University of Tehran

School of Electrical and Computer Engineering

Pattern Recognition

Assignment 7

Due Date: 5th of Tir

Corresponding TAs:

Maryam Ahmadinejad - ahmadinejad.mry@ut.ac.ir

Jamshid Hassanpour - hassanpourjamshid@ut.ac.ir

Khordad 98

Use dataset ‘DS1’ (uploaded on course page) for problems 1-4.

PROBLEM 1

Implement Agglomerative hierarchical clustering using predefined python libraries (scikit-learn).

a. Report Confusion matrix.
b. Compare mean distances in clusters and report accuracy.

PROBLEM 2

Implement Sequential clustering using predefined python libraries (scikit-learn) and report measures mentioned in previous problem.

PROBLEM 3

K-means is a type of optimization-based clustering in which an objective function should be minimized. It aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a *prototype* of the cluster. Given a set of observations $(x_1, \dots, x_n)$ where each observation is a d-dimensional real vector, k-means clustering aims to partition the n observations into $k \leq n$ sets $S = \{S_1, \dots, S_n\}$ so as to minimize the within-cluster sum of squares (WCSS) (i.e. variance). Formally, the objective is to find:

$$\arg\min_{S} \sum_{i=1}^{k} \sum_{x \in S_i} \left|\left|x - \mu_i\right|\right|^2$$

Where μ_i is the mean of points in S_i.

K-Means Algorithm:

The *K*-means clustering algorithm uses iterative refinement to produce a final result. The algorithms starts with initial estimates for the *K* centroids, which can either be randomly generated or randomly selected from the data set. The algorithm then iterates between two steps:

1. Data assignment step
2. Centroid update step

The algorithm iterates between steps one and two until a stopping criteria is met (i.e., no data points change clusters, the sum of the distances is minimized, or some maximum number of iterations is reached).

Detailed algorithm is as below:

1- Begin: initialize $n, k, \mu_1, \dots, \mu_k$
2- Do classify n samples according to nearest μ_i
3- Recomputed μ_i
4- Until no change in μ_i
5- Return $\mu_1, \dots, \mu_k$
6- End

a. Implement the algorithm.
b. Implement K-means clustering using predefined python libraries (scikit-learn).
c. Compare measures (mean distance and accuracy) for each part.

PROBLEM 4

Cluster validity measures describe the quality of a complete clustering. *Separation Index* is one of such measures that is proportional to the ratio of between to within distance in clusters:

$$SI = \min_{j}\left\{\min_{i(i\neq j)}\left\{\frac{d(S_i,S_j)}{\max_{l} d(S_l,S_l)}\right\}\right\}$$

Where:

$$d(S_i,S_j) = \min\{d(x_i,x_j)|x_i \in S_i, x_j \in S_j\}$$

$$d(S_l,S_l) = \min\{d(x_i,x_j)|x_i,x_j \in S_l\}$$

Calculate this measure for problems 1-3. Report and compare the results. Which method has better performance?

PROBLEM 5

K-means clustering can be used in image compression. It works on clustering specific (K) numbers of colors to represent the image color instead of actual number of colors and in this way, it reduces image size. Obviously, it clusters pixels with colors similar to each other and considers one value for them.

Image "*bird.png*" is uploaded for this assignment. It has dimension of *rows*columns* pixels and each pixel consists three channels of RGB showing color and intensity. Image data can be considered as arrays of [rows*columns, 3]. Using *'image_compression.py'*, find cluster centers for this image as [*centroid*, 3]. You ought to:

Cluster image pixels using predefined python libraries for K-means.

Find suitable value K, report accuracy and attach your '*compress_image.png'* by your report.

NOTES

1. Please make sure you reach the deadline because there would be no extra time available.
2. Late policy would be as bellow:
   - Every student has a budget for late submission during the semester. This budget is two weeks for all the assignments.
   - Late submission more than two weeks may cause lost in your scores.
3. Analytical problems can be solved on papers and there is no need to type the answers. The only thing matters is quality of your pictures. Scanning your answer sheets is recommended. If you are using your smartphones you may use scanner apps such as CamsScanner or google drive application.
4. Simulation problems need report as well as source codes and results. This report must be prepared as a standard scientific report.
5. You have to prepare your final report including the analytical problems answer sheets and your simulation report in a single pdf file.
6. Finalized report and your source codes must be uploaded to the course page as a ".zip" file (not ".rar") with the file name format as bellow:

   PR_Assignment #[Assignment Number]_Surname_Name_StudentID.zip

7. Plagiarisms would be strictly penalized.
8. You may ask your questions from corresponding TAs.