Assignment 2 – INF552

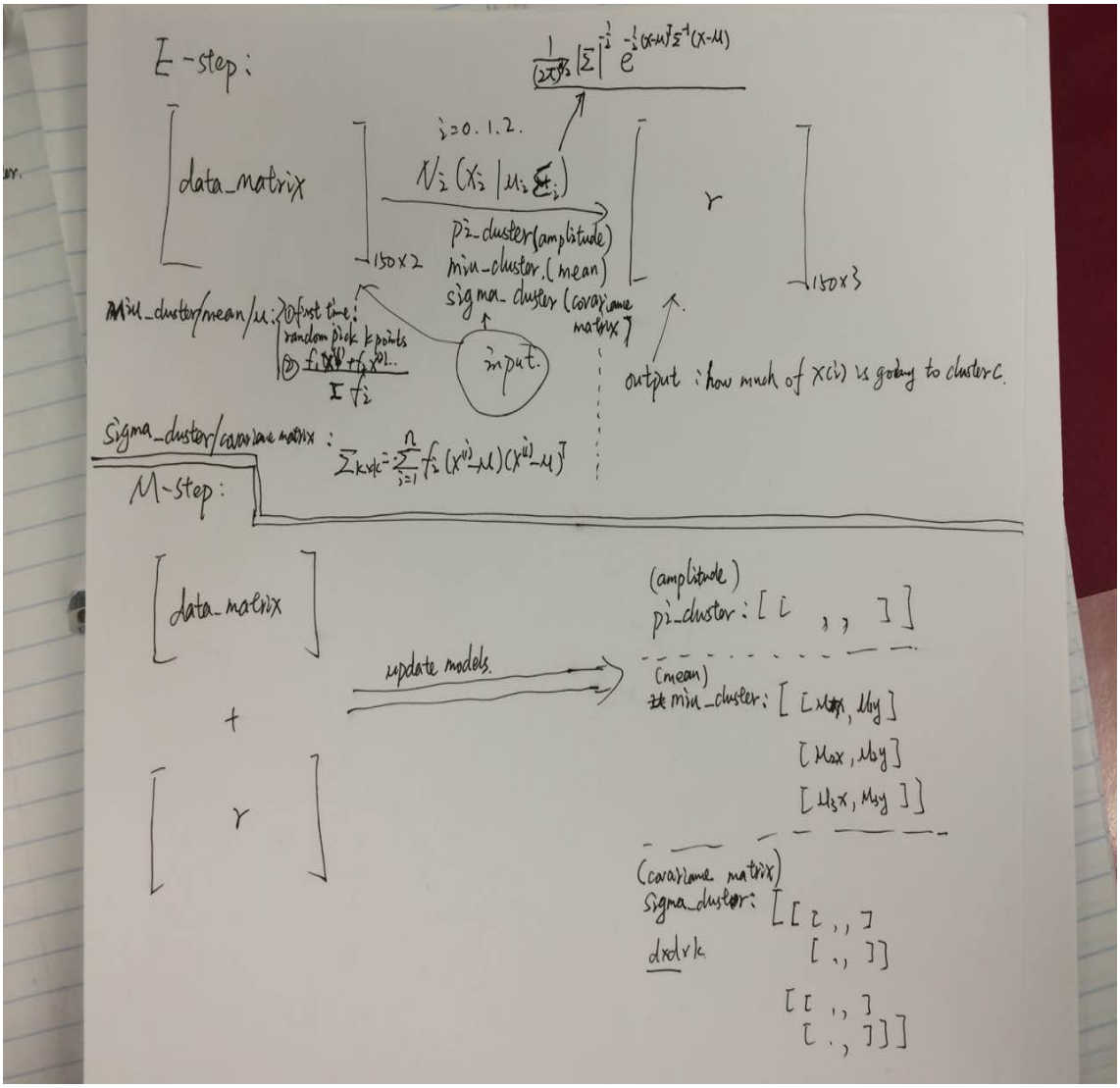
Yi Ren 5527205170
yiren@usc.edu

Part 1: Implementation

Haoran Que and I were in a group at beginning, we discussed the homework and he use Matlab but I use Python. Because both of us finished all the parts of the homework so we decide to separate. But we gain knowledge when we discussed.

K-cluster:
(1). Language: Python 2.7.11
(2). Tool: Mac + IDLE
(3). Data structure:
data: Array[Array[]] (matrix like; 150(number_of_point) X 2(dimension))
centroid: Array[] (matrix like; length of K)
cluster: Array[Array[]] (matrix like)
(4). Logic:
Step1: get centroid by random pick up three points
Step2: calculate clusters by compare the distances of each point to each cluster
Step3: update the centroid of the new cluster
Step4: repeat step2 and step3
(5). Optimization:
When compare whether the clusters or not change, I compare the centroid, which is a list with three element, not the clusters, which is a two dimension long list.

EM: GMM
(1). Language: Python 2.7.11
(2). Tool: Mac + IDLE
(3). Data structure:
data_matrix: Array[Array[]] (matrix like; 150(number_of_point) X 2(dimension))
miu_cluster: Array[Array[]] (matrix like)
sigma_cluster: Array[Array[Array[]]] (matrix like)
pi_cluster: Array[Array[]] (matrix like)
r: Array[Array[]] (matrix like)
(4). Logic:



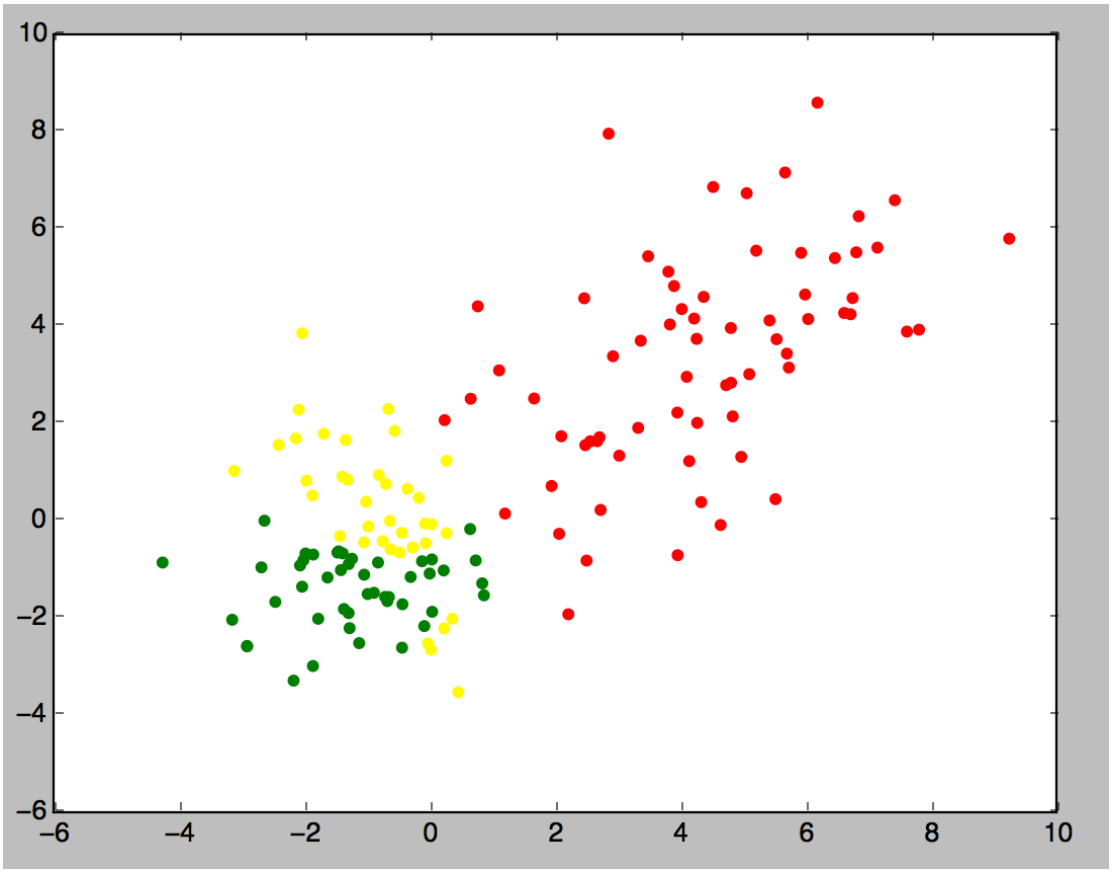
(5). Optimization: I can choose E-step first or M-step first. However, if I choose M-step first, I have to random pick up 50 points for each cluster, and the three Gaussian models will be very similar, which means there will be more times of loop. Therefore, I choose E-step firstly to minimize loop times and random pick up 3 points as miu_cluster(initial mean)

(6). In this algorithm, because the (x-miu) is a horizontal vector, the covariance matrix could be negative. For example: (x-miu) = [-1, 1], so $(x\text{-miu})^T$ * (x-miu) has negative element.

Compare the results of the two algorithms.

(1). I run both of the algorithm more than 20 times, I find that the similarity of the K-Means result are higher than GMM, which means the k-means are more stable.

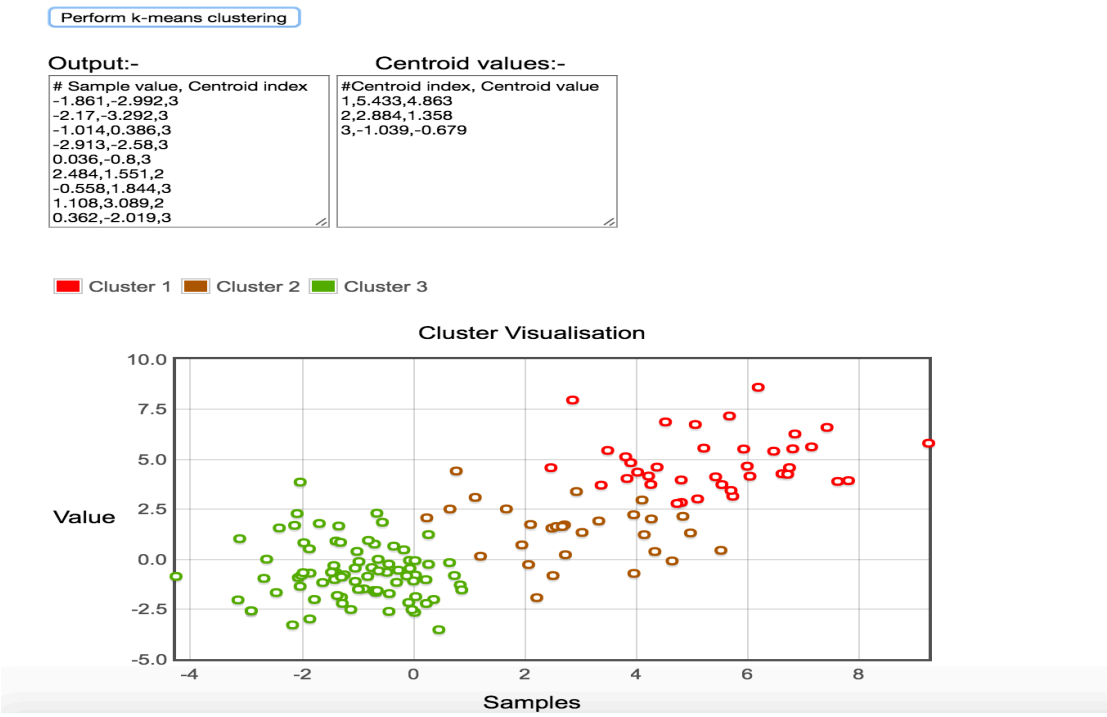
(2). There may be some overlapping between clusters when implement GMMs: like the clusters of yellow point and cluster of green points, the yellow points are separate by green points.

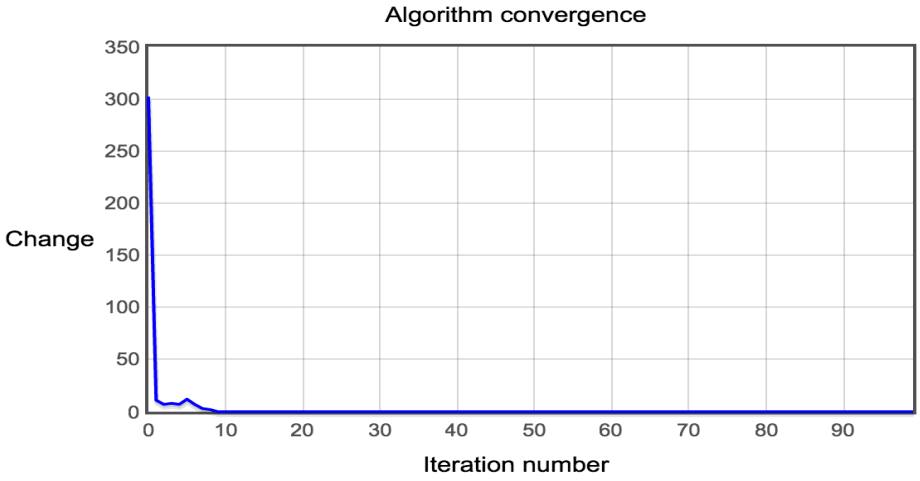


part 2:

Package for k-means:

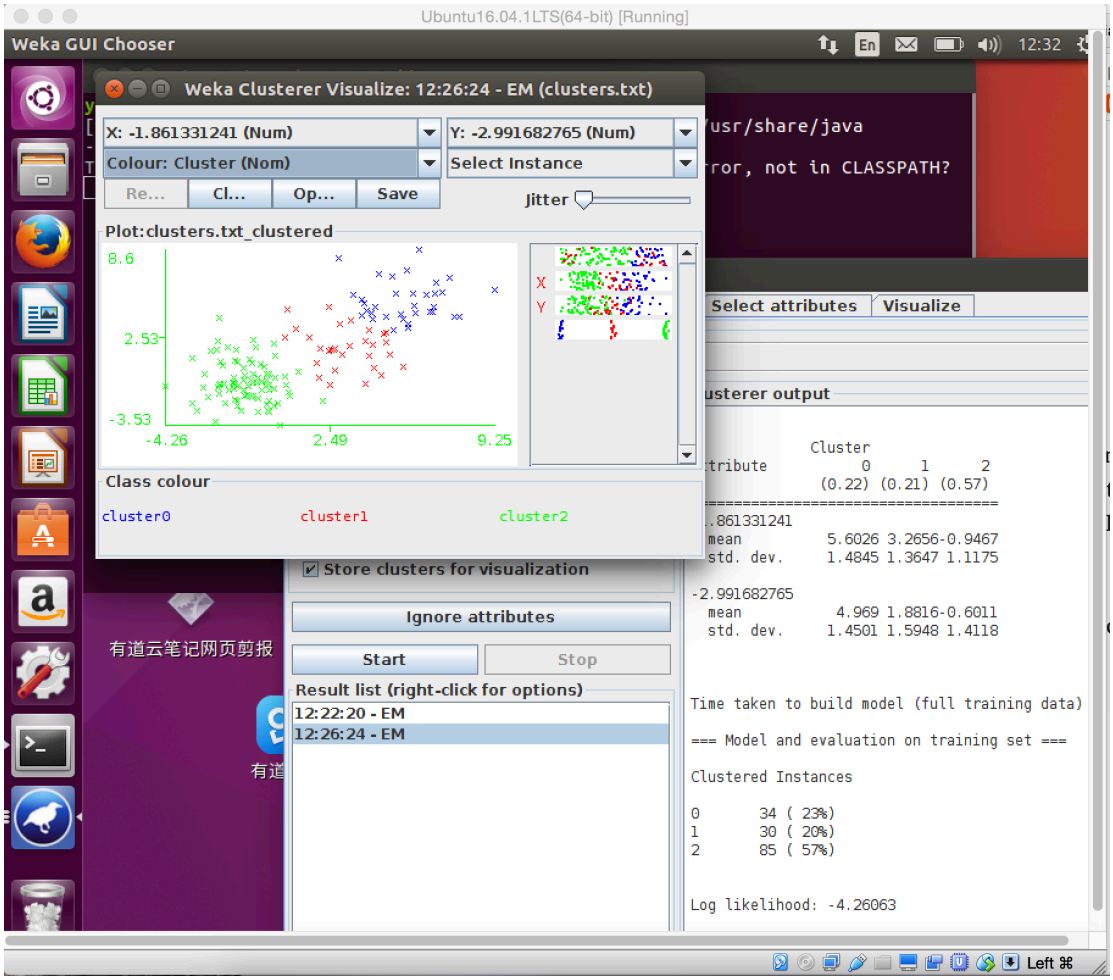
On the internet, there are some online k-means calculators, which are very easy to use:





The advantage of this online package is very easy to use, and it gives you the convergence information.
The disadvantage of this online package is that the input has iteration times, which means I have to tell how many time should it iterate.

Package for EM:
weka is a good package for implement EM:



The first advantage of Weka when implement EM is effective, for a data set like

clusters.txt, it only take 0.56 second according to the software. I can set a bigger threshold to decrease the times of loop to increase the efficiency.
The second advantage of Weka is flexibility, you can easily change x_axis and y_axis.
The third advantage if Weka is that it produce a picture of value distribution on each dimension (the picture under “Jitter”).

part 3:

(1). Application of k-means:
Market segmentation: divide customer into different segmentation and use different ways to make promotion to them.
(2). Application of GMMs:
Statistic population: when we want to statistic the heights of people of different ethnicities, such as African-American, Asian, Caucasian, and Latino, we can implement every GMM to each ethnicity because the heights of people in each ethnicity have Normal Distribution. Then, there would be four GMMs in total.