Project Report

Submitted by Sri Chaitra Kareddy.

Data Input:

• Iris-data.csv – contains comma separated 4 values in one line and 150 lines. Each row is considered as one data point of 4 features

Parameters:

* k - the number of desired clusters
* r - the number of random iterations
* 𝜎 - the value to be used to determine weights in spectral clustering

Output:

* A comma separated file containing n integer values, each specifying the cluster of that corresponding row.
* The quantization error printed on the console

Programs:

Part I:

• lloyd.py  
o Read the file for the matrix XT. Performs k-means on XT with given k number of clusters and r number of iterations. Computes the final clustering, calculates and prints the quantization error. Writes to file the final clustering.

Make sure the python file and the input data are in the same directory

Execution: python3 lloyd.py <input\_data\_file> <number\_of\_clusters> <number\_of\_iterations> <output\_file\_name>  
o Example: python3 lloyd.py iris-data 2 10 clusters.csv

• kmeanspp.py  
o Read the file for the matrix XT. Performs kmeans++ on XT with given k number of clusters and r number of iterations. Computes the final clustering, calculates and prints the quantization errors. Writes to file the final clustering  
o Make sure the python file and the input data are in the same directory

o Execution: python3 kmeanspp.py <input\_data\_file> <number\_of\_clusters> <number\_of\_iterations> <output\_file\_name>

o Example: python3 kmeanspp.py iris-data 4 9 clusters.csv

Part II:

• spectral.py  
o Read the file for the matrix XT. Calculates the Lagrange matrix L and computes k eigen vectors and runs kmeanspp on the vectors. Calculates and prints the quantization error.

Writes to file the clustering:  
o Make sure the python file and the input data are in the same directory  
o Execution: python3 spectral.py <input\_data\_file> <number\_of\_clusters> <sigma\_value> <output\_file\_name>  
o Example: python3 spectral.py iris-data 3 2.5 clusters.csv

The value of σ that needs to be used by spectral.py is 500.