**ML Homework 6 Report**

1. Code with detail explanations
2. Kernel K-means

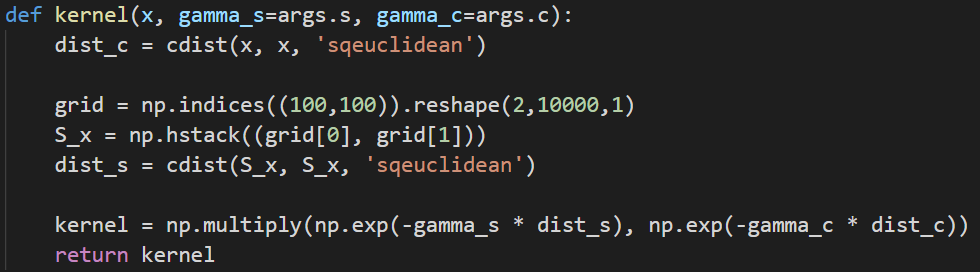
* Kernel

Use the new kernel defined below:



where S(x) is the spatial information of data x , and C(x) is the color information of data x. Both γs and γc are hyper-parameters.

According to the formula above, we define the kernel:



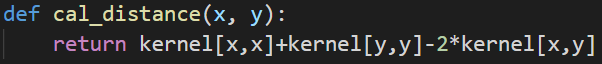
Which sqeuclidean is the Squared Euclidean distance.

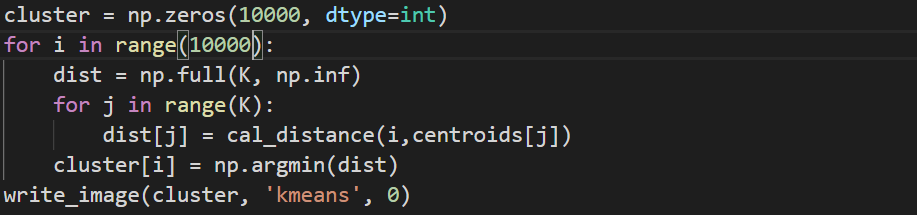
* Initial clustering

Use the distance defined below:



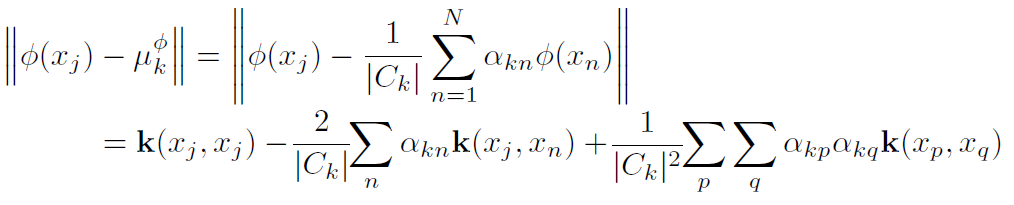
Assign each data point to the closest center point.



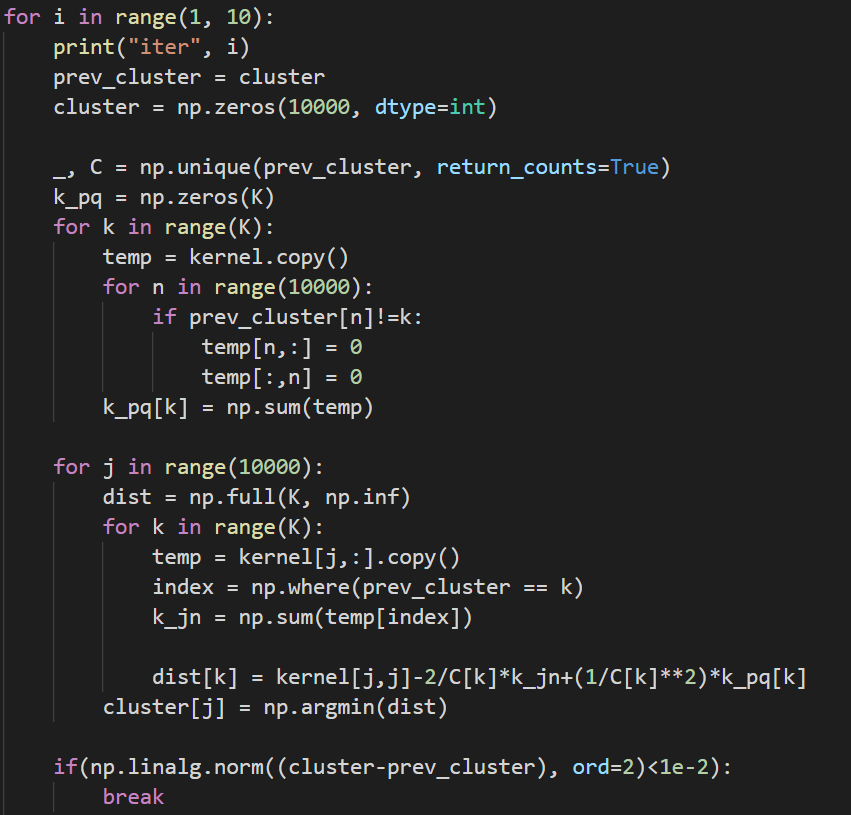


* Iterative Cluster Procedure

Use the function defined below as distance:



Assign to different cluster according to the minimum distance. Recursively until the clustering result converges.

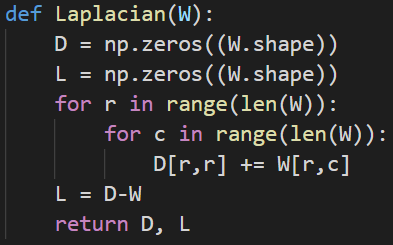


1. Spectral Clustering

* Compute Laplacian

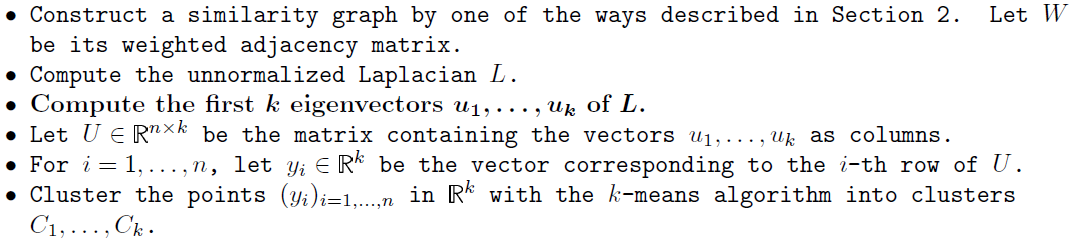
First calculate the D(degree matrix of W), then calculate the Laplacian matrix according to the following formula:

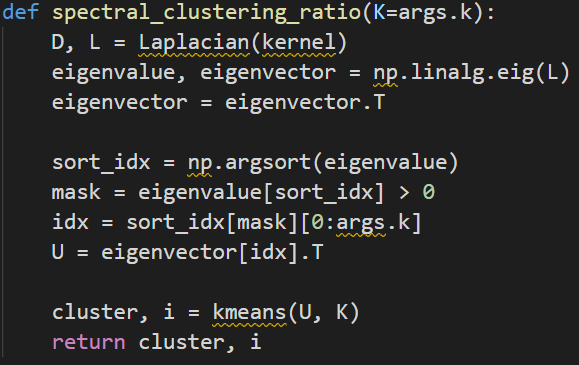




* Ratio cut

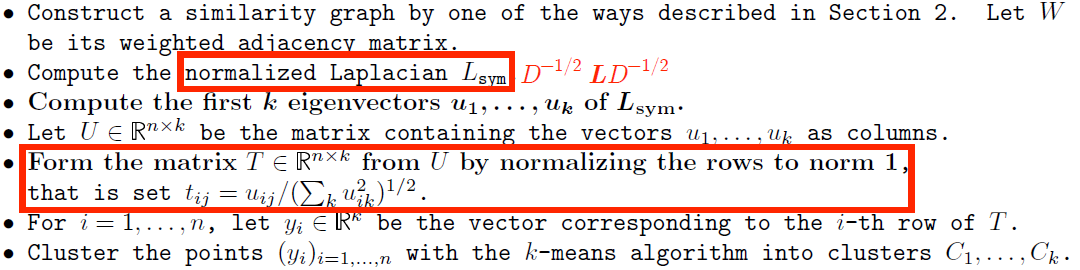
Follow the algorithm below:

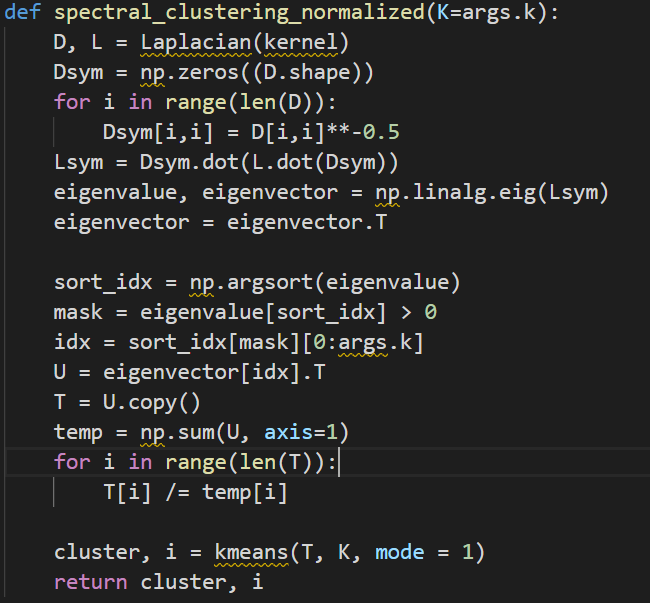




* Normalized cut

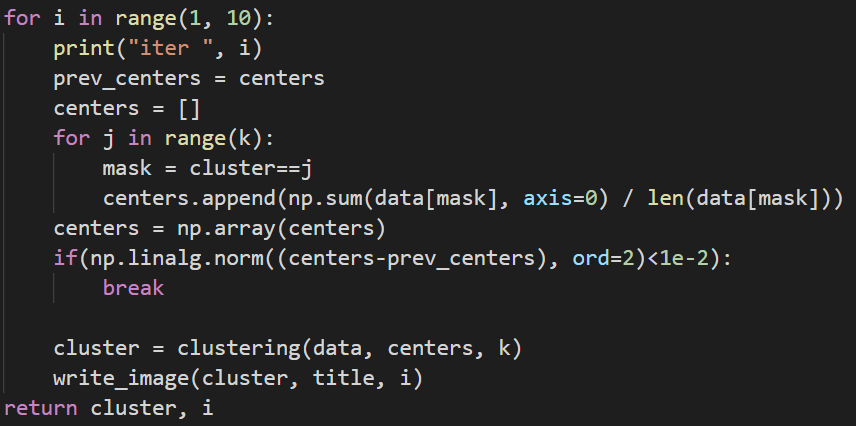
Follow the algorithm below:





* K-means

New centers is equal to average point of each clusters. Assign to different cluster according to the minimum distance. Recursively until the clustering result converges.



1. Part 2

Try more clusters with different value of argument parser k.



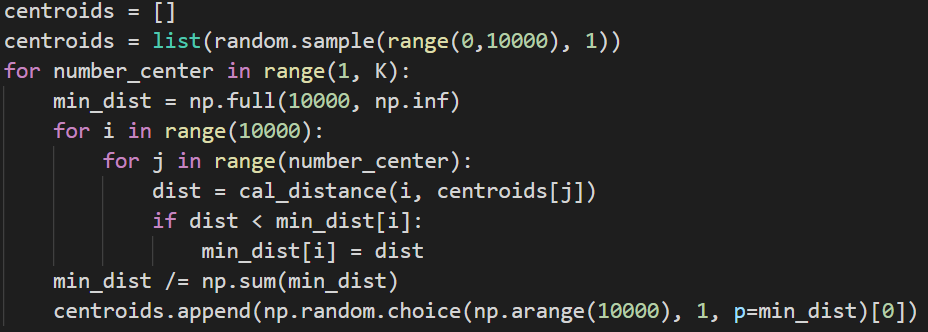
1. Part 3: try different initial way

The simplest way: random pick central points.

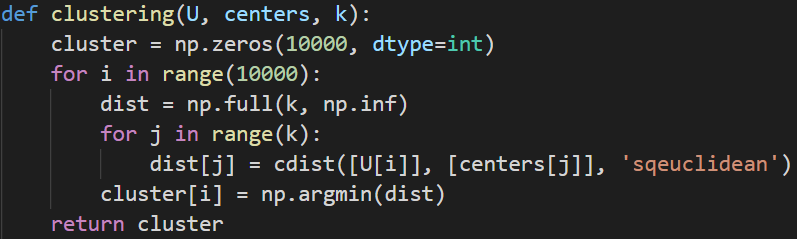


I also use the K-means++ algorithm to select the initial center point of the cluster, which improves that the random selection of the initial center point may cause the clustering results to be very different from the actual distribution of the data.

The basic idea of the K-means++ algorithm is that the mutual distance between the initial cluster centers should be adjusted far.

The specific initial selection process is as follows: Randomly select a point from the input data point as the first center; for each point x in the data set, calculate the distance D(x) between it and the nearest center, and take D(x) as the weight to take the next center . Repeat until k centers are obtained and stop.

1. Part 4: plot eigenspace cluster



1. Experiments settings and results
2. Part 1 (K=3)

|  |  |  |
| --- | --- | --- |
| Kernel-k-means | Ratio cut | Normalized cut |
|  |  |  |

1. Part 2

|  |  |  |  |
| --- | --- | --- | --- |
| K | 2 | 3 | 4 |
| Kernel-k-means |  |  |  |
| Ratio cut |  |  |  |
| Normalized cut |  |  |  |

1. Part 3

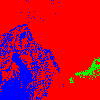
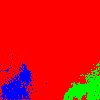
|  |  |  |  |
| --- | --- | --- | --- |
| k-means++ | Kernel-k-means | Ratio cut | Normalized cut |
| random |  |  |  |
| Kmeans++ |  |  |  |

1. Part 4

|  |  |  |
| --- | --- | --- |
|  | Image1 | Image2 |
| Ratio cut |  |  |
| Normalized cut |  |  |

1. Observations and Discussions

* From the results of grouping, we can see the junction of land and sea.
* From the results of part1, Ratio has the best clustering.
* Although k-means will converge with the number of iterations, the result may be worse than before. As can be seen from the two pictures below, although the results of grouping are getting more and more clustered, the outlines disappear.

Iteration = 0 , Iteration = 5 

* Generally speaking, the number of iterations of Spectral Clustering will be much less than that of Kernel K-means. But because of the large amount of calculation of eigenvector, it will take a lot more time than k-means.
* The clustering results can get better results when k = 3 or 4 on image1. When k>4, there may be only one or two dots with a few colors on the graph.
* The appropriate size of the k value depends on different pictures. When the picture contains more color blocks, the k value cannot be too small.
* Different initial clustering methods will greatly affect the initial results, but after iteration, the gap between random and k-means++ will narrow.
* Compared with k-means++, random requires more iterations to converge.
* Data points within the same cluster have the same coordinates in the eigenspace of graph Laplacian