

DBSCAN implement assignment

2013011800 구장회

순서

1. data structure - DBSCAN, Pair, Point class
2. summary of algorithm
3. instruction for compiling

[주의]

[모든 스크린샷은 첨부되어있음]

[실제로는 **dbscan package**에 넣었지만, 제출시에는 **default package**로 했음]

[실행파일은 jar파일로 첨부했음]

data structure

[DBSCAN class]

```
19  /* DBSCAN class
20  *
21  * String input                : path to input file
22  * final int undefined         : final number to point of label(undefined)
23  * final int undefined         : final number to point of label(outlier)
24  * int n                       : # of cluster to be formed
25  * int Eps , MinPts            : Eps , MinPts
26  * int cnum                    : # of all cluster
27  * List<Point> DB               : List of Point
28  * List<Pair> clist             : List of Pair
29  * Map<Integer, List<Point>> cluster : Map of <cluster idx, List of Point>
30  *
31  */
32
33  public class DBSCAN {
34      String input;
35      final int undefined = -10000, noise = -9999;
36      int n, Eps, MinPts, cnum;
37      List<Point> DB;
38      List<Pair> clist;
39      Map<Integer, List<Point>> cluster;
```

[Pair class]

```
284  /*
285   * Pair class
286   *
287   * int id      : cluster idx
288   * int size    : size of cluster
289   *
290   */
291
292  public class Pair implements Comparable<Pair> {
293      int idx, size;
294
295      /* construtor */
296      Pair(int _idx, int _size) {
297          this.idx = _idx;
298          this.size = _size;
299      }
300
301      /* descending order of cluster-size */
302      @Override
303      public int compareTo(Pair o) {
304          // TODO Auto-generated method stub
305          if (size > o.size) {
306              return -1;
307          } else if (size < o.size) {
308              return 1;
309          } else {
310              return 0;
311          }
312      }
313  }
```

[Point class]

```
249  /*
250  * Point class
251  *
252  * int id      : Point idx
253  * int label   : label of a Point(cluster idx / noise / undefined)
254  * double x,y  : x,y_coordinate value
255  *
256  */
257
258  public class Point implements Comparable<Point> {
259      int id, label;
260      double x, y;
261
262      /* constructor */
263      Point(int _id, double _x, double _y) {
264          this.id = _id;
265          this.x = _x;
266          this.y = _y;
267          this.label = undefined;
268      }
269
270      /* ascending order of Point.id */
271      @Override
272      public int compareTo(Point o) {
273          // TODO Auto-generated method stub
274          if (id > o.id) {
275              return 1;
276          } else if (id < o.id) {
277              return -1;
278          } else {
279              return 0;
280          }
281      }
282  }
283
```

summary of algorithms(flow)

[main method]

```
296  /*
297  * main method
298  *
299  * store all parameter as input-filename,n,Eps,MinPts
300  * and call work method after create DBSCAN instances
301  *
302  */
303  public static void main(String[] args) {
304      String input, n, Eps, MinPts;
305      if (args.length != 4) {
306          System.out.println("args error!!!");
307      } else {
308          input = args[0];
309          n = args[1];
310          Eps = args[2];
311          MinPts = args[3];
312          new DBSCAN().work(input, n, Eps, MinPts);
313      }
314  }
315
```

[main workflow]

```
41  /*
42   * work : main process in DBSCAN algorithm
43   *
44   * store all parameter as input-filename,n,Eps,MinPts
45   *
46   * read input-file and store all data(object id , x and y coordinates
47   * process clustering(assign all label of each Point) by DBSCAN algorithm
48   * process clustering(form all cluster)
49   * write result of clustering
50   *
51   * @param   _input      input filename
52   * @param   _n           # of cluster to be formed
53   * @param   _Eps,_MinPts Eps,MinPts
54   *
55   */
56  void work(String _input, String _n, String _Eps, String _MinPts) {
57      input = _input;
58      n = Integer.parseInt(_n);
59      Eps = Integer.parseInt(_Eps);
60      MinPts = Integer.parseInt(_MinPts);
61      DB = new ArrayList<Point>();
62      readInputFile();
63      dbscan();
64      clustering();
65      writeOutputFile();
66      //test();
67  }
```

summary of algorithms(implementatation details)

```
/*
 * DBSCAN algorithm : assign all label of each Point
 *
 * for all Point which has no label,
 *
 * retrieve neighbor of the Point as set.
 * if neighbor's size < MinPts, then assign label as noise.
 * if not, the point being core-point.
 * so increment cluster idx(cnum).
 * and enqueue all neighbor except core-point.
 *
 * for all Point in queue,
 *
 * if the Point's label is noise,
 * then assign label as same cluster idx before.
 * if the Point's label is undefined,
 * then continue.
 * enqueue all neighbor of the Point which are dense.
 *
 */
```

```
void dbscan() {}
```

```
/*
 * clustering : form all cluster
 *
 * make clist as list<(cluster-idx,size of cluster)>.
 * make cmap as map<cluster-idx,list of Points>.
 *
 */
```

```
void clustering() {}
```

```

/*
 * write clustering result in output file
 */
void writeOutputFile {}

/*
 * RangeQuery : find neighbor
 *
 * for all Point q,
 * if dist(p,q) <= Eps, add to set of Point(neighbor)
 *
 * return set of neighbors
 *
 *
 * @param      p                Point as a center.
 * @return Set<Point>      set of neighbors
 */
Set<Point> RangeQuery(Point p) {}

{
/*
 * read inputfile
 */
void readInputFile {} } {

/*
 * distance between two Points as Euclidean distance in 2d.
 *
 */

double dist(Point p, Point q) {}

```

instruction for compiling

[Environments]

OS : Mac OS

Language : java

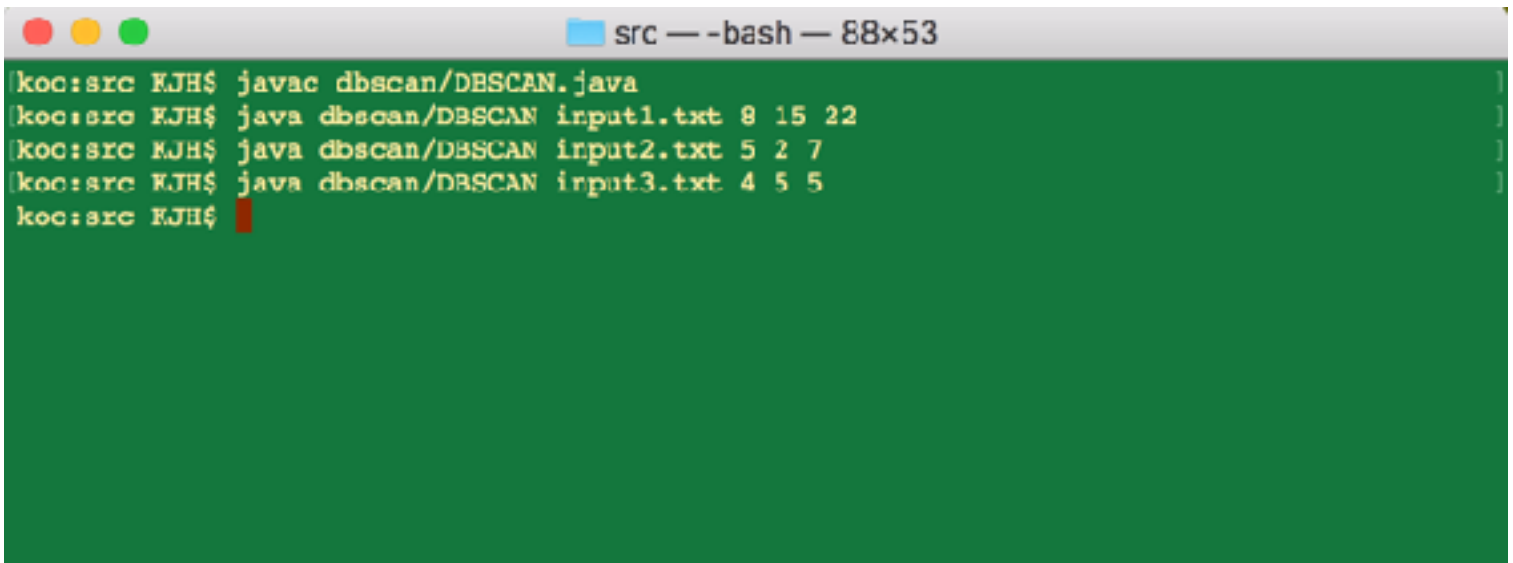
[Screenshot-실행전]

```
src — -bash — 88x53
[koo:src KJH$ pwd
/Users/KJH/eclipse-workspace/DataScience/src
[koo:src KJH$ ls -al
total 1152
drwxr-xr-x 42 KJH  staff   1.3K May 10 21:05 .
drwxr-xr-x  9 KJH  staff   288B Apr  9 16:04 ..
-rw-r--r--@ 1 KJH  staff    10K May 10 19:59 .DS_Store
drwxr-xr-x 10 KJH  staff   320B May  9 22:24 DecisionTree
drwxr-xr-x  8 KJH  staff   256B May 10 20:05 dbscan
-rwxr-xr-x@ 1 KJH  staff   210K May  9 23:44 input1.txt
-rw-r--r-- 1 KJH  staff    7.5K May 10 02:55 input1_cluster_0.txt
-rwxr-xr-x@ 1 KJH  staff    8.3K Mar 29 2014 input1_cluster_0_ideal.txt
-rw-r--r-- 1 KJH  staff    7.0K May 10 02:55 input1_cluster_1.txt
-rwxr-xr-x@ 1 KJH  staff    6.4K Apr  7 2011 input1_cluster_1_ideal.txt
-rw-r--r-- 1 KJH  staff    6.7K May 10 02:55 input1_cluster_2.txt
-rwxr-xr-x@ 1 KJH  staff    1.0K Apr  7 2011 input1_cluster_2_ideal.txt
-rw-r--r-- 1 KJH  staff    4.3K May 10 02:55 input1_cluster_3.txt
-rwxr-xr-x@ 1 KJH  staff    8.9K Apr  7 2011 input1_cluster_3_ideal.txt
-rw-r--r-- 1 KJH  staff    4.1K May 10 02:55 input1_cluster_4.txt
-rwxr-xr-x@ 1 KJH  staff    8.3K Apr  7 2011 input1_cluster_4_ideal.txt
-rw-r--r-- 1 KJH  staff   943B May 10 02:55 input1_cluster_5.txt
-rwxr-xr-x@ 1 KJH  staff    2.0K Apr  7 2011 input1_cluster_5_ideal.txt
-rw-r--r-- 1 KJH  staff   883B May 10 02:55 input1_cluster_6.txt
-rwxr-xr-x@ 1 KJH  staff    1.0K Apr  7 2011 input1_cluster_6_ideal.txt
-rw-r--r-- 1 KJH  staff   847B May 10 02:55 input1_cluster_7.txt
-rwxr-xr-x@ 1 KJH  staff    7.9K Apr  7 2011 input1_cluster_7_ideal.txt
-rwxr-xr-x@ 1 KJH  staff    57K Apr 25 2015 input2.txt
-rw-r--r-- 1 KJH  staff    2.8K May 10 02:55 input2_cluster_0.txt
-rwxr-xr-x@ 1 KJH  staff    2.6K Mar 26 2014 input2_cluster_0_ideal.txt
-rw-r--r-- 1 KJH  staff    2.1K May 10 02:55 input2_cluster_1.txt
-rwxr-xr-x@ 1 KJH  staff    2.1K Mar 26 2014 input2_cluster_1_ideal.txt
-rw-r--r-- 1 KJH  staff    1.7K May 10 02:55 input2_cluster_2.txt
-rwxr-xr-x@ 1 KJH  staff    1.3K Mar 26 2014 input2_cluster_2_ideal.txt
-rw-r--r-- 1 KJH  staff    1.0K May 10 02:55 input2_cluster_3.txt
-rwxr-xr-x@ 1 KJH  staff    1.1K Mar 26 2014 input2_cluster_3_ideal.txt
-rw-r--r-- 1 KJH  staff   853B May 10 02:55 input2_cluster_4.txt
-rwxr-xr-x@ 1 KJH  staff    3.5K Mar 26 2014 input2_cluster_4_ideal.txt
-rwxr-xr-x@ 1 KJH  staff    60K Apr 25 2015 input3.txt
-rw-r--r-- 1 KJH  staff    2.6K May 10 02:56 input3_cluster_0.txt
-rwxr-xr-x@ 1 KJH  staff    2.7K Mar 26 2014 input3_cluster_0_ideal.txt
-rw-r--r-- 1 KJH  staff    2.2K May 10 02:56 input3_cluster_1.txt
-rwxr-xr-x@ 1 KJH  staff    2.7K Mar 26 2014 input3_cluster_1_ideal.txt
-rw-r--r-- 1 KJH  staff    2.2K May 10 02:56 input3_cluster_2.txt
-rwxr-xr-x@ 1 KJH  staff    3.2K Mar 26 2014 input3_cluster_2_ideal.txt
-rw-r--r-- 1 KJH  staff    2.2K May 10 02:56 input3_cluster_3.txt
-rwxr-xr-x@ 1 KJH  staff    2.7K Mar 26 2014 input3_cluster_3_ideal.txt
koo:src KJH$
```

[screenshot-컴파일, 실행]

컴파일 : `$ javac dbscan/DBSCAN.java`

실행 : `$ java dbscan/DBSCAN [inputfile] [n] [Eps] [MinPts]`

A screenshot of a terminal window with a green background. The window title bar shows 'src — -bash — 88x53'. The terminal contains the following commands and their outputs:

```
[koc:src KJH$ javac dbscan/DBSCAN.java ]
[koc:src KJH$ java dbscan/DBSCAN input1.txt 8 15 22 ]
[koc:src KJH$ java dbscan/DBSCAN input2.txt 5 2 7 ]
[koc:src KJH$ java dbscan/DBSCAN input3.txt 4 5 5 ]
[koc:src KJH$ ]
```