Report

When k is equal to 2, we are implementing 2-means algorithm on the dataset for random choosing 2 centroids on the existing data point. I ran the algorithm 100 times based on different initial centroid points. When the seed value is set to be 5, 29, 33 and 88, we end up getting total 60 misclassified points for both clusters with converged centroids, which is the minimum number of misclassified value I have found. One cluster has 10 positive values and 162 negative values. Another cluster has 347 positive values and 50 negative values. From the result, we can see that even through the algorithm is trying to classify the data points into different clusters, the actual percentage of number of data points being positive is 5.8% for cluster one and 87.4% for cluster two. If we denote cluster one to be negative cluster and cluster two to be positive cluster. Clearly we can see that the percentage of the number of data points being misclassified is 5.8% for cluster one and 12.6% for cluster two. Because we are setting the initial centroids randomly, we cannot get lucky every time to get this optimal result from 100 runs. At most time, we are end up getting a one cluster has 2 positive and 120 negative, the other one has 355 positive and 90 negative. So the percentage of cluster one has positive is 1.64% and the percentage of cluster two has positive is 79.8%. If we denote cluster one to be negative cluster and cluster two to be positive cluster. Clearly we can see that the percentage of the number of data points being misclassified is 1.64% for cluster one and 20.2%. And we can easily find out this total error rate is higher than what we get before. Thus we can conclude that even though this algorithm is trying to get different cluster separated, but the best result from 100 runs is 5.8% errors for cluster one and 12.6% errors for cluster two. The reasons can be different, it maybe because this algorithm is not good enough to separate this dataset or maybe because this dataset is not well separated itself.