**CSC 381 Project #9.2 <IsoData Clustering>(Java)**

**Jian Chen**

**Due date: 04/21/2017**

**Submit Date: 04/21/2017**

**Algorithm Steps for the implementation for this project:**

step 0:

- Open the input file

- K ask the use from console

- numRow, numCol numPts get from input file.

- Dynamically allocate all 1-D and 2-D arrays accordingly

Step 1: call loadPointSet

Step 2: call assignLabel

Step 3: call mapPoint2Image

Step 4: call PrettyPrint // output to output-2

step 5: For each subset, compute the centroid, XX, YY, XY moments

// equations are given in class

step 6: determine the “angle” of the best fitted line passing the centroid.

// equations are given in class

Step 7: repeat step 5 and step 6 until all K subsets are processed.

step 8: 8.1: for each point, p(x,y), in the pointSet array

compute the K orthogonal distance, dist,

from p(x, y) to all best fitted lines.

// similar to the distance equation of Hough transform.

// In Hough transform, the orthogonal distance is computed from the // origin to the line of a given angle.

// Here, the origin is (x, y), so, you just need to use x and y as // the origin. Or, you may look online for the distance equation from // a point to a line of given angle.

8.2: min\_i <-- determine which dist(p,ci) is minimum

8.3: if min\_i is not the same as p's old label

change p's label to min\_i

And changeLabel set to true

step 9: repeat step 8 until all points in pointSet are process.

Step 10: repeat step 3 to step 9 until no point changes its label.

Step 11: Output the info of pointSet (x, y, lable) to Output-1 file.

Step 12: closed all files