CS584 Assignment 5 Report

Jingyu Zhu A20311779

1. Problem Statement

In this assignment, I implement the K-means clustering algorithm and EM algorithm with mixture Gaussian, which is also for the clustering. And then I measure the performances for both of the algorithms according to the distance. The datasets I used are the Iris dataset and the wine dataset.

2. Proposed solution

K-means algorithm

K-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype f the cluster.

Expectation Maximization algorithm

An expectation maximization algorithm is an iterative method for finding maximum likelihood or maximum a posteriori estimates of parameters in statistical models, where the model depends on unobserved latent variables. The EM iteration alternates between performing an expectation (E) step, which creates a function for the expectation of the log-likelihood evaluated using the current estimate for the parameters, and a maximization (M) step, which computes parameters maximizing the expected log-likelihood found on the E step. These parameter-estimates are then used to determine the distribution of the latent variables in the next E step.

3. Implementation Details

K-means algorithm

A typical implementation of K-means clustering is as following.

Given a set of observations $(x_1, x_2, \dots, x_n)$, where each observation is a d-dimensional real vector, k-means clustering aims to partition the n observations into k ($\leq$ n) sets $S = \{S_1, S_2, \dots, S_k\}$ so as to minimize the within-cluster sum of squares (WCSS). In other words, its objective is to find:

$$argmin_S \sum_{i=1}^{k} \sum_{x \in S_i} ||x - \mu_i||^2$$

where μ_i is the mean of points in S_i.

Given an initial set of k means $m_1^{(1)}, \dots, m_k^{(1)}$, the steps of K-means clustering algorithm are as following:

Assignment step: Assign each observation to the cluster whose mean yields the least within-cluster sum of squares (WCSS). Since the sum of squares is the squared Euclidean distance, this is intuitively the "nearest" mean.

$$S_i^{(t)} = \{x_p : ||x_p - m_i^{(t)}||^2 \leq ||x_p - m_j^{(t)}||^2 \forall j, 1 \leq j \leq k \},$$

where each x_p is assigned to exactly one $S^{(t)}$, even if it could be is assigned to two or more of them.

Update step: Calculate the new means to be the centroids of the observations in the new clusters.

$$m_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{x_j \in S_i^{(t)}} x_j$$

Since the arithmetic mean is a least-squares estimator, this also minimizes the within-cluster sum of squares (WCSS) objective.

The algorithm has converged when the assignments no longer change. Since both steps optimize the WCSS objective, and there only exists a finite number of such partitioning, the algorithm must converge to a local optimum.

EM algorithm:

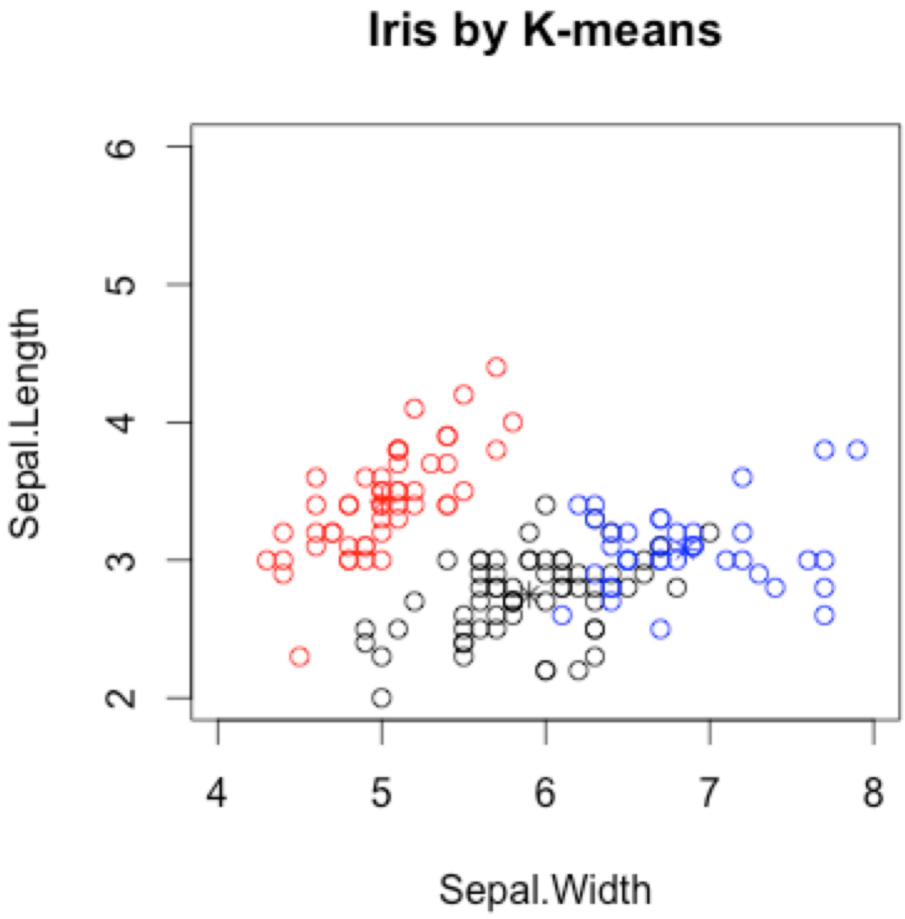
The EM algorithm seeks to find the maximum likelihood estimate of the marginal likelihood by iteratively applying the following two steps:

1. Expectation step (E step): Calculate the expected value of the log likelihood function, with respect to the conditional distribution of Z given X under the current estimate of the parameters

2. Maximization step (M step): Find the parameter that maximizes this quantity.

4. Results and Discussions

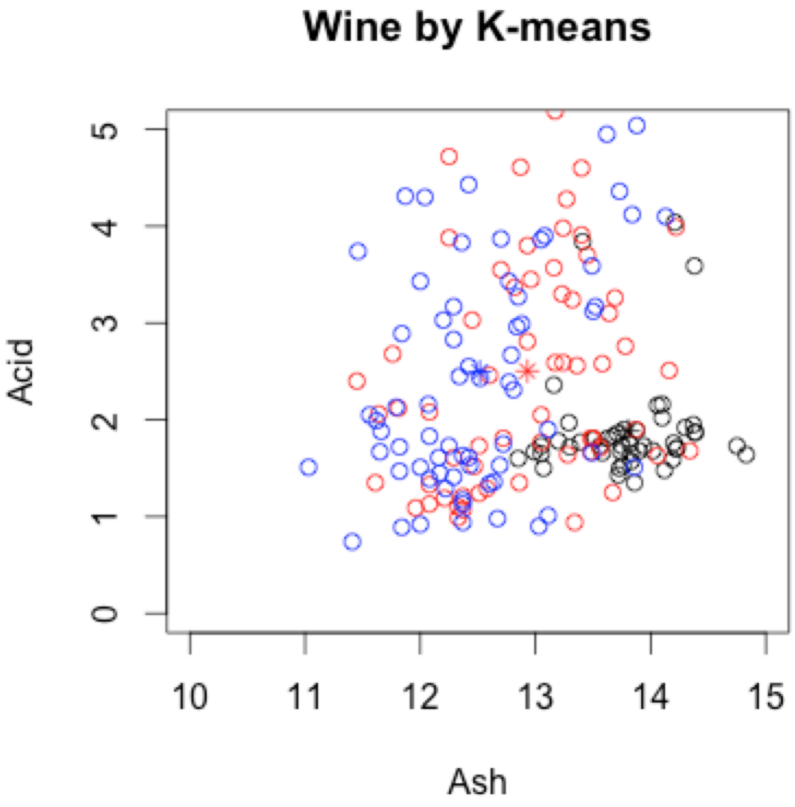
(1) K-means algorithm on Iris Dataset



	k	Euclidean Distance
[1,]	2	128.33667
[2,]	3	97.22487
[3,]	4	84.55605
[4,]	5	76.44783
[5,]	6	74.42394
[6,]	7	68.99989
[7,]	8	65.91021
[8,]	9	61.96843
[9,]	10	57.83311

We can see that if we reduce the parameter k, we can better performance in k-means algorithm.

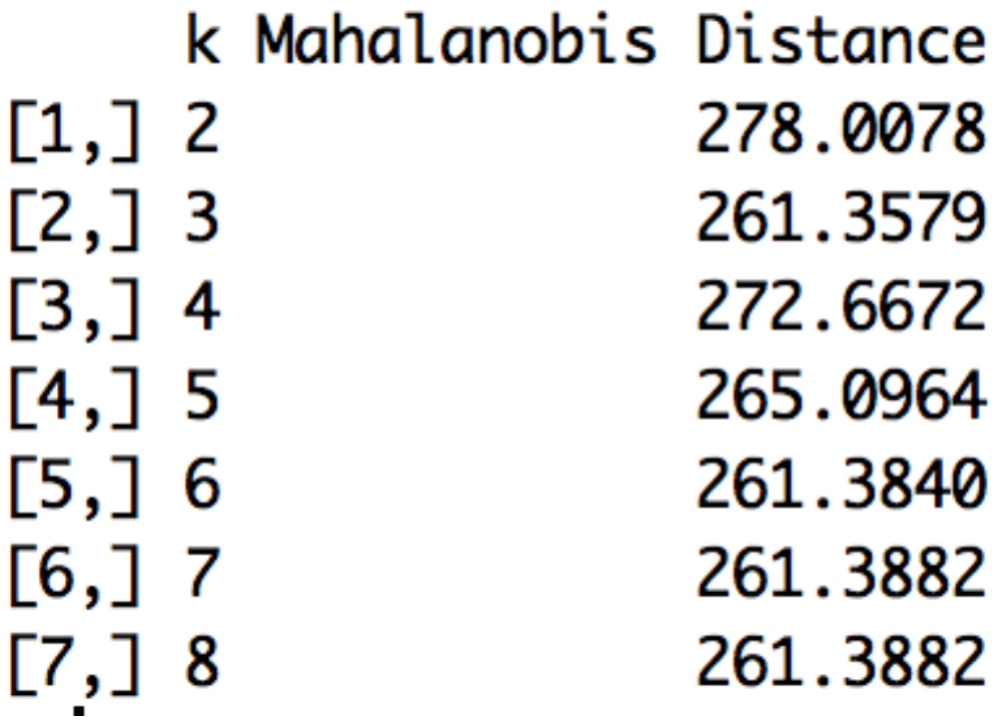
(2) K-means algorithm on wine Dataset



```
> result
      k Performance
 [1,]  2   23755.049
 [2,]  3   16555.679
 [3,]  4   12833.390
 [4,]  5   10766.475
 [5,]  6    8731.915
 [6,]  7    7923.037
 [7,]  8    7439.574
 [8,]  9    7163.171
 [9,] 10    5447.436
```

We can see that since the classes in the wine dataset are not well separated, the three clusters we got interleaved and the Euclidean Distances are quite long.

(3) EM algorithm on Iris Dataset



	k	Mahalanobis Distance
[1,]	2	278.0078
[2,]	3	261.3579
[3,]	4	272.6672
[4,]	5	265.0964
[5,]	6	261.3840
[6,]	7	261.3882
[7,]	8	261.3882

We can see that when the k is large, if we increase k, the improvement of the Distance is quite small. And when the k is equal 3, which is the true number of the Iris classes, the performance is optimal. Therefore, we can conclude the EM algorithm has a better performance than K-means on Iris dataset.

(4) EM algorithm on Wine Dataset

	k	Mahalanobis Distance
[1,]	2	334.7920
[2,]	3	334.7813
[3,]	4	334.7706
[4,]	5	334.7600
[5,]	6	334.7494
[6,]	7	334.7385

Since the wine dataset has three non-separable dataset, we can see that the performance got little improved with the increase of k.