**2012003385 Jin Hyun Kim**

**<Programming Assignment #3>**

**1. Algorithm Summary & Environment**

I use python 2.7 and I worked on JetBrains Pycharm Community Edition 2017.1.1 with Windows 10 64bit. I used one python file named ‘clustering.py’ and it was based on the DBSCAN algorithm. In addition to the main function, I have implemented four additional functions as follows.

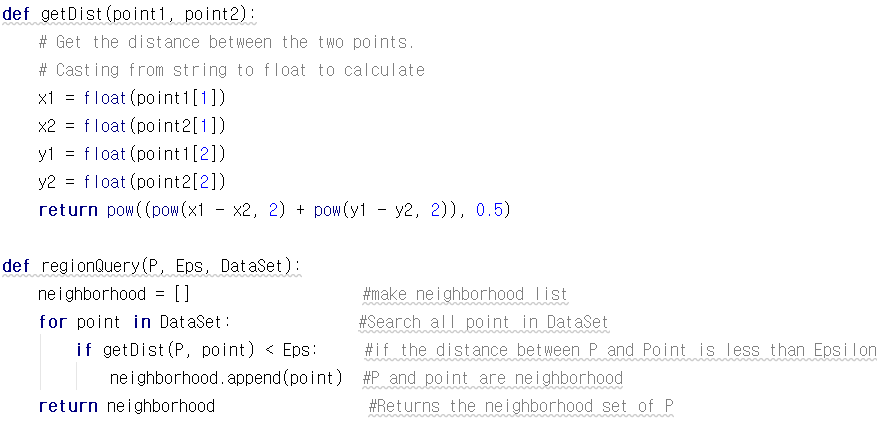
DBSCAN : Perform DBSCAN by receiving dataset, n, Eps, and MinPts as parameters. It also receives input file name and is used to determine the name of the output file.

expandCluster : I extend the cluster by searching for the neighborhood from point P.

regionQuery : Search the dataset and return a list of neighboring points.

getDist : Returns the distance between two points.

**2. Detailed description of codes**



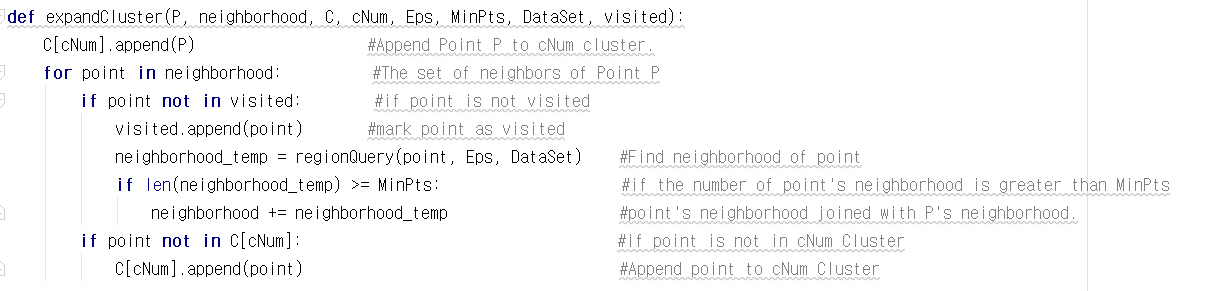
The getDist function returns the euclidean distance between two points.

The regionQuery function returns a set of points(neighborhood) that are within the Epsilon range from a particular point P. Get the distance and search the dataset through the for loop.



First, Search the dataset through the for loop. For each point in dataset, if the point does not exist in the visited list, append it to the visited list. Then, the neighborhood of the point is obtained through the regionQuery function. If the number of neighbors is greater than the MinPts value, I make a new Cluster and increase the number of Cluster. Then, execute the expandCluster function.

Assuming that the cluster is completed by escaping the for loop, we will use only n clusters in order of size. So, assign the key to len function and sort the cluster. Finally, I go through the for statement n times and input the value of object\_id of each cluster into output file.

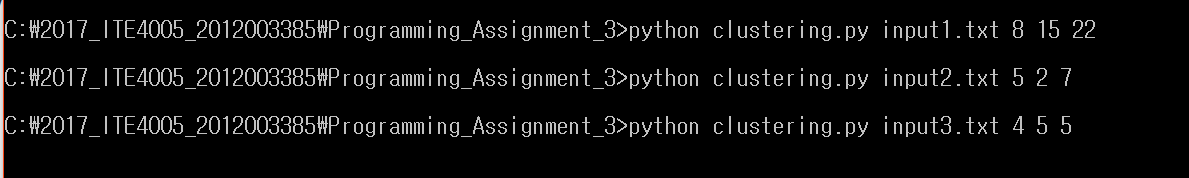


It is a function to expand the cluster by referring to the neighborhood list from a specific point P. If the point in the P-neighborhood is not in the visited list, add it to the visited list. Then the neighborhood of the point is obtained through the regionQuery function, and if the number of point-neighborhood is more than the MinPts number, the P-neighborhood is expanded. Finally, if the point is not in the cNum Cluster, add it.

**3. Instruction for compiling codes**

First, download the zip file form gitlab and extract it.

Then open the commandline and move to the location where ‘clustering.py’ exists and enter the following using data file in the commandline.



Then we can see that the result file is created.

