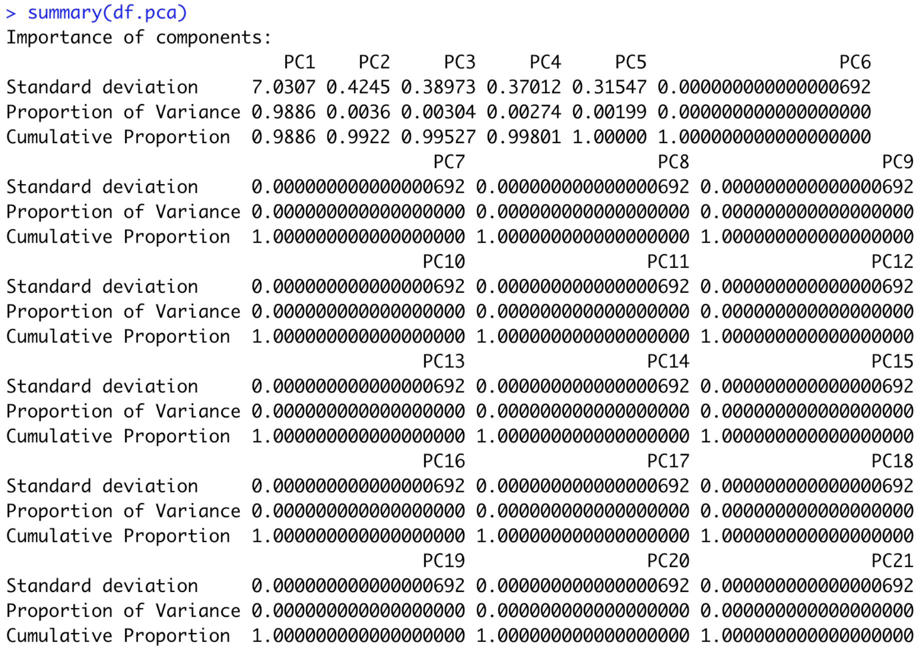
Problem 4

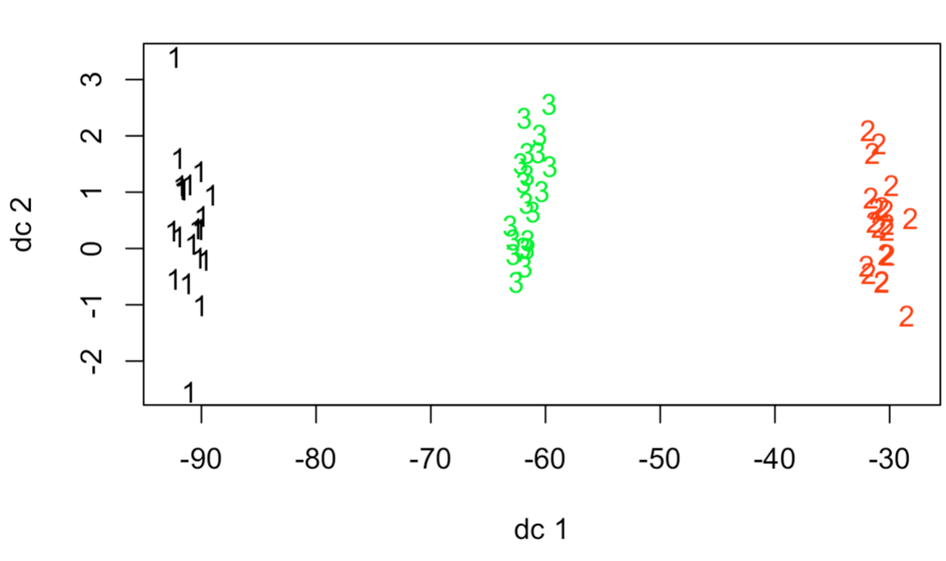
a) After running PCA on the 60x50 created matrix, the following are the components retrieved:



```
> summary(df.pca)
Importance of components:
                          PC1    PC2     PC3     PC4     PC5                  PC6
Standard deviation     7.0307 0.4245 0.38973 0.37012 0.31547 0.000000000000000692
Proportion of Variance 0.9886 0.0036 0.00304 0.00274 0.00199 0.000000000000000000
Cumulative Proportion  0.9886 0.9922 0.99527 0.99801 1.00000 1.000000000000000000
                                        PC7                  PC8                  PC9
Standard deviation     0.000000000000000692 0.000000000000000692 0.000000000000000692
Proportion of Variance 0.000000000000000000 0.000000000000000000 0.000000000000000000
Cumulative Proportion  1.000000000000000000 1.000000000000000000 1.000000000000000000
                                       PC10                 PC11                 PC12
Standard deviation     0.000000000000000692 0.000000000000000692 0.000000000000000692
Proportion of Variance 0.000000000000000000 0.000000000000000000 0.000000000000000000
Cumulative Proportion  1.000000000000000000 1.000000000000000000 1.000000000000000000
                                       PC13                 PC14                 PC15
Standard deviation     0.000000000000000692 0.000000000000000692 0.000000000000000692
Proportion of Variance 0.000000000000000000 0.000000000000000000 0.000000000000000000
Cumulative Proportion  1.000000000000000000 1.000000000000000000 1.000000000000000000
                                       PC16                 PC17                 PC18
Standard deviation     0.000000000000000692 0.000000000000000692 0.000000000000000692
Proportion of Variance 0.000000000000000000 0.000000000000000000 0.000000000000000000
Cumulative Proportion  1.000000000000000000 1.000000000000000000 1.000000000000000000
                                       PC19                 PC20                 PC21
Standard deviation     0.000000000000000692 0.000000000000000692 0.000000000000000692
Proportion of Variance 0.000000000000000000 0.000000000000000000 0.000000000000000000
Cumulative Proportion  1.000000000000000000 1.000000000000000000 1.000000000000000000
```

(In total 50 new features are there)

Following is the plot of the first two principle axis:

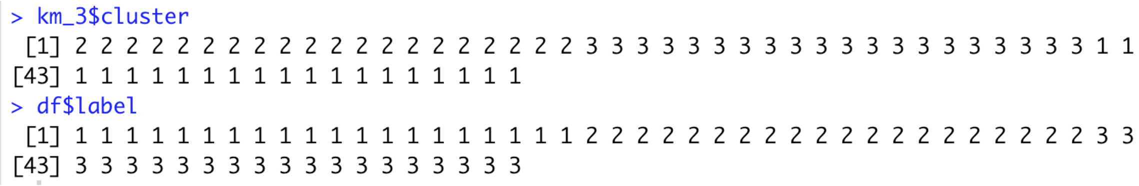


The three classes are distinctive enough.

b) After running K-means clustering with K = 3 and nstart = 100 (with 100 different starting centroids), the following are the classification results:

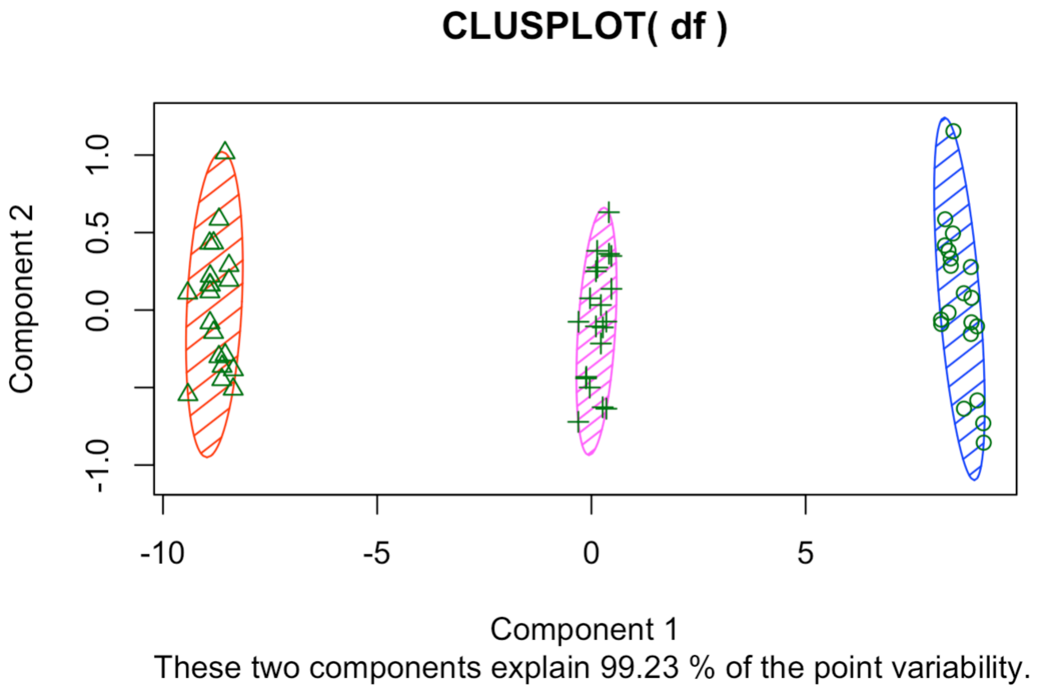
```
> table(km_3$cluster)

 1  2  3 
20 20 20 
> table(df$label)

 1  2  3 
20 20 20 
```

The following shows that the labels have been assigned differently:



```
> km_3$cluster
 [1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 1 1
[43] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
> df$label
 [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3
[43] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
```

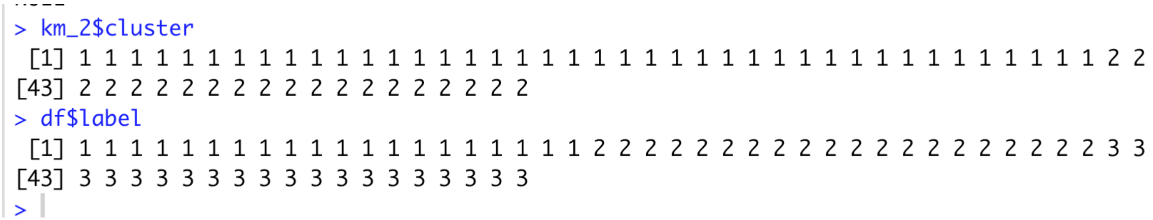
This does not affect our learning as the goal of clustering is to separate the observations of an unlabeled data. The obtained clusters and true class labels are the same.



c) K-means clustering with K=2:

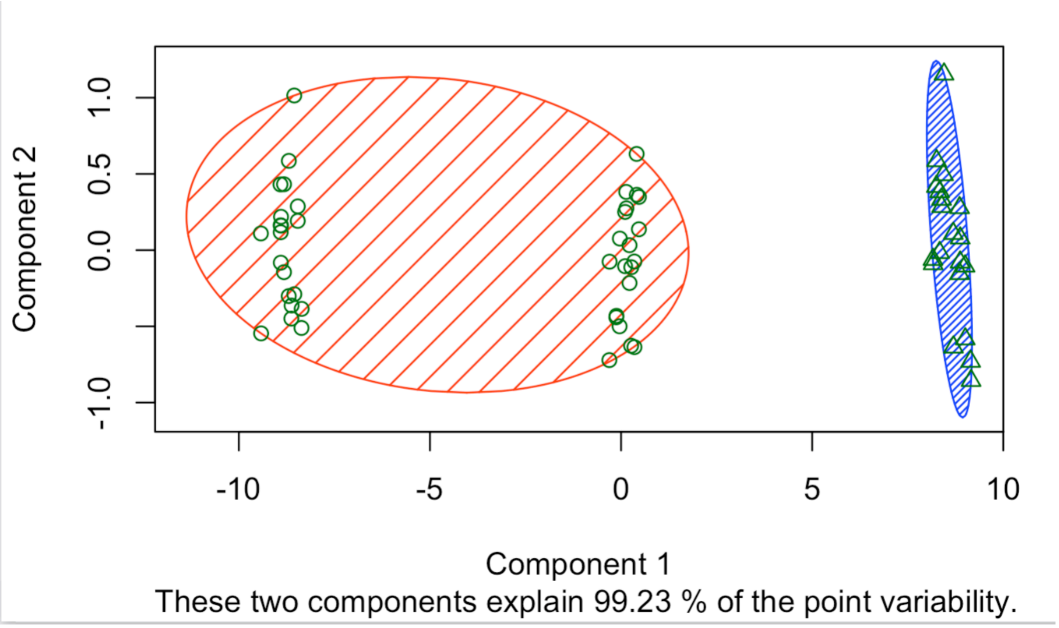
```
> table(km_2$cluster)

 1  2 
40 20 
> table(df$label)

 1  2  3 
20 20 20 
```

The class labels are as follows:

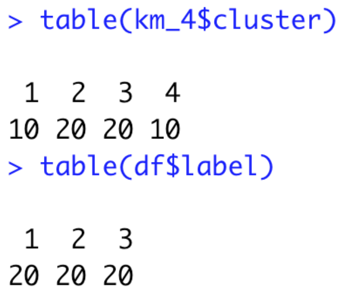


```
> km_2$cluster
 [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2
[43] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
> df$label
 [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3
[43] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
> 
```

The data has 3 actual labels, now that we the algorithm is forced to assign the observation into 2 clusters; it combines the data points of classes ‘1’ and ‘2’ in ‘1’ and renames label of ‘3’ as ‘2’. This indicates that 3 clusters are ideal in the situation and 2 forced clusters combines the 2/3 clusters into 1.

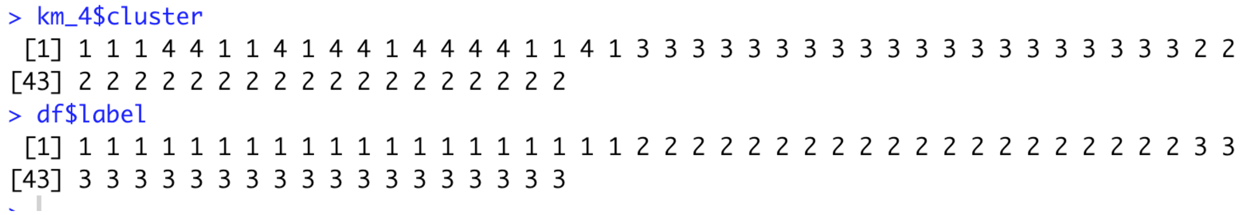


K- means clustering with K=4:



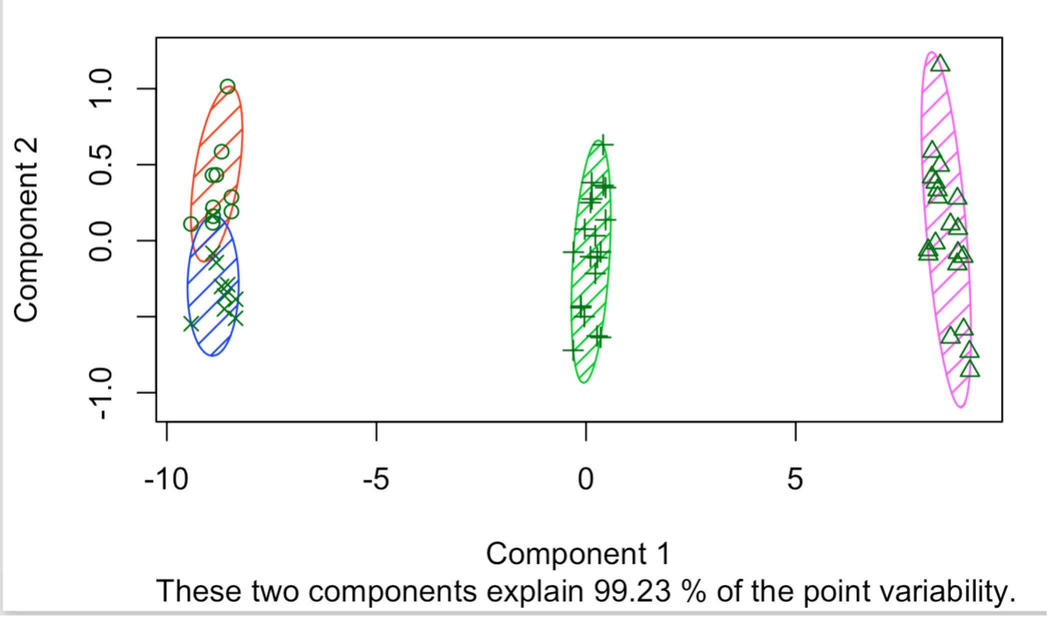
```
> table(km_4$cluster)

 1  2  3  4 
10 20 20 10 
> table(df$label)

 1  2  3 
20 20 20 
```

Now the data of true labeled ‘1’ has been divided into clusters ‘1’ and ‘4’ due to forceful 4 clustering. The following can be seen here:

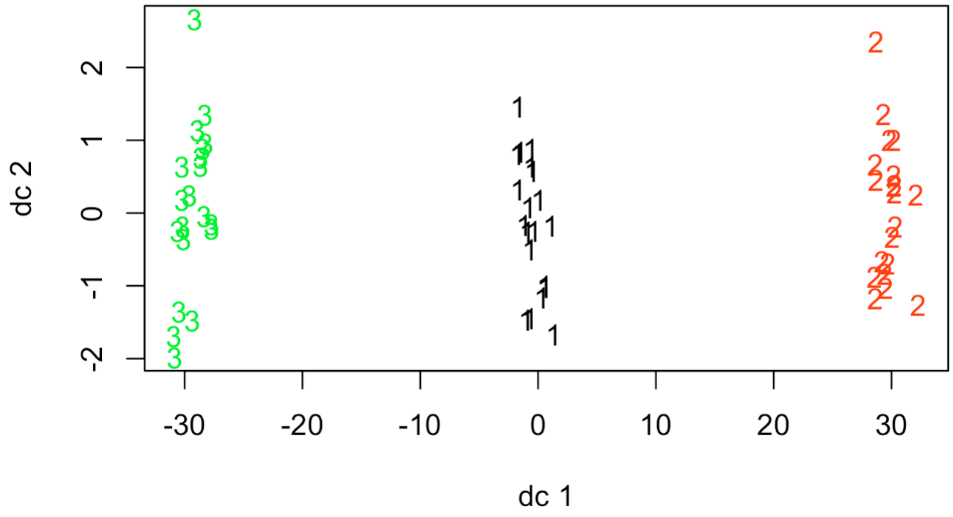


```
> km_4$cluster
 [1] 1 1 1 4 4 1 1 4 1 4 4 1 4 4 4 4 1 1 4 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 2 2
[43] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
> df$label
 [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3
[43] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
```

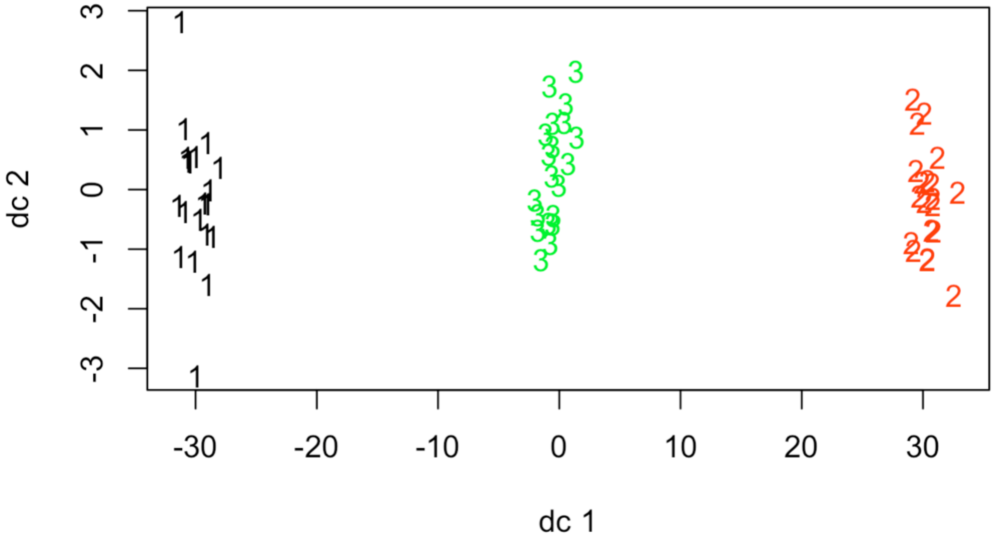
The above two indicates that the first 10 observations (true labelled ‘1’ have the maximum distance amongst each other. Simultaneously, these observations are similar to the next 10 observations, as compared to the similarity between the last two 10 observations.



d) Running K-means clustering with K=3 on 2 principle axis gives accurate results. The difference is that the features have been reduced from 50 to 2, making it easier to classify accurately every time. The clusters can be seen to drift away from each other:



e) The following are the clusters with scaled data:



The data points in respective clusters have shrunk closer to the centroid of the cluster, indicating easier classification due to scaled data.