Northeastern University
CS6220 – Data Mining Techniques
Fall 2017, Derbinsky

Homework 2, Part 2

This is the second part of Homework 2, which focuses on solving small datasets using three clustering algorithms. For this part of the assignment you will submit a single PDF (`p2.pdf`).

1 K-Means

You are to work out, by hand, the K-Means algorithm for a small dataset. In particular, you are to cluster the following 8 data points ...

A $(4, 9)$

B $(2, 10)$

C $(1, 2)$

D $(2, 5)$

E $(6, 4)$

F $(8, 4)$

G $(7, 5)$

H $(5, 8)$

around 3 centroids assuming **Forgy** initialization ...

α $(2, 10)$

β $(1, 2)$

γ $(5, 8)$

For the first iteration, show **all** work, including distance calculation (assume Euclidean), centroid assignment, and calculation of the new centroid locations.

For **all** iterations, until convergence, plot on a 2D, 10×10 grid the eight data points (colored and/or shaped per their associated centroid), and the three centroids.

2 Agglomerative Hierarchical

You are to complete, by hand, the example in the class slides, using **all three inter-cluster measures: MIN, MAX, & AVG**. In particular, you are to cluster the following 6 data points ...

p1 $(0.40, 0.53)$

p2 $(0.21, 0.38)$

p3 $(0.35, 0.32)$

p4 $(0.26, 0.19)$

p5 $(0.08, 0.41)$

p6 $(0.45, 0.30)$

For each iteration, include the portion of the proximity matrix that is appropriate, as well as distances between clusters; the two clusters chosen to be merged; as well as the distance between those clusters. You do not need to include the final iteration (i.e. when there are only two clusters remaining). However, note that the slides included rounded approximations of the first iteration – re-compute these values, such that you do not make a mistake due to lost precision.

3 DBSCAN

You are to cluster, and visualize, a small dataset using DBSCAN ($\epsilon = 7.5, MinPts = 3$). You have been provided a file, `dbscan.csv`, that has the following columns for each point in the dataset:

cluster originally empty, provided for your convenience

pt a unique id for each data point

x point x-coordinate

y point y-coordinate

num_neighbors number of neighbors, according to the coordinates above

neighbors the id's of all neighbors within ϵ

As you can see, a tedious $\mathcal{O}(n^2)$ portion of the work has been done for you. Your job is to execute, point-by-point, the DBSCAN algorithm, logging your work. For example ...

pt 0: $2 < MinPts$, so cluster=-1

pt 1: $3 \geq MinPts$, so cluster=0
`to_visit`=$[40, 75]$, `visited`=$\{1\}$

- **pt 40**: cluster=0, $3 \geq MinPts$, so adding neighbors
  `to_visit`=$[75, 28]$, `visited`=$\{1, 40\}$
- **pt 75**: cluster=0, $3 \geq MinPts$, so adding neighbors
  `to_visit`=$[28, 4]$, `visited`=$\{1, 40, 75\}$
- **pt 28**: cluster=0, $3 \geq MinPts$, so adding neighbors
  `to_visit`=$[4, 12]$, `visited`=$\{1, 28, 40, 75\}$
- **pt 4**: cluster=0, $3 \geq MinPts$, so adding neighbors
  `to_visit`=$[12, 56]$, `visited`=$\{1, 4, 28, 40, 75\}$
- **pt 12**: cluster=0, $2 < MinPts$
  `to_visit`=$[56]$, `visited`=$\{1, 4, 12, 28, 40, 75\}$
- **pt 56**: cluster=0, $3 \geq MinPts$, so adding neighbors
  `to_visit`=$[66]$, `visited`=$\{1, 4, 12, 28, 40, 56, 75\}$
- **pt 66**: cluster=0, $2 < MinPts$
  `to_visit`=$[]$, `visited`=$\{1, 4, 12, 28, 40, 56, 66, 75\}$

pt 2: $1 < MinPts$, so cluster=-1

pt 3: $1 < MinPts$, so cluster=-1

pt 4: cluster=0, so skip

...

When you have completed your log, compile a list of points (sorted by id) in each of the clusters. Cluster -1 is special, representing noise points. After that comes cluster 0, 1, ...

Now, visualize the clusters using a different color for each found cluster (excluding noise). It may help to create circles of radius=ϵ to better see the result.

Extra Credit: it turns out that the solution provides a hint to one of your professor's favorite magazines. Find the current issue of this magazine (available digitally), look to the table of contents, and find the first article, which has a single author. Provide that author's name, place of employment, and three interesting facts.