Machine Learning Homework 6

Kernel K-means and Spectral Clustering

Due Date 23:55 26th December.

I. **Homework Objective:** Use whatever your favorite language to code out kernel k-means, spectral clustering (both normalized cut and ratio cut). You should consider spatial similarity and color similarity upon the clustering.

II. **Data:** Two 100*100 images are provided, and each pixel in the image should be treated as a data point, which means there are 10000 data points in each image.

III. **Kernel:** For both kernel k-means and spectral clustering, please use the new kernel defined below to compute the Gram matrix.

$$k(x, x') = e^{-\gamma_s \|S(x)-S(x')\|^2} \times e^{-\gamma_c \|C(x)-C(x')\|^2}$$

This new defined kernel is basically multiplying two RBF kernels in order to consider spatial similarity and color similarity at the same time. $S(x)$ is the spatial information (i.e. the coordinate of the pixel) of data x, and $C(x)$ is the color information (i.e. the RGB values) of data x. Both γ_s and γ_c are hyper-parameters which you can tune in your own way.

IV. Requirements:

▸ Part1: You need to make videos or GIF images to show the clustering procedure (visualize the cluster assignments of data points in each iteration, colorize each cluster with different colors) of your kernel k-means and spectral clustering (both normalized cut and ratio cut) programs. (Hint : Numpy can help you to solve the eigenvalue problem.)

▸ Part2: In addition to cluster data into 2 clusters, try more clusters (e.g. 3 or 4) and show your results. (You also need to make videos or GIF images)

▸ Part3: For the initialization of k-means clustering used in kernel k-means, (e.g. k-means++) and spectral clustering (both normalized cut and ratio cut), try different ways and show corresponding results. (You also need to make videos or GIF images)

▸ Part4: For spectral clustering (both normalized cut and ratio cut), you can try to examine whether the data points within the same cluster do have the same coordinates in the eigenspace of graph Laplacian or not. You should plot the result and discuss it in the report.

V. **Report**:

- Submit a report in pdf format. The report should be written in English.
- Report format:
  - a. code with detailed explanations (40%)
    - Paste the screenshot of your functions with comments and explain your code. For example, explain the process to clustering and show different initialization methods, etc.
    - Note that if you don't explain your code clearly, you cannot get any points in section b and c either.
      - Part1 (15%)
      - Part2 (5%)
      - Part3 (10%)
      - Part4 (10%)
  - b. experiments settings and results (20%) & discussion (30%)
    - Show everything we asked you to show
      - Part1 (5%) & (5%)
      - Part2 (5%) & (5%)
      - Part3 (5%) & (10%)
      - Part4 (5%) & (10%)
  - c. observations and discussion (10%)
    - Anything you want to discuss, such as comparing the performance between different kernels or the execute time of different settings.

VI. **Turn in**:

1. Report (.pdf)
2. Source code
3. Videos or GIF images of clustering procedure

You should zip all above in one file and name it like ML_HW6_yourstudentID_name.zip, e.g. ML_HW6_0856XXX_王小明.zip.

P.S. If the zip file name has format error or the report is not in pdf format, a penalty will be imposed (-10). Please submit your homework before the deadline, late submission is not allowed.

Note that if you miss any one of the requirements (report, or source code), you cannot get any score!

♦ Packages allowed in this assignment:

You are only allowed to use numpy, scipy.spatial.distance, package for reading image and visualizing results. Official introductions can be found online.
Important: scikit-learn and SciPy is not allowed.