe that for K = 5 we get best clustering results with K-means. We also observe the effect of outliers on clusters and also K-means can only find convex clusters, so it leads to some errors which can be fixed by using a density based clustering algorithm like DBSCAN.

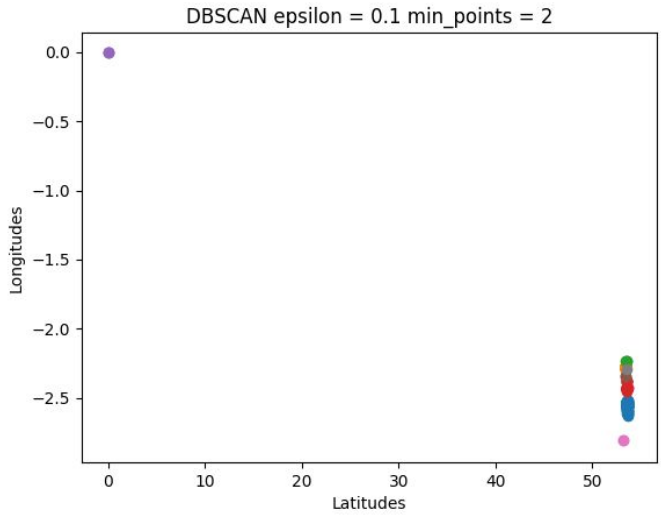
TASK 3

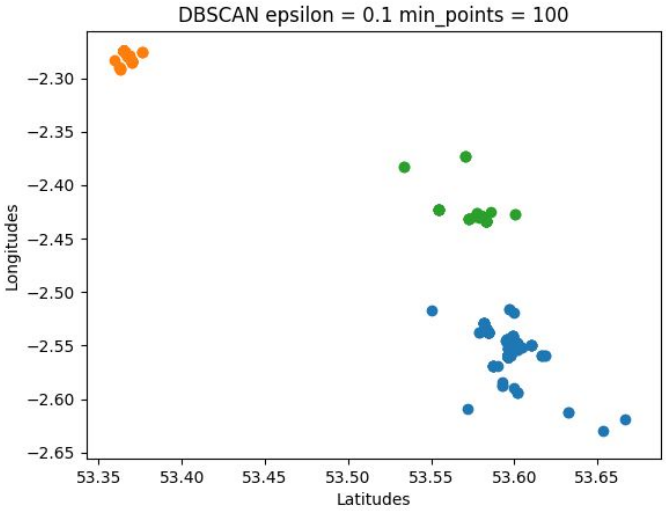
Clustering results with DBSCAN:






DBSCAN epsilon = 1 min_points = 10
Longitudes
Latitudes
−1.8
−2.0
−2.2
−2.4
−2.6
−2.8
−3.0
52.4
52.6
52.8
53.0
53.2
53.4
53.6
53.8


DBSCAN epsilon = 0.1 min_points = 2
Longitudes
Latitudes
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
0
10
20
30
40
50


DBSCAN epsilon = 0.1 min_points = 100
Longitudes
Latitudes
−2.30
−2.35
−2.40
−2.45
−2.50
−2.55
−2.60
−2.65
53.35
53.40
53.45
53.50
53.55
53.60
53.65

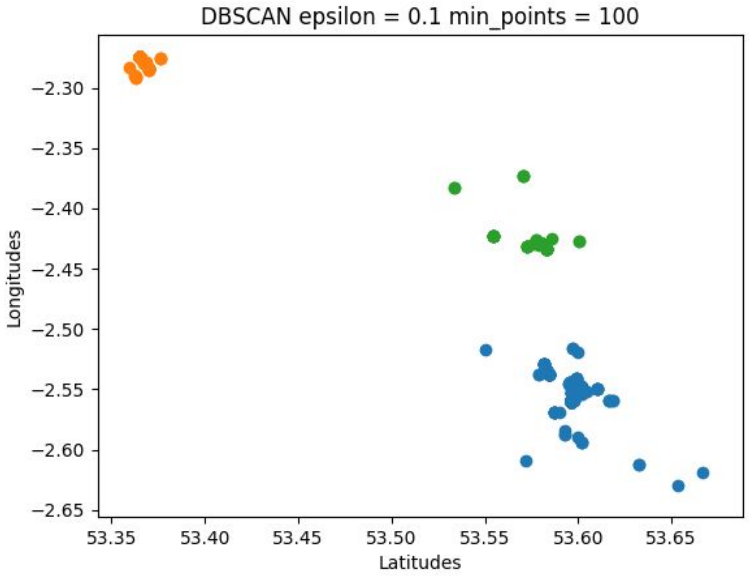
In these graphs I have not shown the noise points so that we can observe the important clusters in the data. DBSCAN produced the best results out of the four algorithms. It filters out noise points and doesn't require number of clusters as input. The best result is obtained for epsilon = 0.1 and min_points = 10.

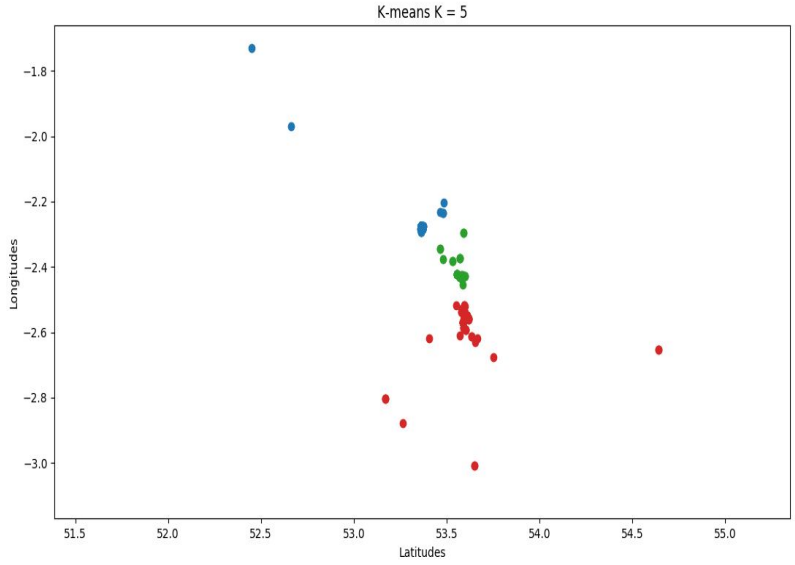
The clusters formed are very sensitive to epsilon values, with it's increase the number of clusters decreases.

We also observe that for very low min_points value, like 2 some outliers also start forming clusters. But, increasing it from 10 to 100 leads to very little change in the clusters.

TASK 4

Comparison of DBSCAN and K-means



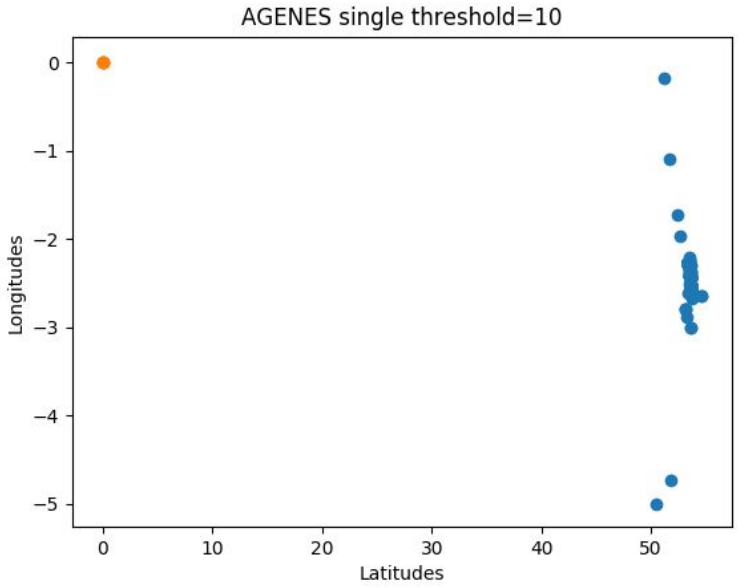


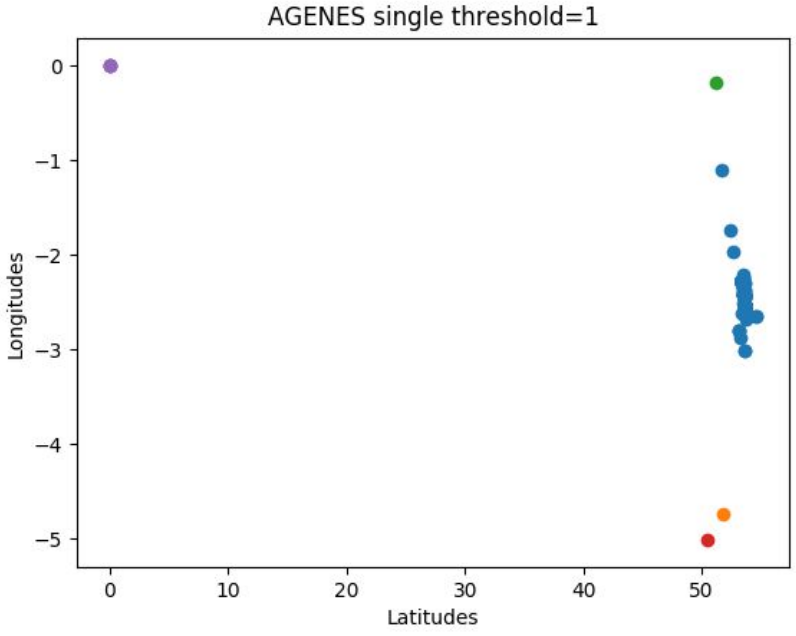
Comparison of DBSCAN and K-means

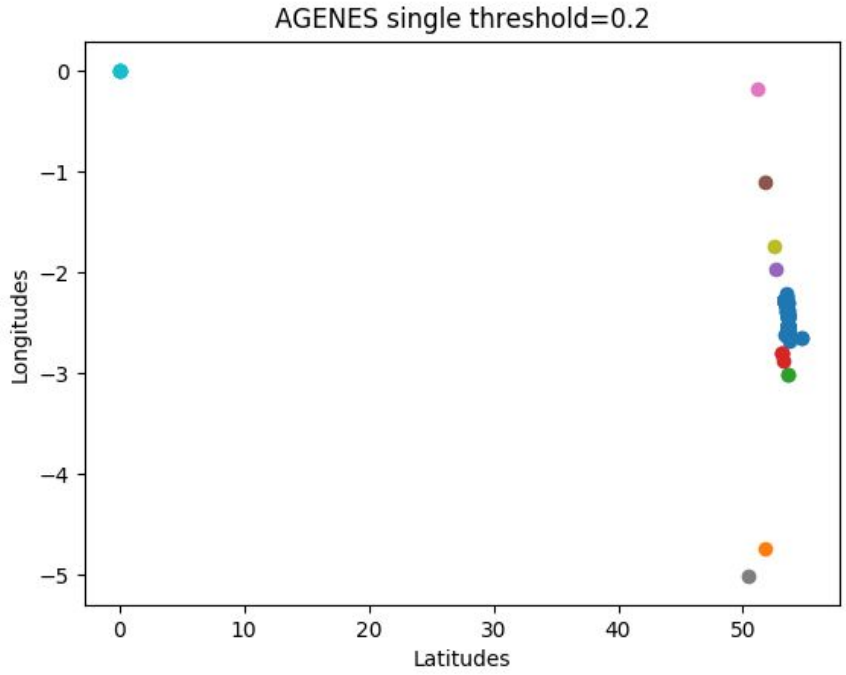
Comparing the best clusters of both the algorithms (zoomed graph in case of K-means) we observe that DBSCAN produces higher quality clusters. The clusters in case of DBSCAN are more dense and the noise points are removed, so it gives us a better visualization of the data. Also, since the clusters are not spherical so K-means fails to produce good quality clusters.

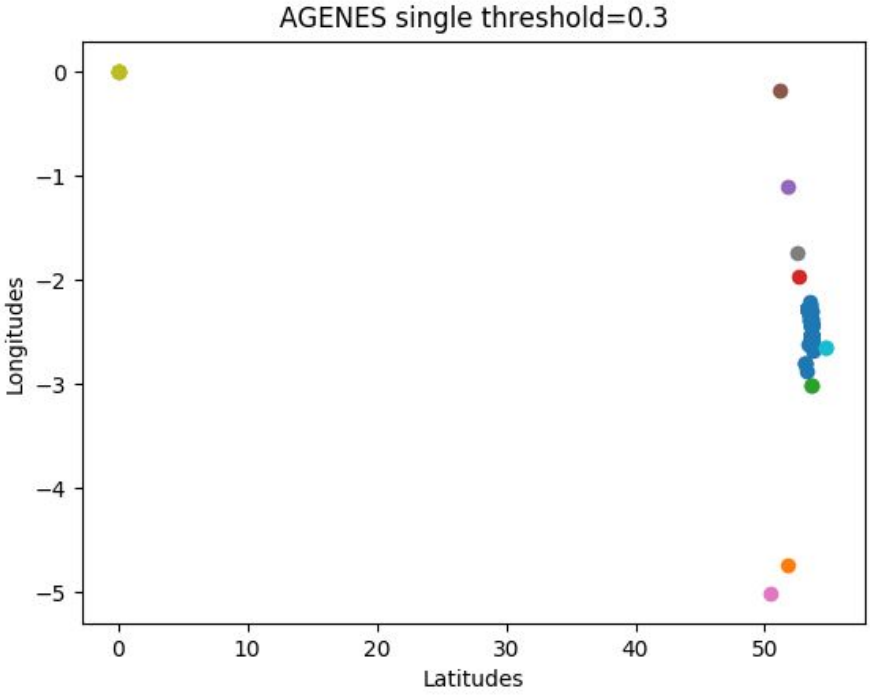
TASK 5

Single Linkage Agglomerative clustering






AGENES single threshold=0.2
Longitudes
Latitudes
0
−1
−2
−3
−4
−5
0
10
20
30
40
50


AGENES single threshold=0.3
Longitudes
Latitudes
0
−1
−2
−3
−4
−5
0
10
20
30
40
50

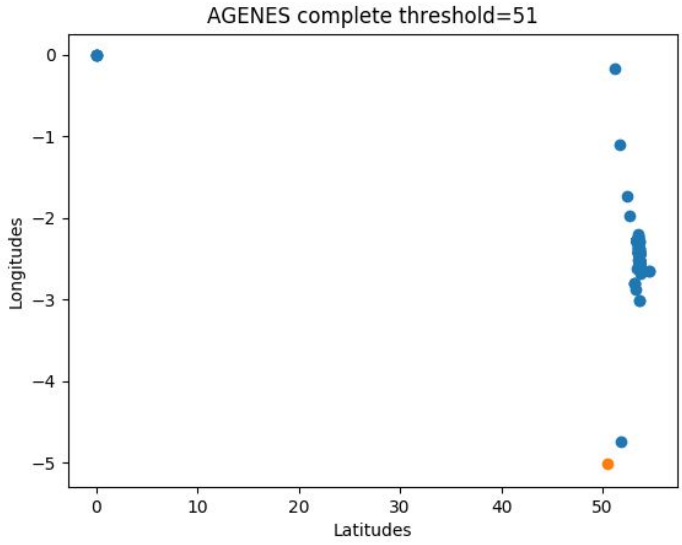
Single linkage agglomerative clustering

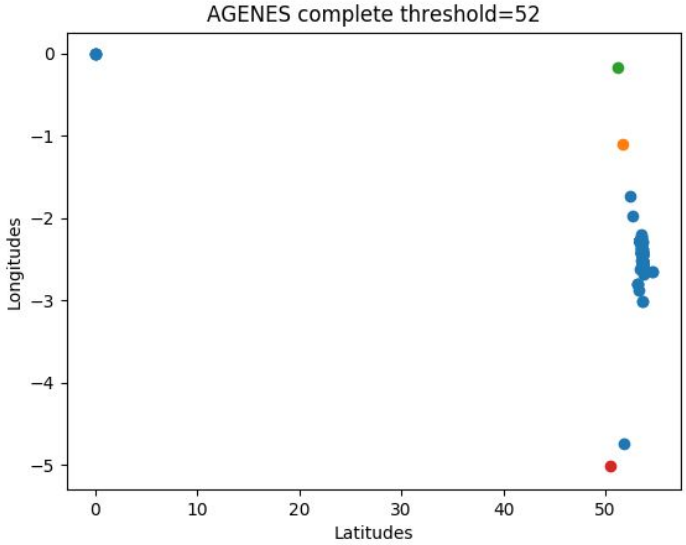
Number of cluster changing with threshold

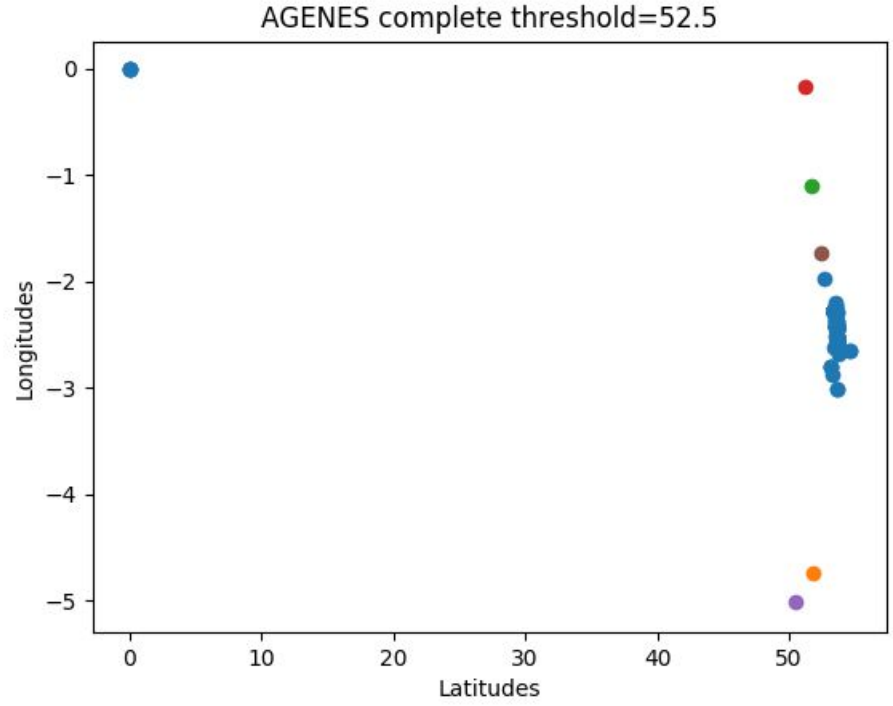
Clusters = 25 val= 0.0306573123740482 08
Clusters = 24 val= 0.0341728571969048
Clusters = 23 val= 0.034267627711879 51
Clusters = 22 val= 0.036093763907355 94
Clusters = 21 val= 0.038528447425246 72
Clusters = 20 val= 0.044564868068918434
Clusters = 19 val= 0.052909154236294 41
Clusters = 18 val= 0.064291209049138 65
Clusters = 17 val= 0.080291920446580 32
Clusters = 16 val= 0.098901037861087 98
Clusters = 15 val= 0.103419161343533 42
Clusters = 14 val= 0.111740727458704 99
Clusters = 13 val= 0.121708887469241 69
Clusters = 12 val= 0.165835832512155 58
Clusters = 11 val= 0.296101690224828 46
Clusters = 10 val= 0.319532770133206
Clusters = 9 val= 0.346183311108435 86
Clusters = 8 val= 0.762499513081811 6
Clusters = 7 val= 0.889673716703490 4
Clusters = 6 val= 0.954252943311154 6
Clusters = 5 val= 1.088190669290084
Clusters = 4 val= 1.376689117386346 3
Clusters = 3 val= 2.372410889012485 8
Clusters = 2 val= 50.695295299837 24

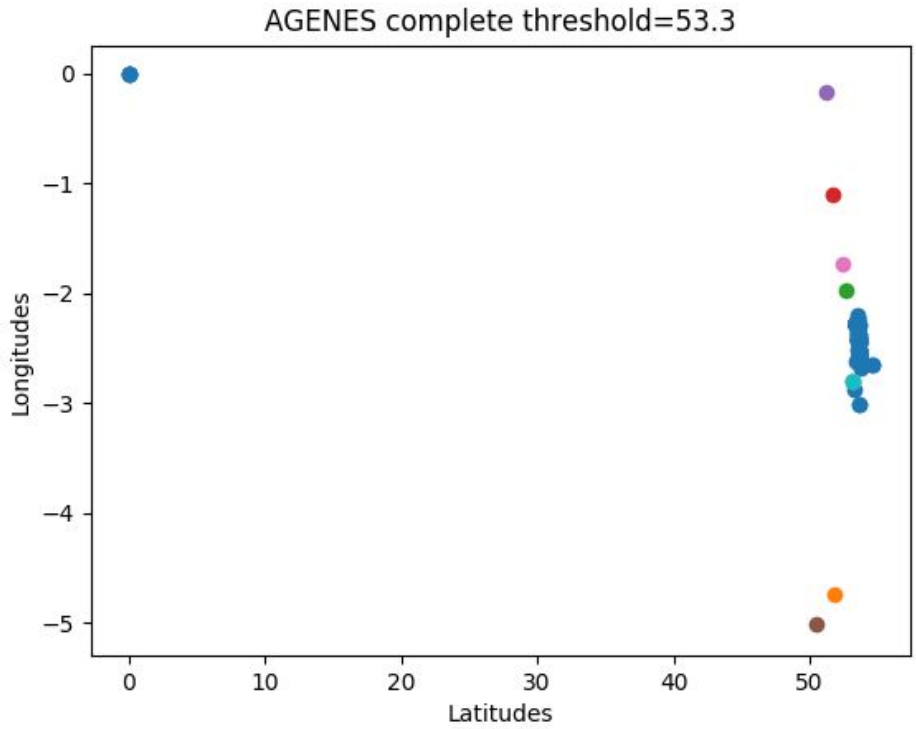
TASK 5

Complete Linkage Agglomerative clustering






AGENES complete threshold=52.5
Longitudes
Latitudes
0
−1
−2
−3
−4
−5
0
10
20
30
40
50

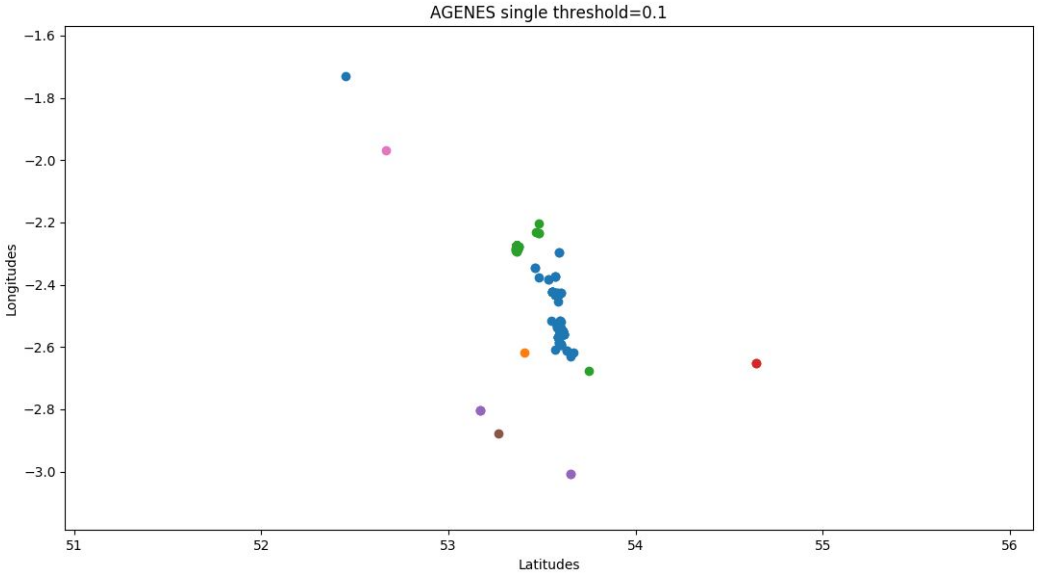

AGENES complete threshold=53.3
Longitudes
Latitudes
0
−1
−2
−3
−4
−5
0
10
20
30
40
50

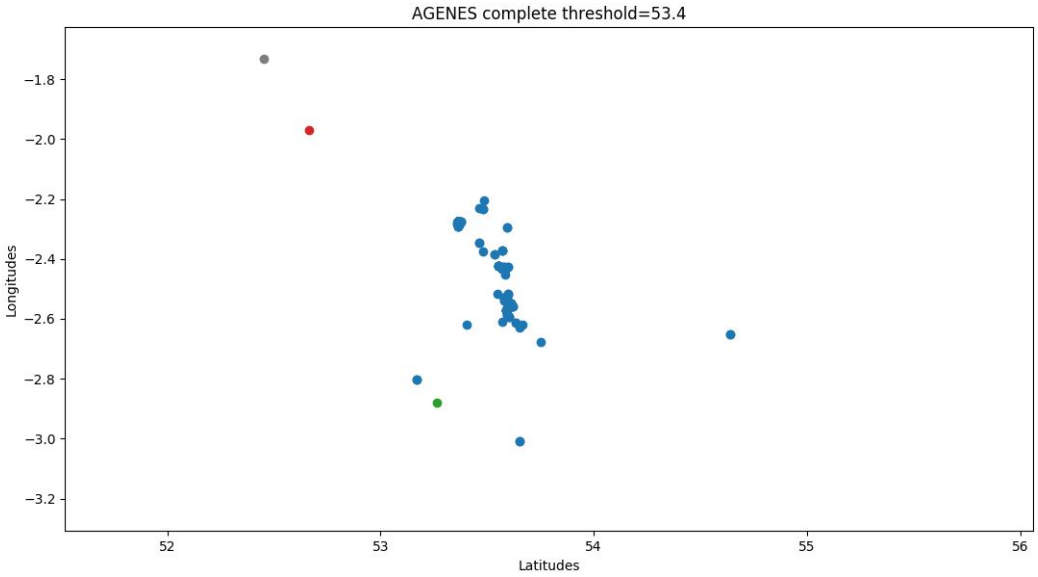
Complete linkage agglomerative clustering

Number of cluster changing with threshold

Clusters = 25 val= 53.41317201627914
Clusters = 24 val= 53.41317201627914
Clusters = 23 val= 53.41317201627914
Clusters = 22 val= 53.41317201627914
Clusters = 21 val= 53.41317201627914
Clusters = 20 val= 53.41317201627914
Clusters = 19 val= 53.4122594242068
Clusters = 18 val= 53.4122594242068
Clusters = 17 val= 53.41212886438529
Clusters = 16 val= 53.41212886438529
Clusters = 15 val= 53.41105255131923
Clusters = 14 val= 53.41105255131923
Clusters = 13 val= 53.40812883928181
Clusters = 12 val= 53.40812883928181
Clusters = 11 val= 53.34386362420635
Clusters = 10 val= 53.245603960901875
Clusters = 9 val= 53.245603960901875
Clusters = 8 val= 53.245603960901875
Clusters = 7 val= 52.70110505946175
Clusters = 6 val= 52.479841547936296
Clusters = 5 val= 52.015916287355836
Clusters = 4 val= 51.74962911351704
Clusters = 3 val= 51.16303404489612
Clusters = 2 val= 50.69529529983724

Comparison of single link and complete link





Comparison of single link and complete link

We observe that complete link gives better results since it tends to minimize the cluster diameter and the original clusters are also compact. The single link fails to identify the clusters as shown in the zoomed graphs.

In general both these algorithms are slow and performs worse than DBSCAN.

TASK 6

Since, the program for this task was running very slow, I randomly selected 1/5th of original data. Also, I observed a problem with the given distance function, it would always result in those 5 users which have min number of transactions because the distance function is not normalized. I have produced the following output following the instructions given in the assignment.

Note: DBSCAN doesn’t not work well for this problem as there are very few points and they are spread apart, they all will be treated as outliers or we can interpret them as a single cluster.

Top 5 closest users and clustering of their check-in data using DBSCAN