CMPUT 307 Lab-7

Clustering

Submission Deadline: **March 24**

You are NOT allowed to import any other package (except for numpy and matplotlib) to finish this lab. Please only submit your own work and give credit to any material you referenced.

PART 1

Implement the kmeans algorithm, test your algorithm on the given data (data_Aggregation.npy) and print out the result. Note: use **vectorized operation** for better efficiency.

a) Given n 2D data points (array of size (n, 2)) and m 2D cluster centers (array of size (m, 2)), calculate the distances from those n points to m centers.

```
distance(data, centers): Calculate the distances from points to
cluster centers.
```

b) Finish the implementation of kmeans. You can use several random points as the initial centers.

```
kmeans(data, n_centers): Divide data into n_centers clusters, return
the cluster centers and assignments.
```

c) The quality of the clustering result can be calculated using the total sum of squared errors (distortion). You can find the formula for this in page 54 of the point cloud slides. Implement the function for calculating distortion.

```
distortion(data, centers, assignments): Calculate the distortion of
the clustering.
```

Answer the following questions.

1. What's the minimum and maximum number of possible clusters when you have n points?
2. Experiment with different numbers of clusters, from minimum to maximum. Record the distortions while you change the parameter for kmeans. Draw a graph to show how distortion changes with different numbers of clusters.
3. What's the lowest possible distortion? When will this happen? Explain your answers.
4. What's the optimal number of clusters? How can we get this number in a program (without any human intervention/interaction)?

PART 2

Andrew Ng et al. proposed a spectral clustering method (link to paper), which uses eigenvectors of matrices derived from the data to cluster points. A description of this method is given below:

Given a set of points $S = \{s_1, \ldots, s_n\}$ in $\mathbb{R}^l$ that we want to cluster into k subsets:

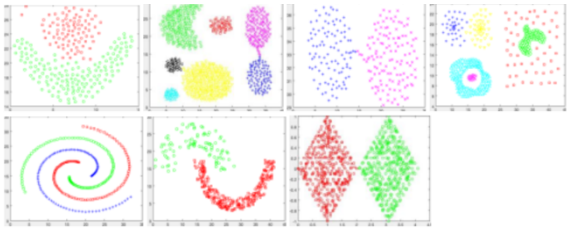
1. Form the affinity matrix $A \in \mathbb{R}^{n \times n}$ defined by $A_{ij} = \exp(-||s_i - s_j||^2/2\sigma^2)$ if $i \neq j$, and $A_{ii} = 0$.
2. Define D to be the diagonal matrix whose (i,i)-element is the sum of A's i-th row, and construct the matrix $L = D^{-1/2}AD^{-1/2}$.[1]
3. Find $x_1, x_2, \ldots, x_k$, the k largest eigenvectors of L (chosen to be orthogonal to each other in the case of repeated eigenvalues), and form the matrix $X = [x_1 x_2 \ldots x_k] \in \mathbb{R}^{n \times k}$ by stacking the eigenvectors in columns.
4. Form the matrix Y from X by renormalizing each of X's rows to have unit length (i.e. $Y_{ij} = X_{ij}/(\sum_j X_{ij}^2)^{1/2}$).
5. Treating each row of Y as a point in $\mathbb{R}^k$, cluster them into k clusters via K-means or any other algorithm (that attempts to minimize distortion).
6. Finally, assign the original point s_i to cluster j if and only if row i of the matrix Y was assigned to cluster j.

a) Implement the function spectral_clustering:
`spectal_clustering(data, n_centers, sigma): Divide data into n_centers clusters, return the assignments.`

You can use your kmeans function from part 1 in the implementation.

b) In your main function, run kmeans and spectral clustering on the 2d data given. Write code to use matplotlib to visualize the clustering results of the two methods and save them as png files.

The ground truth K/n_centers for different data are visualized below: Aggregation (K=7), Bridge (K=2), Compound (K=6), Flame (K=2), Jain (K=2), Spiral (K=3) and TwoDiamond (K=2).



Sigma in spectral clustering is a hyper-parameter you need to find.

c) Answer the following questions:

1. Compare the results of the two clustering methods. Comment on why one method is performing better/worse than the other.
2. Suggest a way to find the optimal sigma.

Submit the following via eClass (as separate files, **DON'T** zip them):
1. lab7.py: Code
2. lab7.pdf: Answers to questions

Points will be deducted if the submitted filename and format doesn't match the requirement.

Grading:

- Code Quality: 5%
- Comments: 5%
- Part 1: 30%
    - distance(): 6%
    - kmeans(): 6%
    - distortion: 6%
    - 1.5% + 1.5% + 3% + 6%
- Part 2: 60%
    - spectal_clustering(): 40%
    - 10% + 10%