**K-Means for Clustering**

***1. The main idea in K-Means***

The idea for using K-Means is intuitive. It can be divided into four steps:

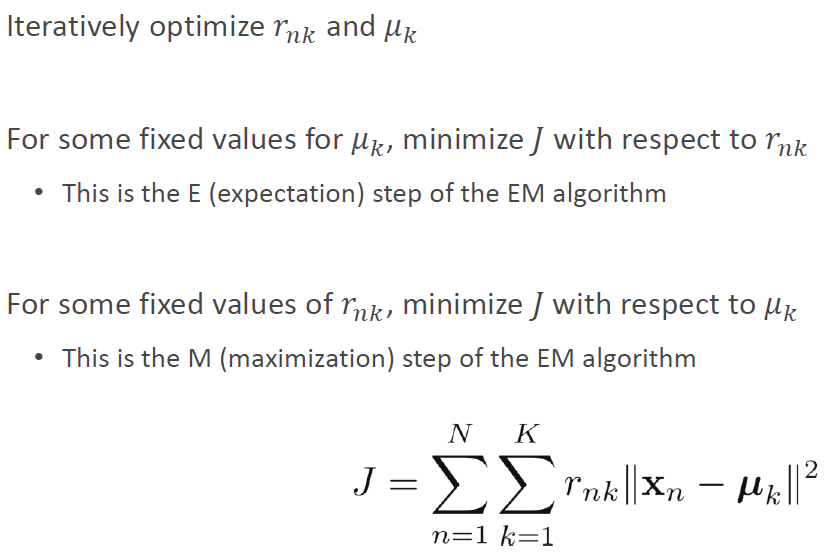
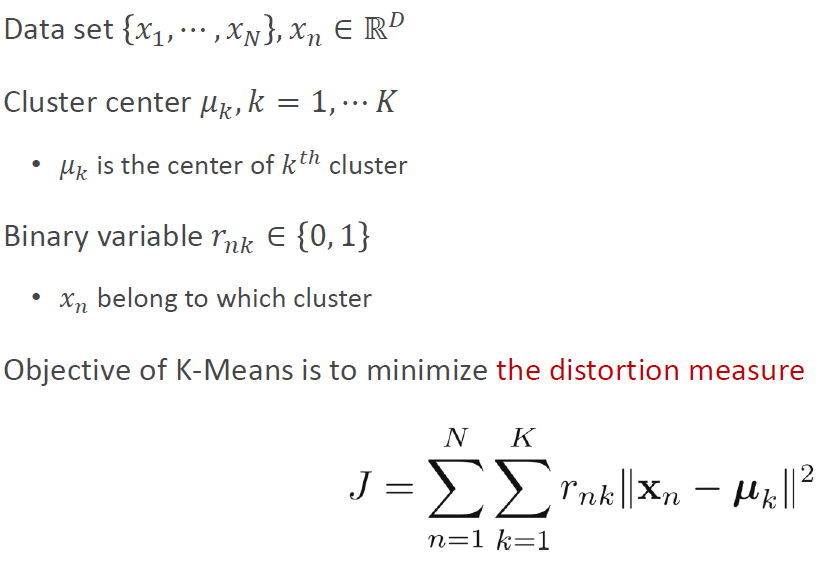
(1) Randomly select K center points.

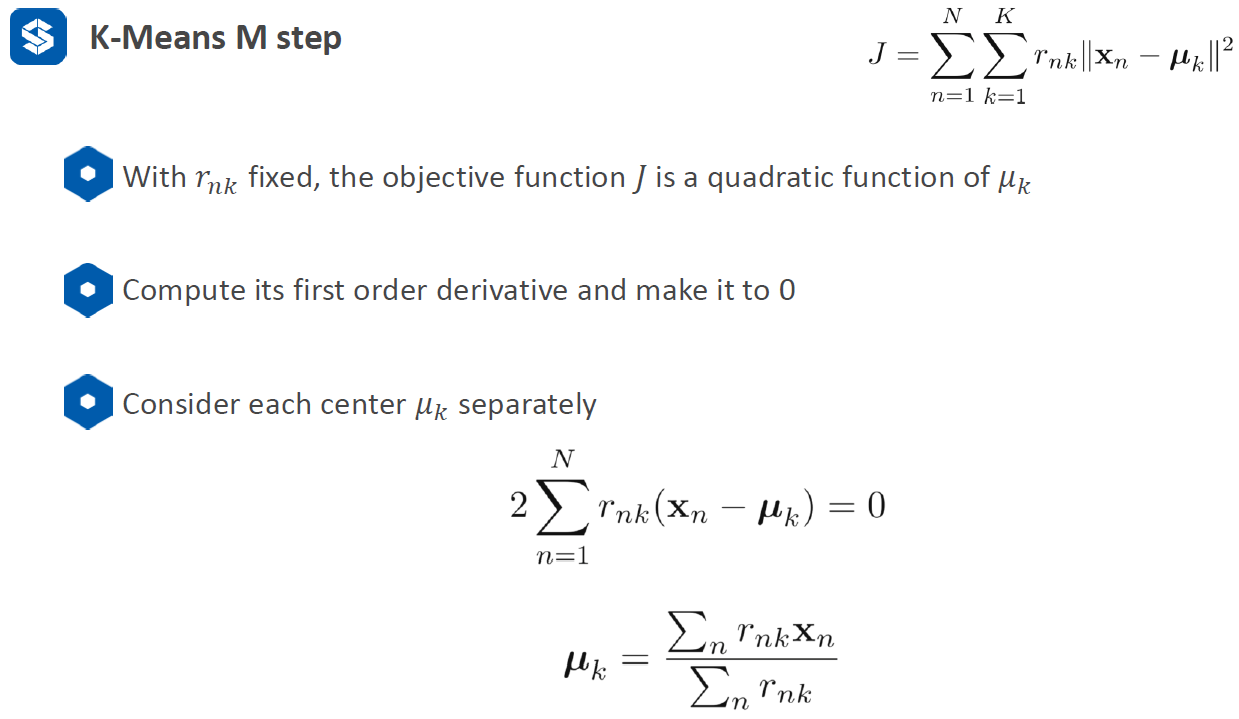
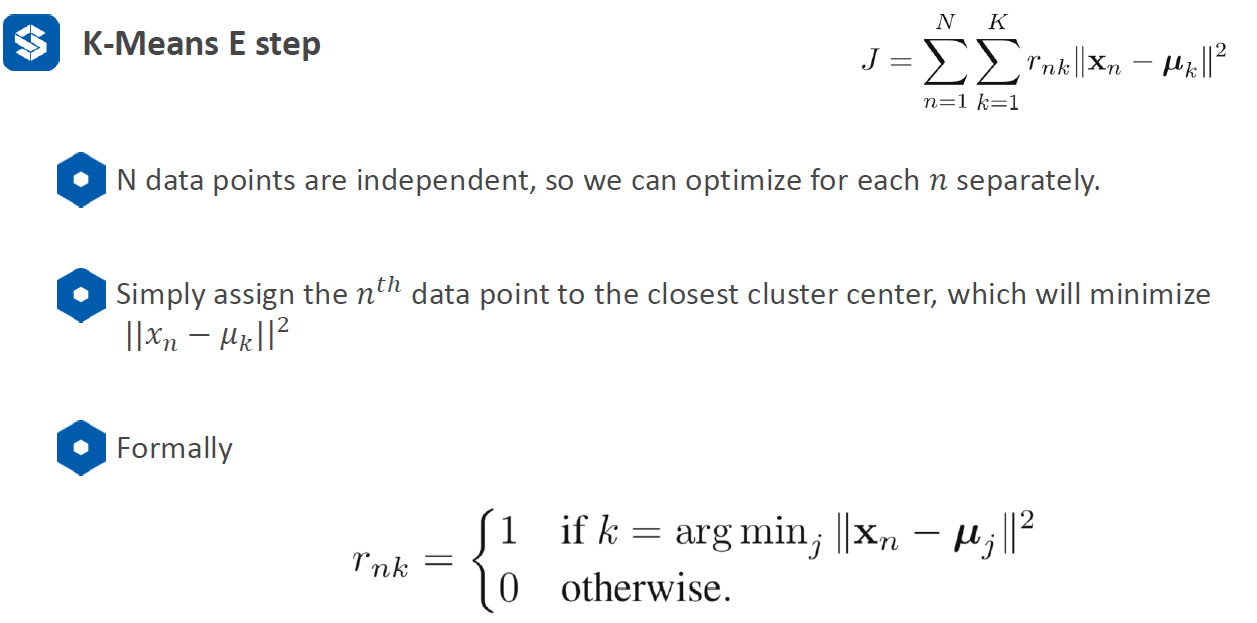
(2) Each data point is assigned to one of the K centers.

(3) Re-compute the K centers by the mean of each group.

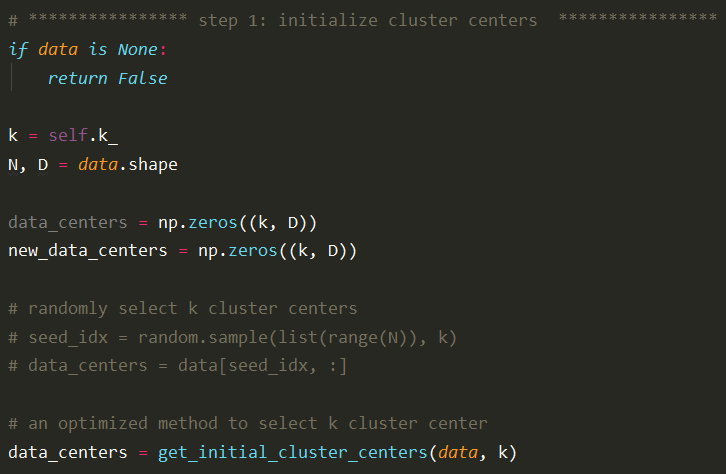
(4) Iterate step 2&3.

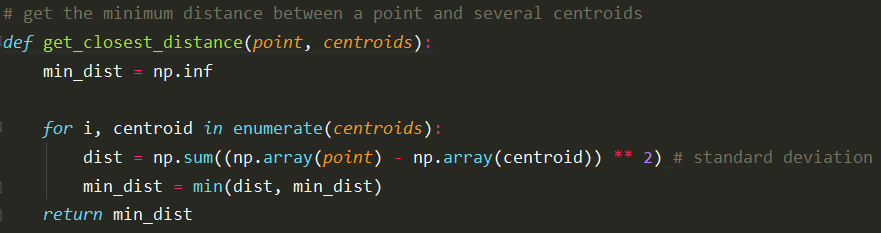
***2. K-Means Definition***



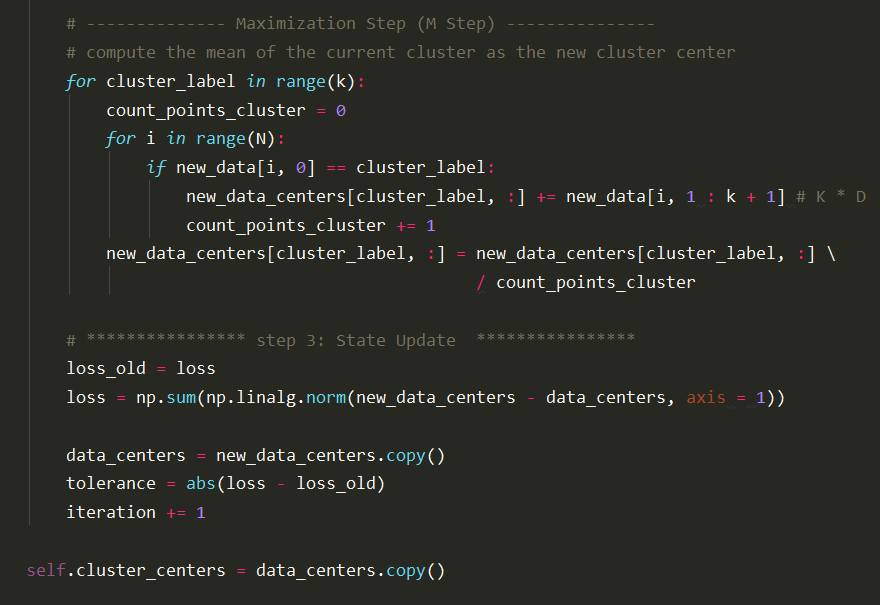
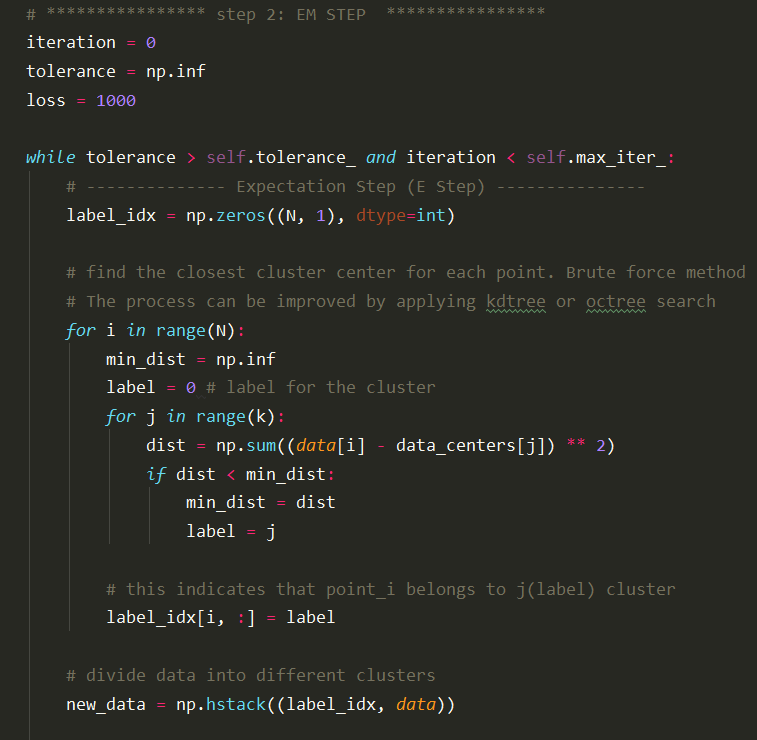


***3. Initialization***





***4. EM Steps***



***5. Predict***

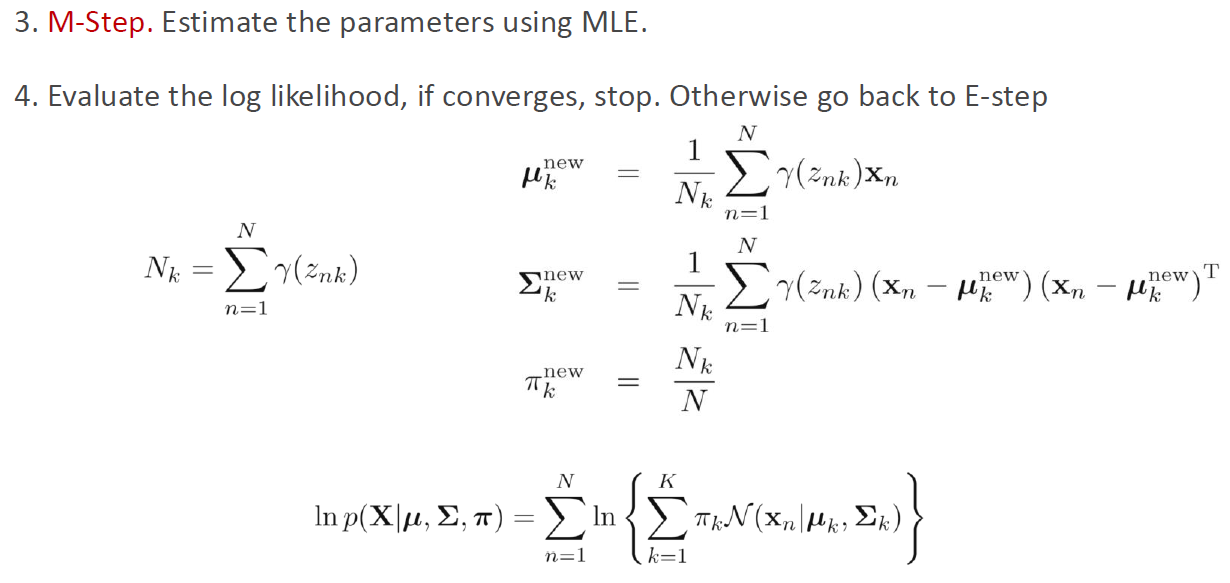
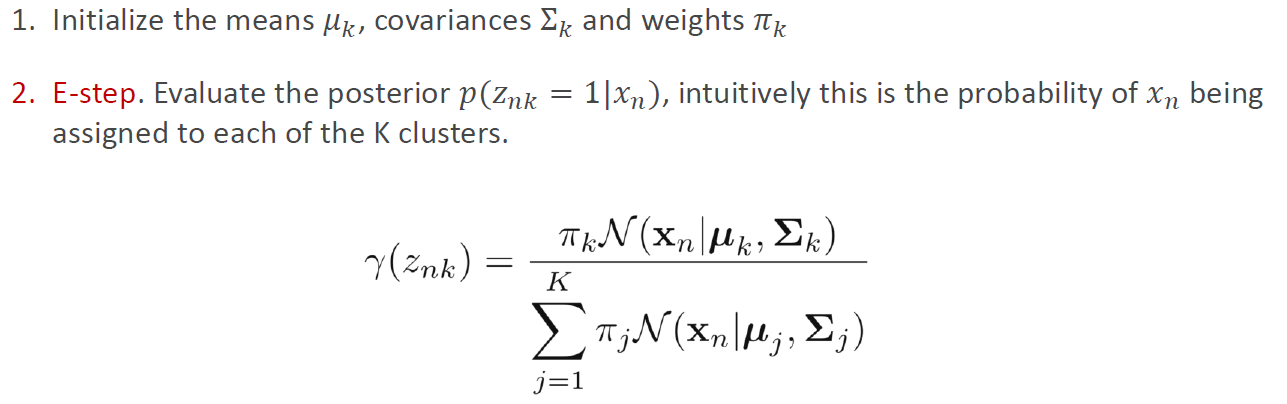
For each point, find out it nearest neighbor and corresponding cluster label, then output all the labels, which are the classified results.



**Gaussian Mixture Model (GMM) for Clustering**

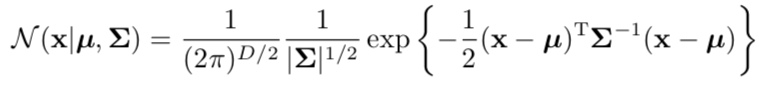
***1. The main purpose in GMM***

Compared to K-means, a cluster in GMM can be represented by a Gaussian Distribution. GMM tells the probability of a data point belonging to each cluster. How does this work? Suppose we have the posterior probability of each data point, which has a dimension of N x K (N is the number of points, K is the number of clusters). This means that for each point, we will have K probability values and each value indicates the probability to a specific cluster. For each single point, we can compare all K values and get the one with highest probability. The label corresponding to this value is the cluster that the data point belongs to. After we consider all the data points, we can get the corresponding cluster labels for these points, and hence the data points are classified as K clusters. Therefore, ***the most important part in GMM is to compute the posterior probability of the data points***. The steps can be divided as:

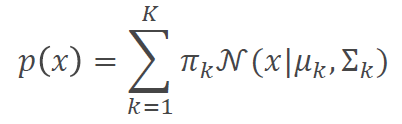


***2. Parameters Update***

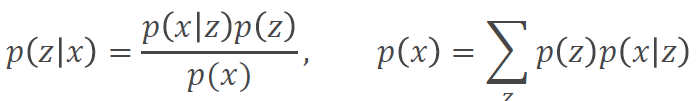
For multivariate Gaussian Distribution with D-dimensional vector, it can be represented as



A Gaussian mixture distribution can be written as a linear combination of single Gaussians with weights:

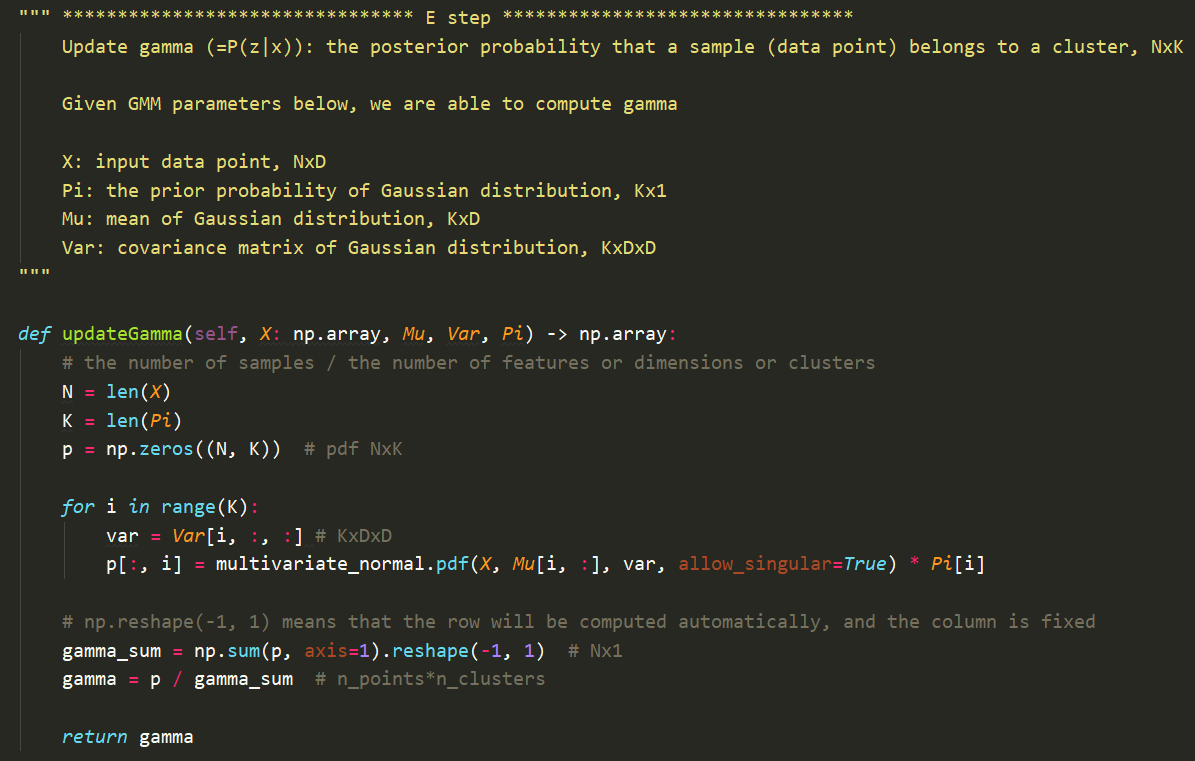
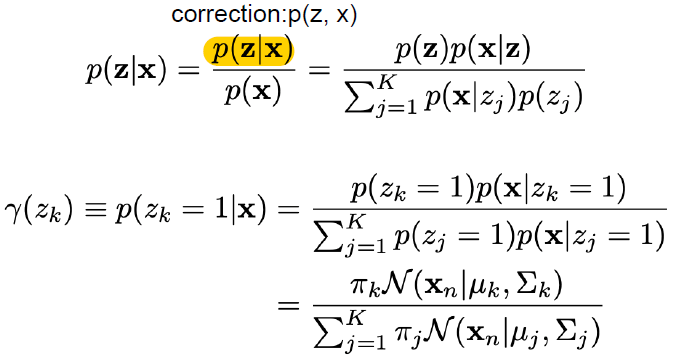
. Note: K is the number of data points, not the number of clusters.

The conditional probability P(z|x) is the clustering “label probability” we want, here is

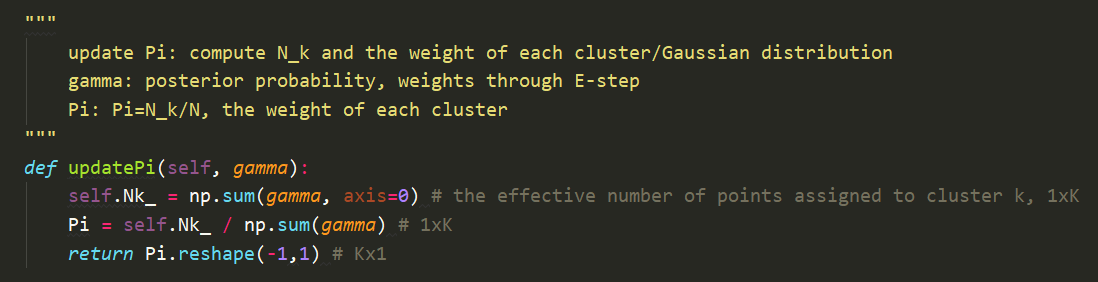
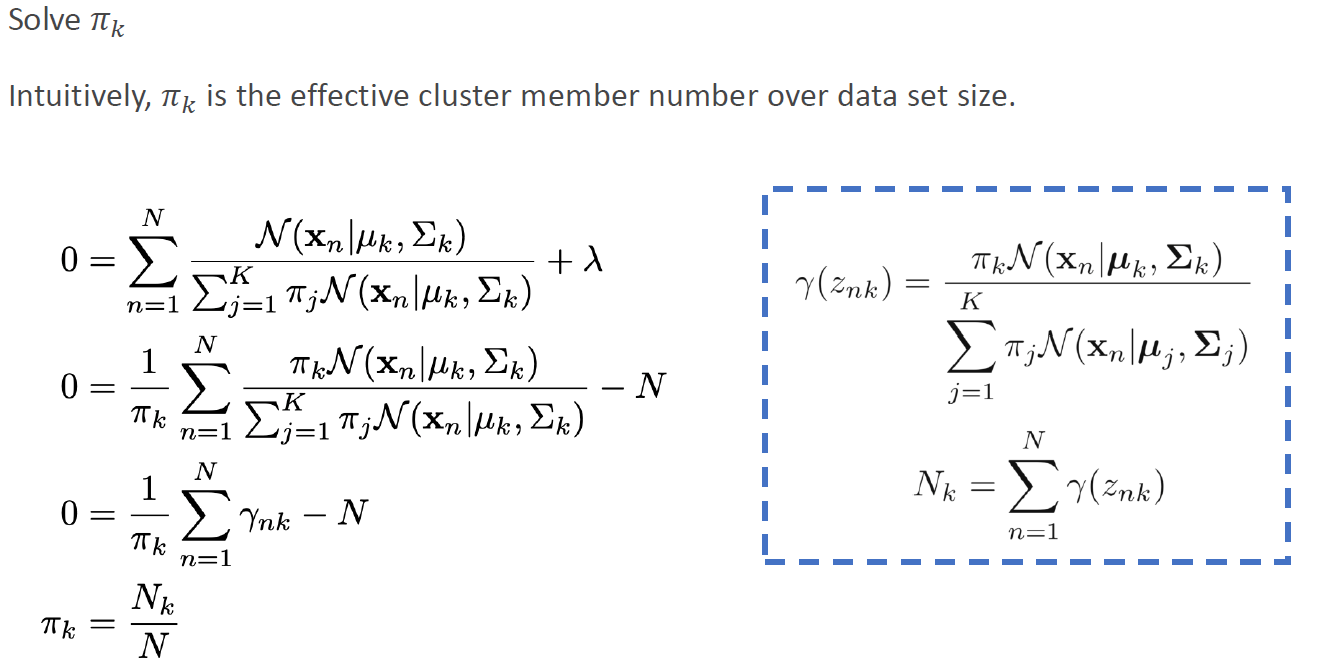


Therefore, the main purpose is to compute . There are other important parameters in GMM, such as , which will also be computed during this process.

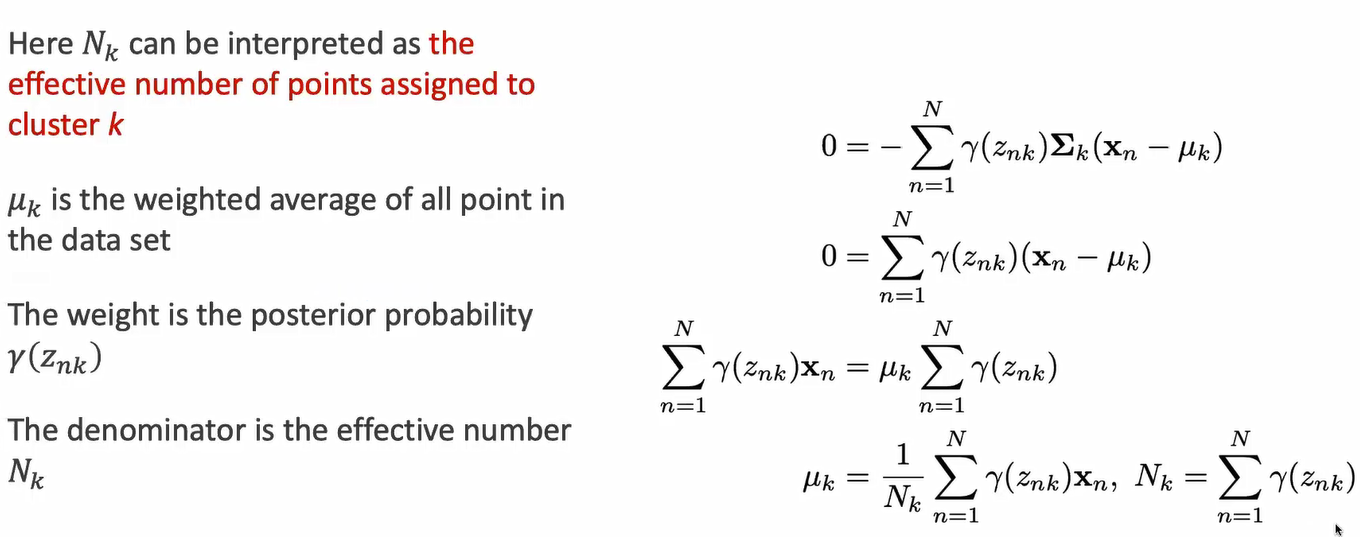
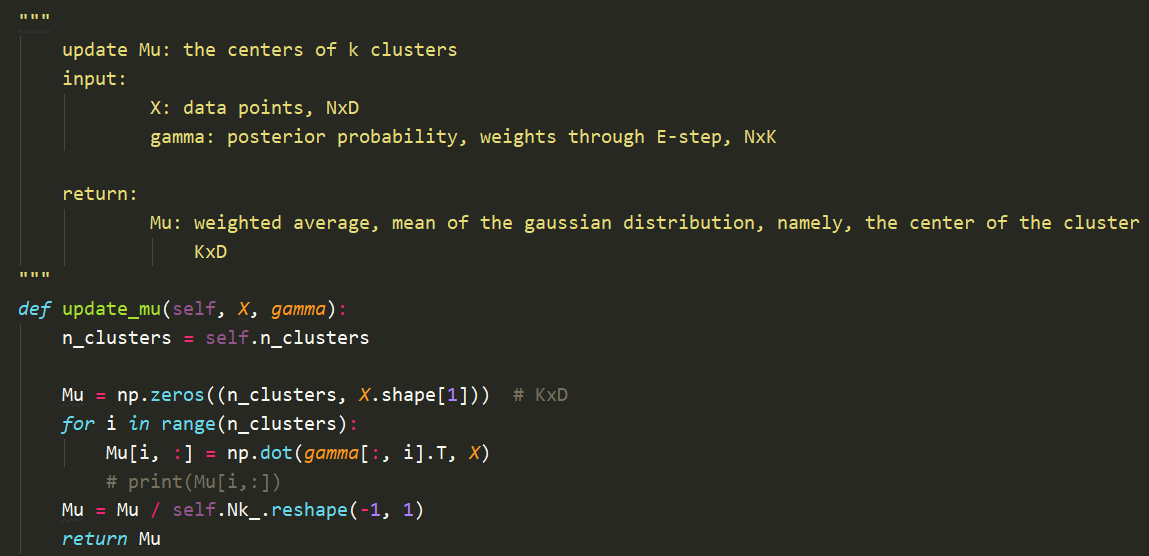
1. Compute



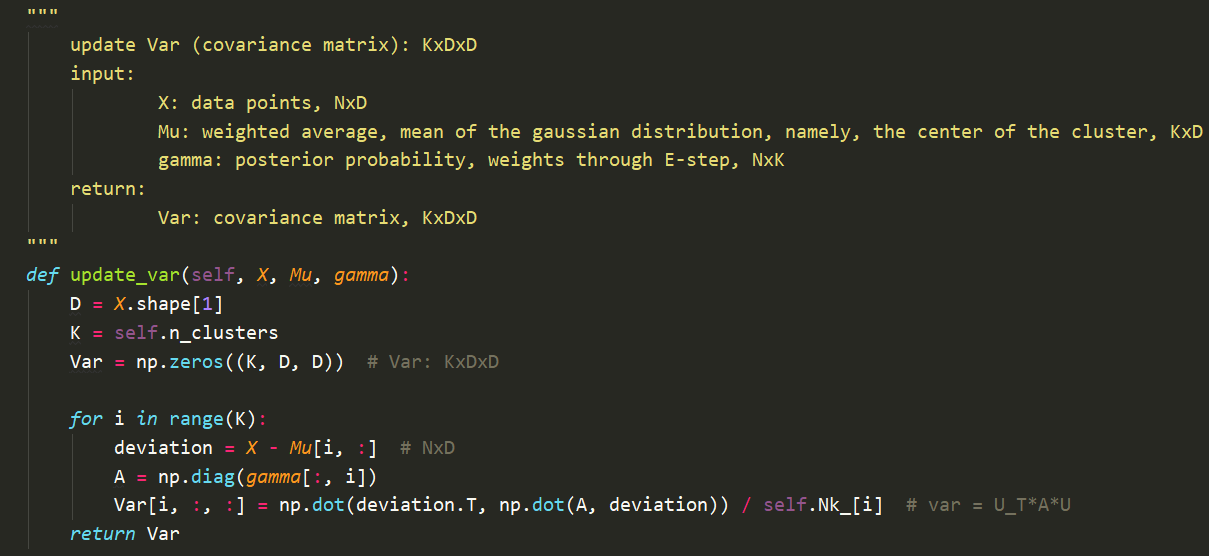
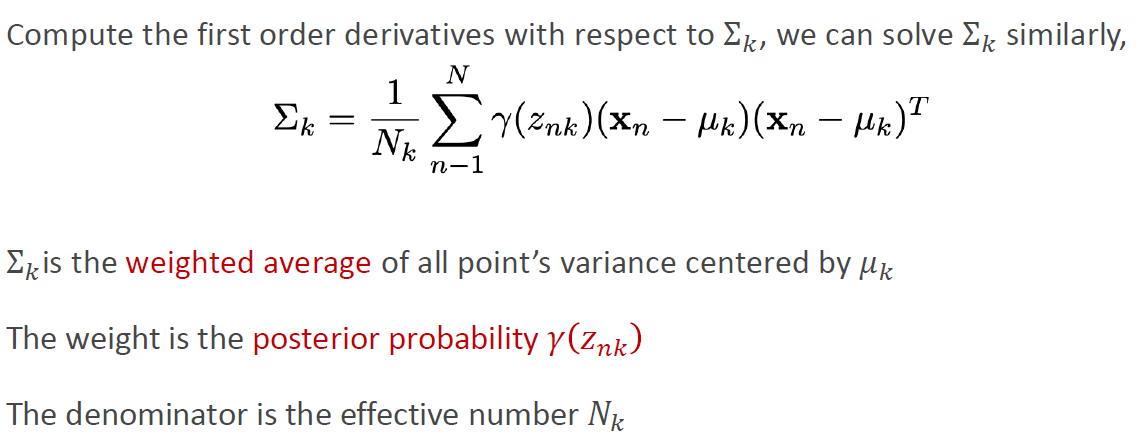
1. Compute



1. Compute

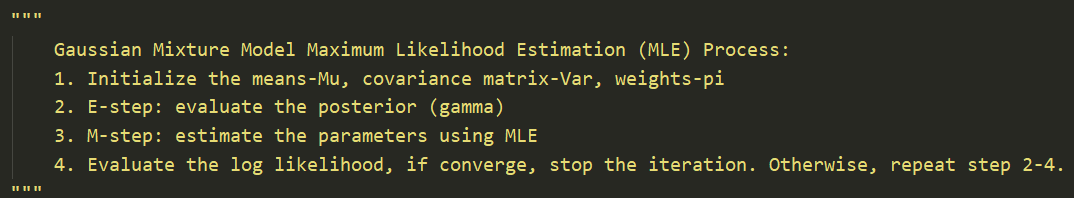
 

1. Compute



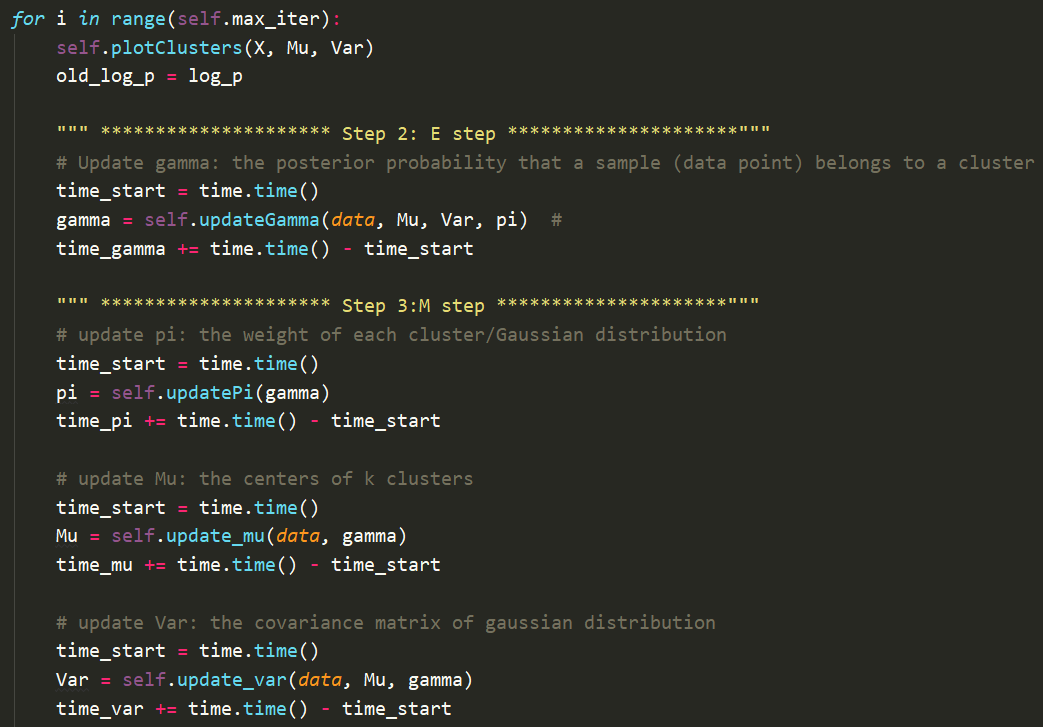
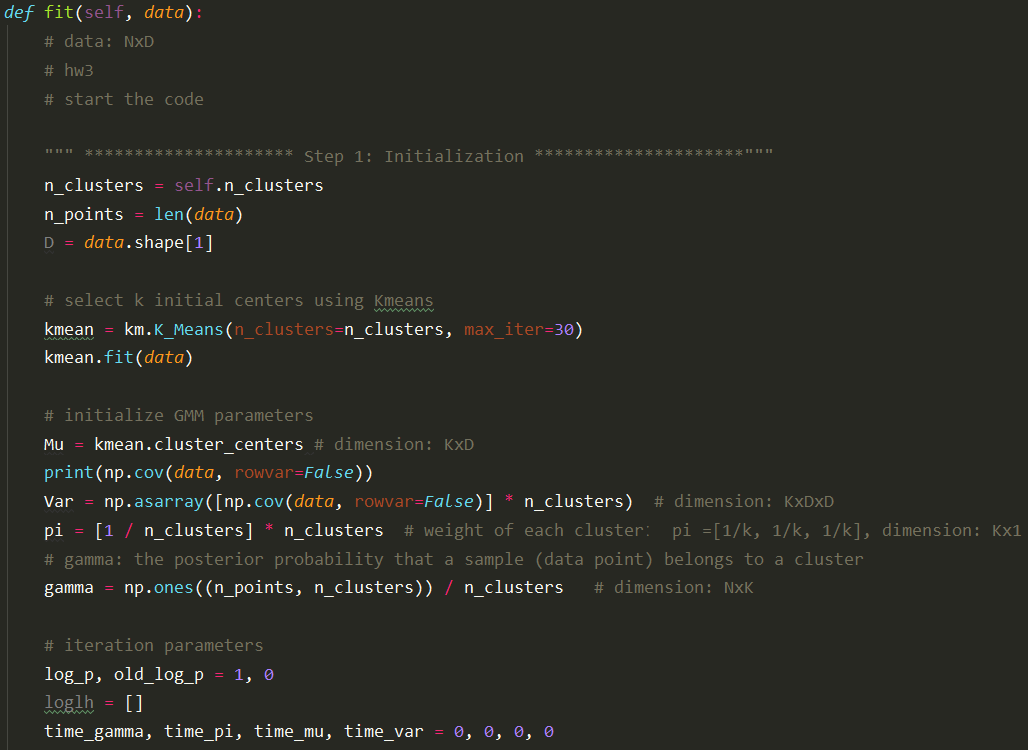
***3. GMM Process***

The process in GMM can be divided into four steps as follows.

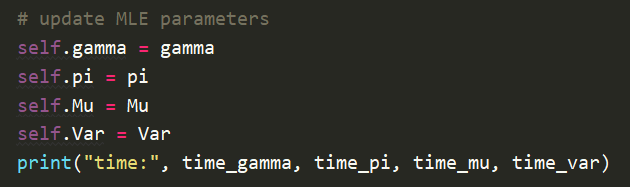
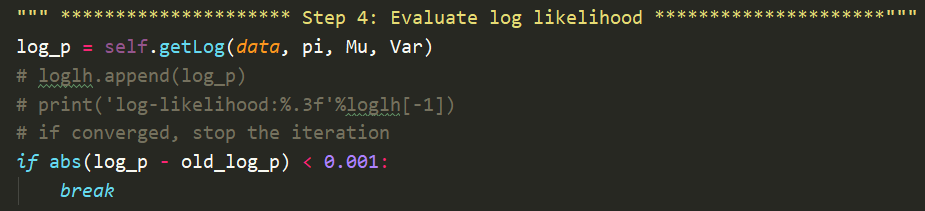


Step 1: Initialization. Note: K-means is used here to get the initial cluster centers.

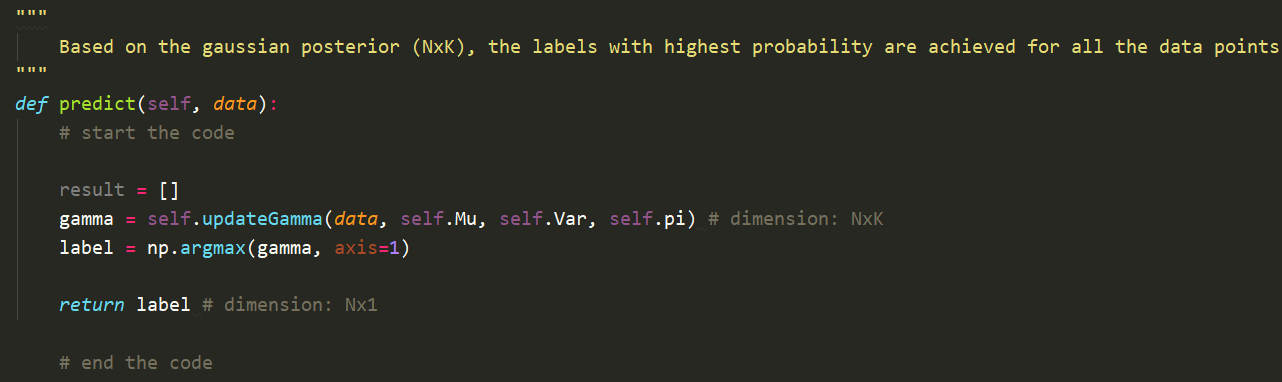
Step 2-3: EM steps. Iteratively update the four parameters and record the time used for each function.



Step 4: Evaluate log likelihood. If the difference between current and previous log likelihood is smaller than 0.001, then we can consider the iteration as converged, and stop the update.



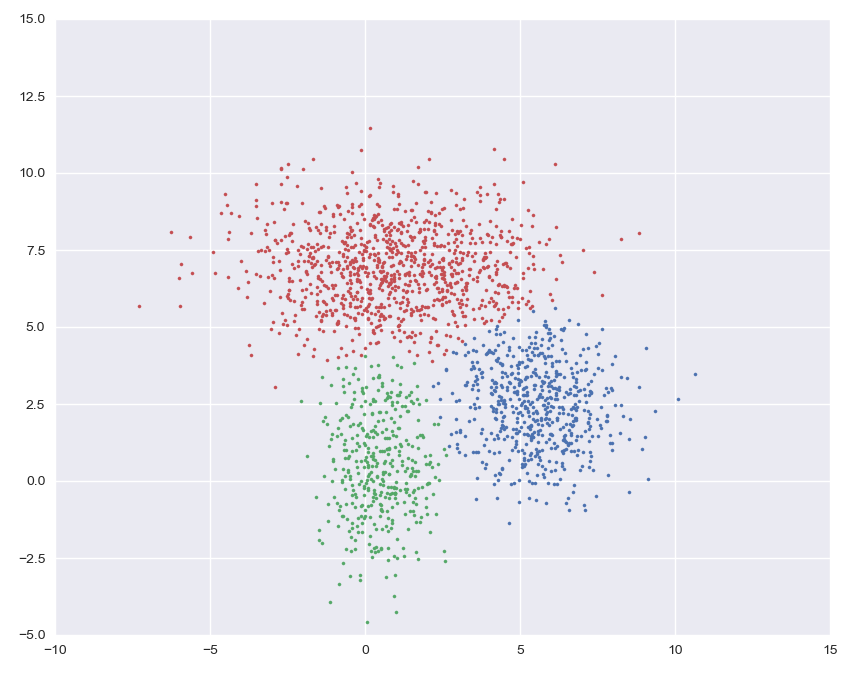
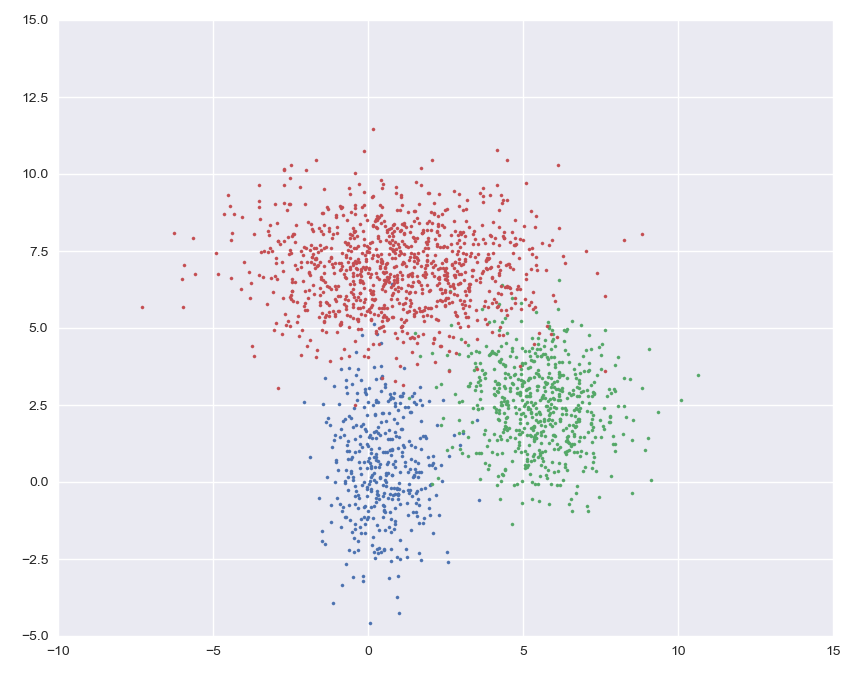
After we fit GMM, we obtain the posterior probability of . However, how can we get the final classification results based on the posterior probability? Here it is.



The posterior is used to output the label with highest probability for each data points, and finally output the labels for all the points. This is the classification result we want for clustering.

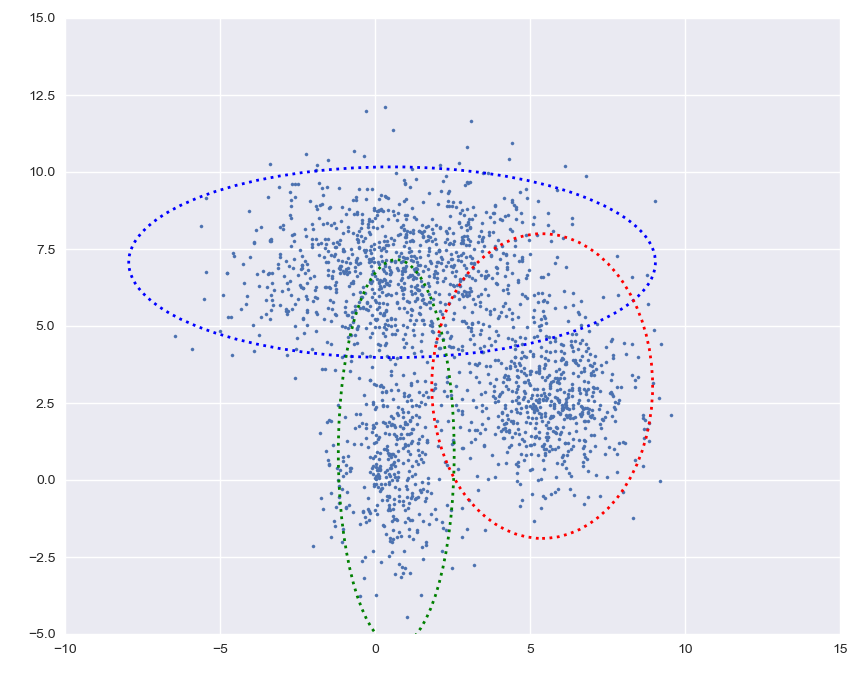
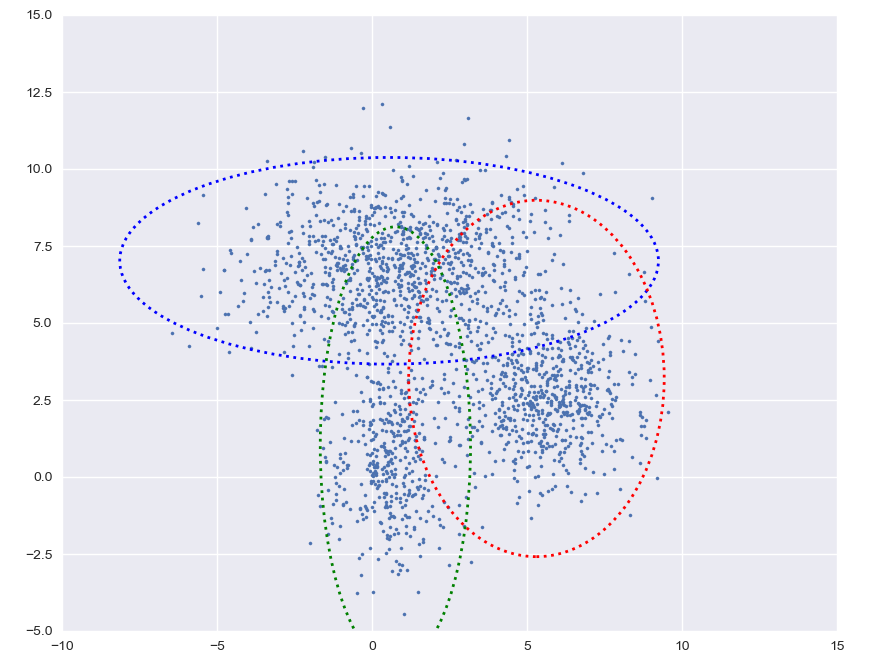
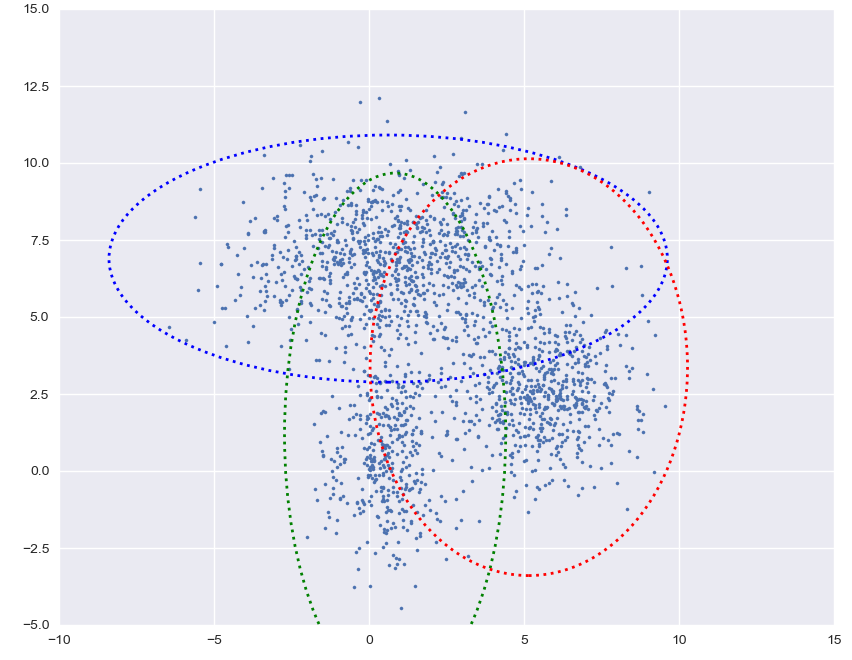
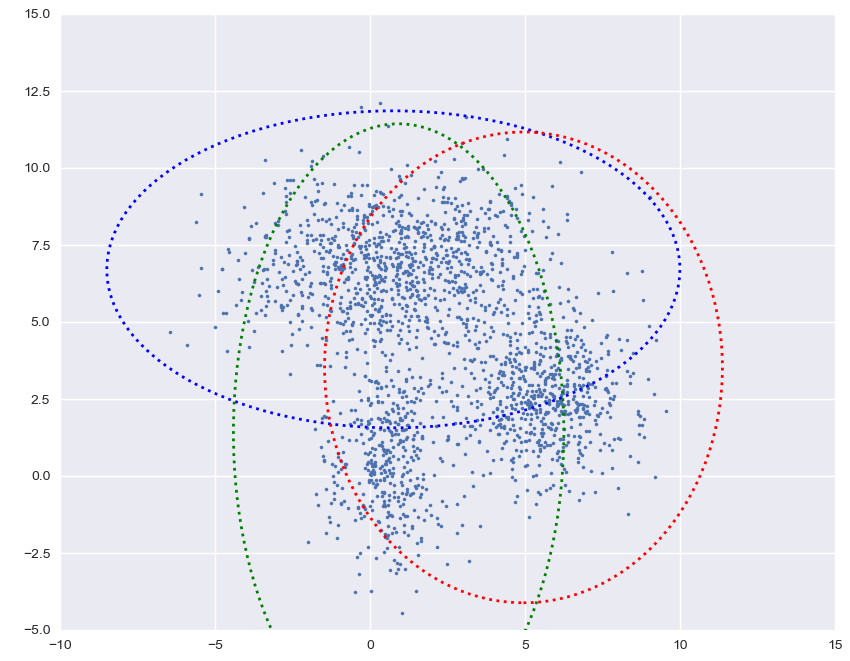
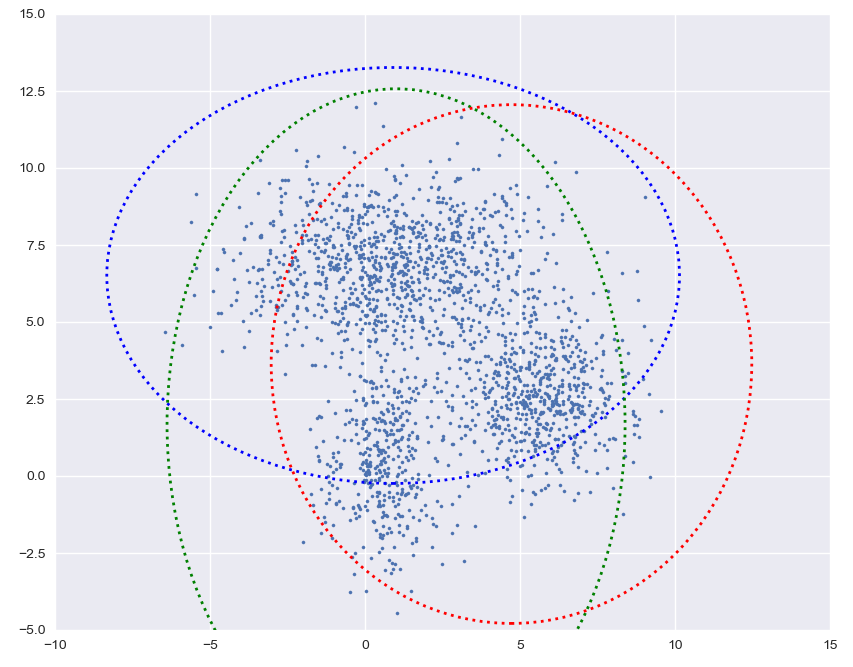
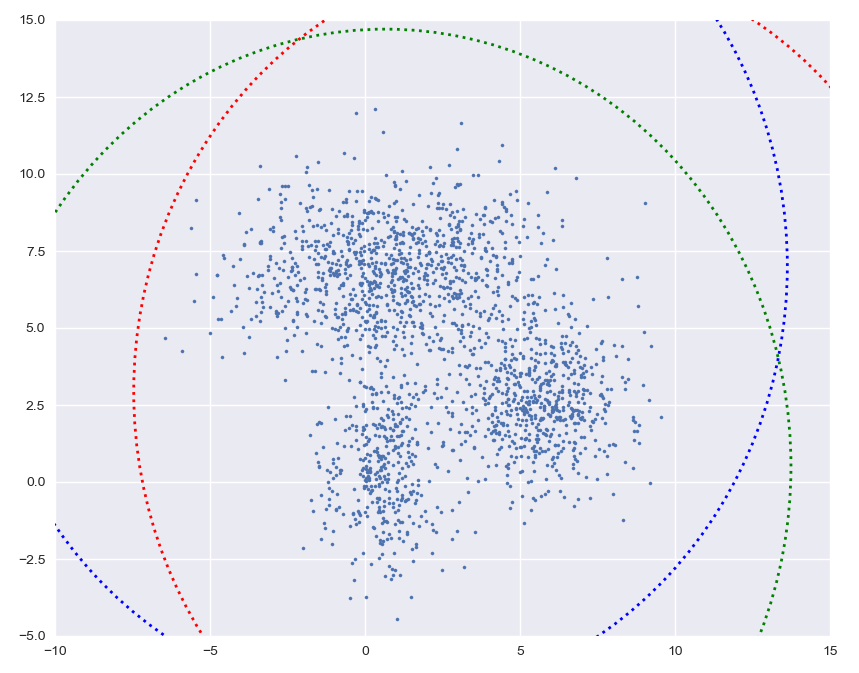
***4. Simple Test for GMM***

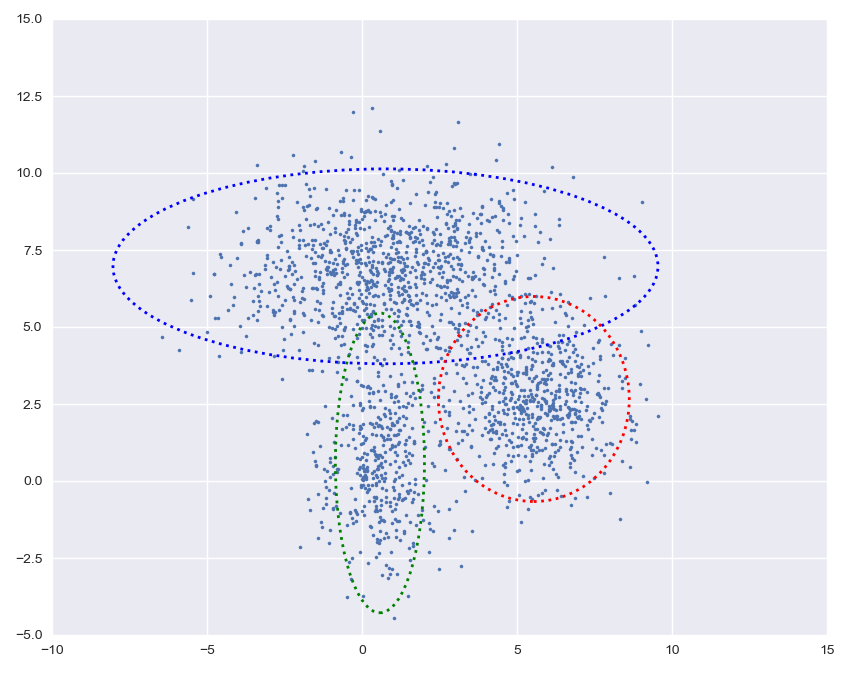
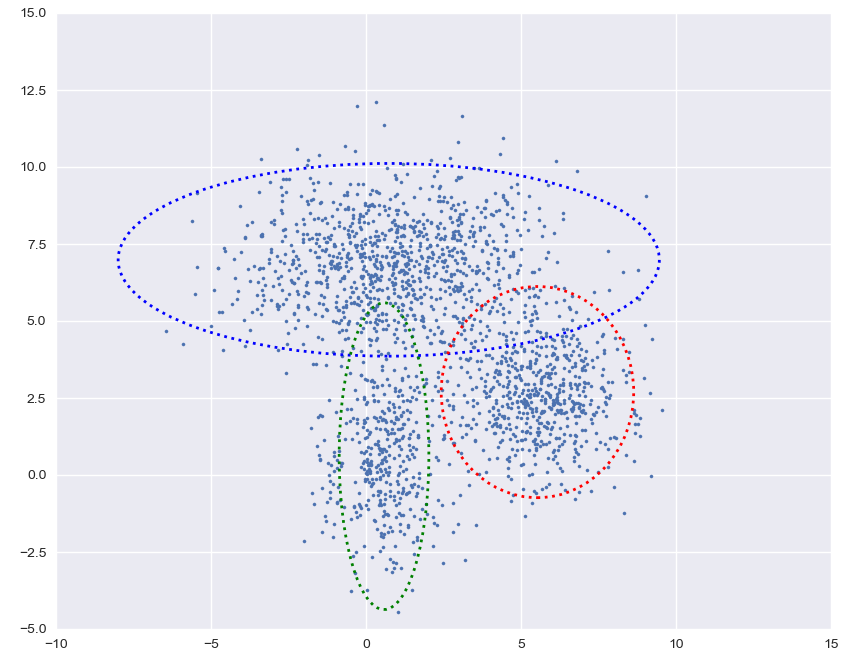
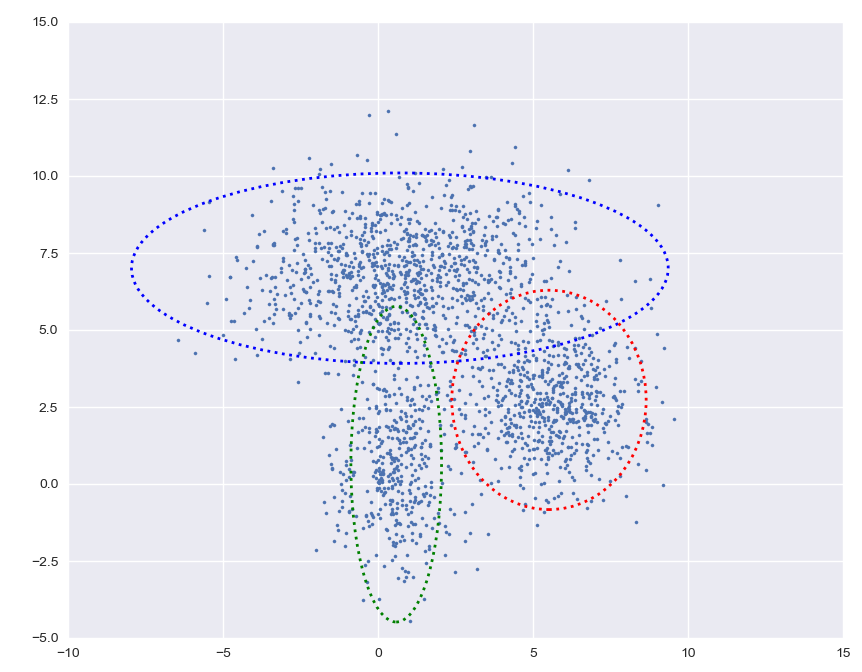
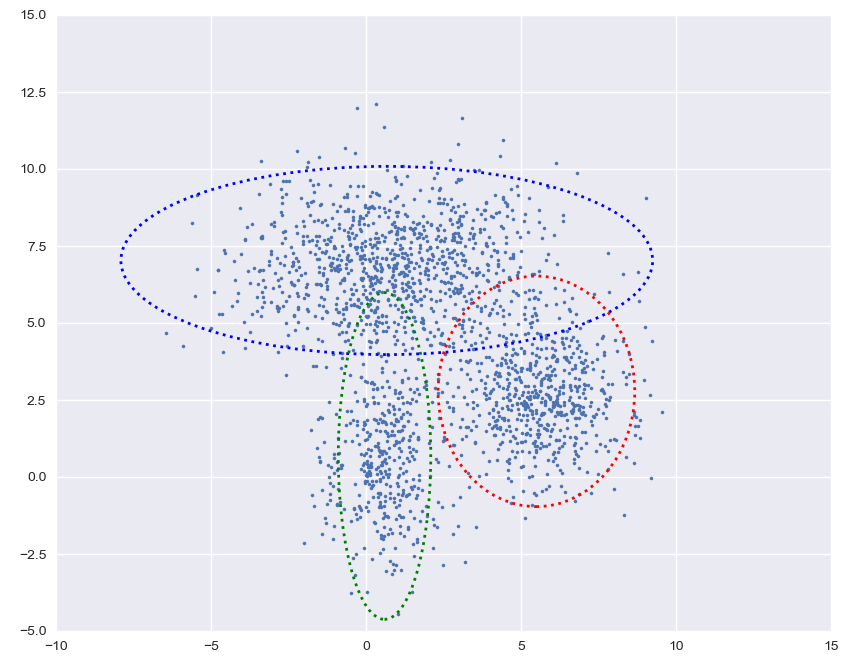
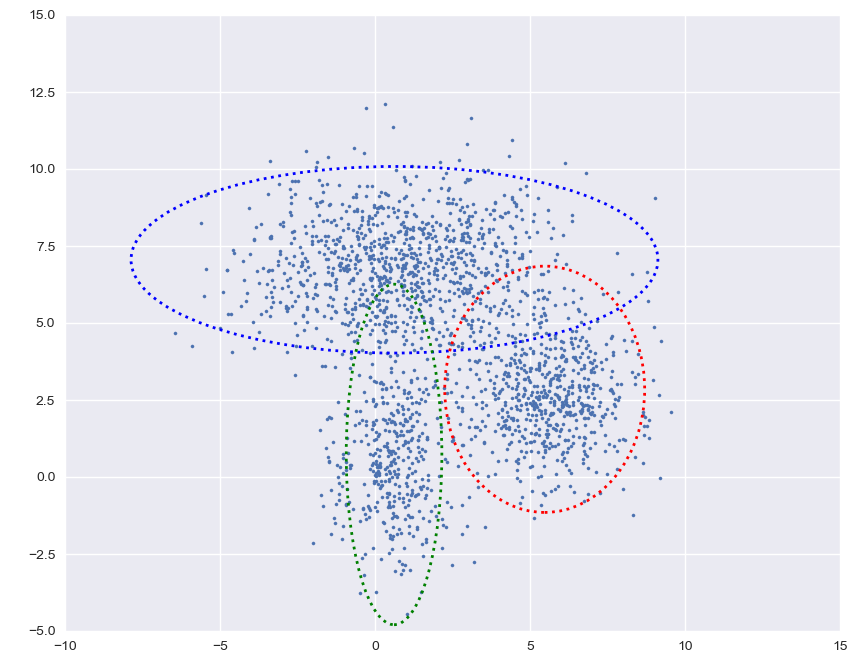
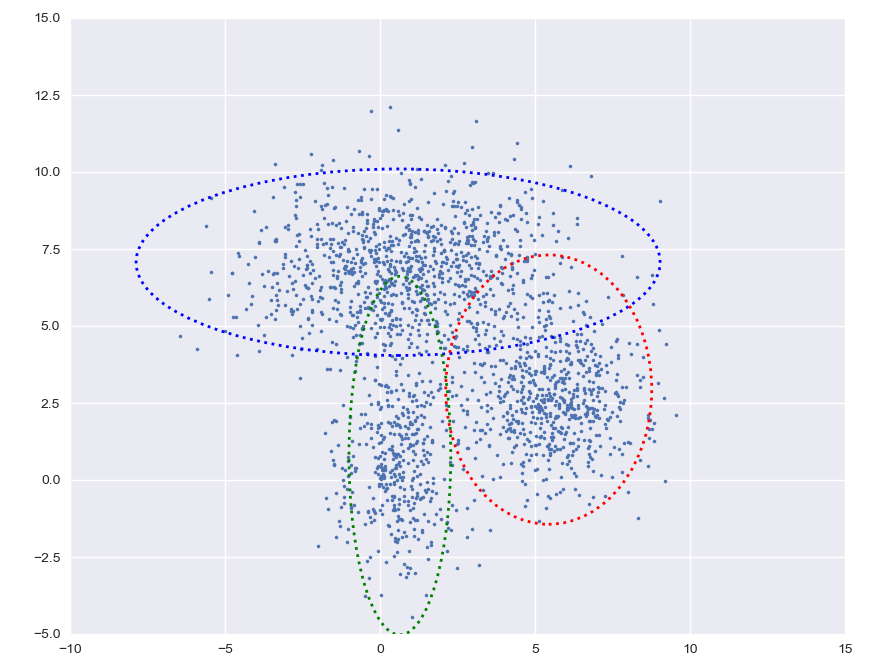
Here we use a simple example to demonstrate how GMM works for clustering. We use 2000 points to generate three clusters with 2D multivariate Gaussian distribution, which are 400, 1000, 600, respectively. Below shows the raw data points and the data points after GMM clustering. The result looks good.



Raw data Data after GMM

Below shows the details how GMM work during the clustering process.



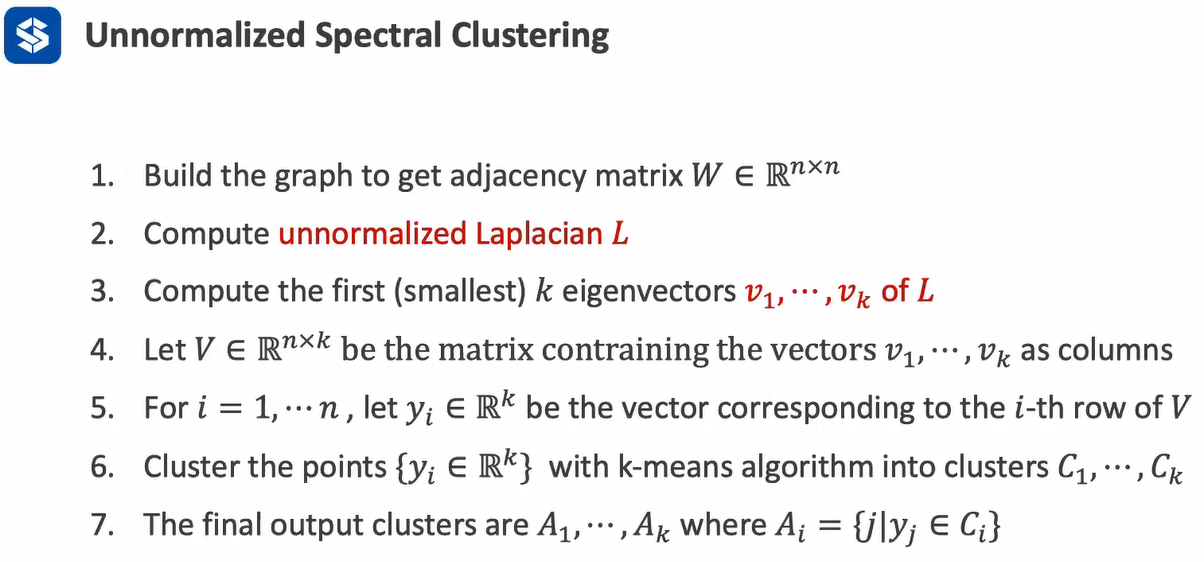


For the last few steps, the clustering becomes stable, which means the iteration tends to converge.

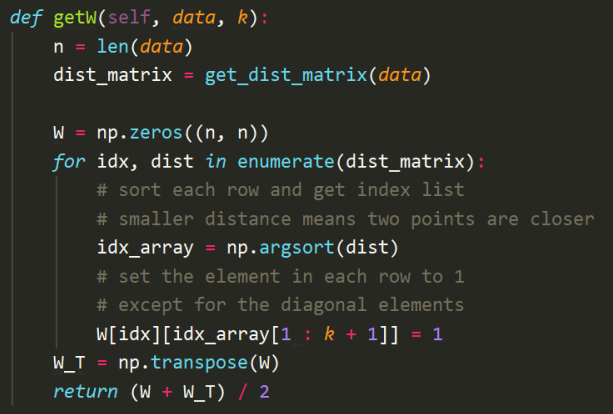
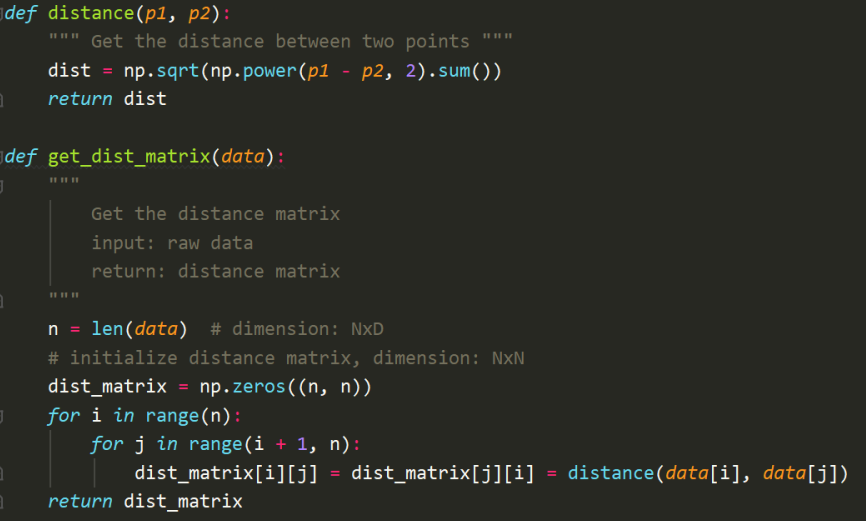
**Spectral Clustering**

GMM and K-Means are based on Euclidean distance, but they do not perform well for other metrics, i.e., connectivity. Thus, we use spectral clustering to deal with connectivity problems because it focuses on the connectivity between points in a graph (similar to connected blocks in a graph).

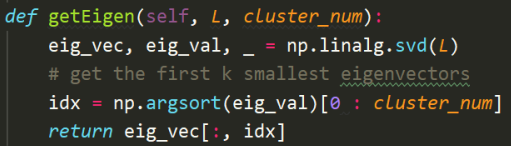
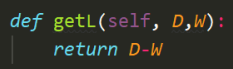
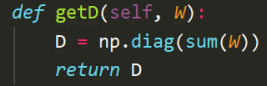
1. ***Procedure for Spectral Clustering***



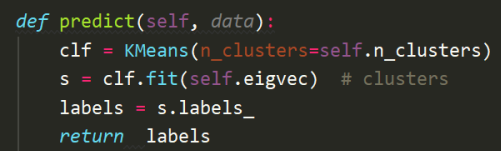
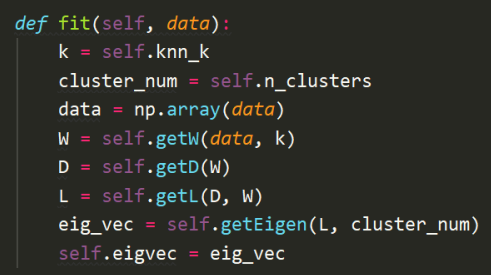
1. ***Compute Adjacent Matrix of Graph***



1. ***Compute Unnormalized Laplace Matrix and Corresponding Eigenvectors***



1. **Fit the model and predict the labels**



Results Comparison by using self-developed clustering algorithms and default algorithm in Scikit-learn

