Data Science

Programming Assignment #3

컴퓨터전공

2013011695 정태화

1. Goal

Perform clustering on a given data set by using DBSCAN.

2. Summary of algorithm

This program goes through following procedure :

1. Read the entire input data.
2. Start clustering using DBSCAN.
3. Sort the result clusters in descending order.
4. Write this cluster result into output text file.

3. Detailed description of code

A. `map<int, pair<bool, pair<double, double> > > readInput(string &)`

This function reads data from given input file, and make each point to map type. Each data is consisted of its id, visited flag(bool), and x-coordinate, y-coordinate.

B. `double getDistance(pair<double, double>, pair<double, double>)`

This function calculates distance between two points.

C. `set<int> getNeighbor(map<int, pair<bool, pair<double, double> > > &, int)`

Gets neighbors of core point within distance Eps. Eps is hyperparameter chosen by user.

D. `vector< set<int> > DBSCAN(map<int, pair<bool, pair<double, double> > > &)`

Arbitrary select a point p, and retrieve all points density-reachable from p using getNeighbor(). If its number of points is more than minpts, it means that p is a core point, and a cluster is formed.

And then, expand this cluster with other points in this cluster until the point is border point, which means it is inside eps, but not core because it does not satisfy minpts.

Continue this process for every points until every points are marked as visited.

E. `vector< set<int> > map_to_set(map<int, int>, int)`

Change cluster result type map to set so that it can be shown as the list of each cluster.

F. `void writeOutput(string &, vector< set<int> >)`

Write each cluster's points list into output text file.

4. Result

- input1 : 98.97037

- input2 : 94.86598

- input3 : 99.97736

5. Instructions for compiling this code

- This project contains `'clustering.cpp'`, `'Makefile'`, `'input1.txt'`, `'input2.txt'`, and `'input3.txt'`.

- In project folder path, just type `'make'` in terminal, or please type below line. This will generate executable file `'clustering'` for linux.

```
$ g++ -O2 -o clustering clustering.cpp --std=c++11
```

- Now you will be able to execute this file with specified arguments.

```
$ ./clustering input1.txt 8 15 22
```

6. Any other specifications

- This code is written in C++11.

- Compiler must support C++11 standard.

- This program is compiled with g++ and xcode.

- This program compilation is tested on macOS High Sierra.