homework7

Sunny Lee

3/30/2021

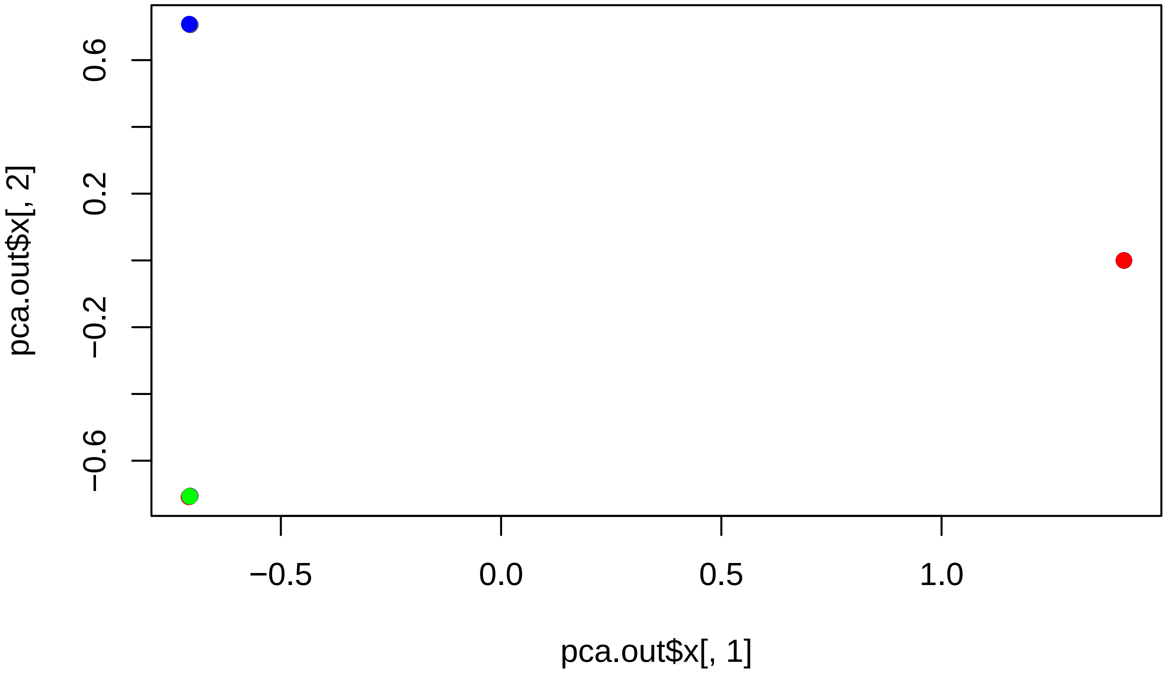
10)

a)

```
set.seed(2)
x <- matrix(rnorm(20*3*50, mean=0, sd=0.001), ncol=50)
x[1:20, 2] <- 1
x[21:40, 1] <- 2
x[21:40, 2] <- 2
x[41:60, 1] <- 1
```

b)

```
pca.out <- prcomp(x)
#summary(pca.out)
plot(pca.out$x[, 1], pca.out$x[, 2], col = c("red", "green", "blue"), pch = 19)
```



c)

```
km.out3 <- kmeans(x, 3, nstart = 20)
table(km.out3$cluster, c(rep(1,20), rep(2,20), rep(3,20)))
```

```
##
##       1  2  3
##   1  0 20  0
##   2  0  0 20
##   3 20  0  0
```

From our table, we see that each of the classifications have exactly 20 observations. Though they are "misclassified", this table will depend on where the centroids are initialized. What is important is that our k-means clustering has correctly separated our data into three clusters.

d)

```
km.out2 <- kmeans(x, 2, nstart = 20)
table(km.out2$cluster, c(rep(1,20), rep(2,20), rep(3,20)))
```

```
##
##       1  2  3
##   1 20  0 20
##   2  0 20  0
```

Using 2 clusters, we see that our data is still clustered quite well, but two of the clusters from the 3 have combined into 1 cluster.

e)

```
km.out4 <- kmeans(x, 4, nstart = 20)
table(km.out4$cluster, c(rep(1,20), rep(2,20), rep(3,20)))
```

```
##
##       1  2  3
##   1 11  0  0
##   2  9  0  0
##   3  0 20  0
##   4  0  0 20
```

Using 4 clusters, we see that we still have good separation, but now one of the clusters has been effectively split into two groups.

f)

```
km.pca <- kmeans(pca.out$x[, 1:2], 3, nstart = 20)
table(km.pca$cluster, c(rep(1,20), rep(2,20), rep(3,20)))
```

```
##
##       1  2  3
##   1  0  0 20
##   2  0 20  0
##   3 20  0  0
```

Running our k-means clustering on the Principal Components, we still see each cluster has exactly 20 observations, meaning our k-means clustering was just as effective at clustering the data from the principal components as it was from the whole data set. This means instead of running k-means clustering on a matrix of size 60x50, we can run our clustering algorithm on a matrix of size 60x2, cutting computational costs.

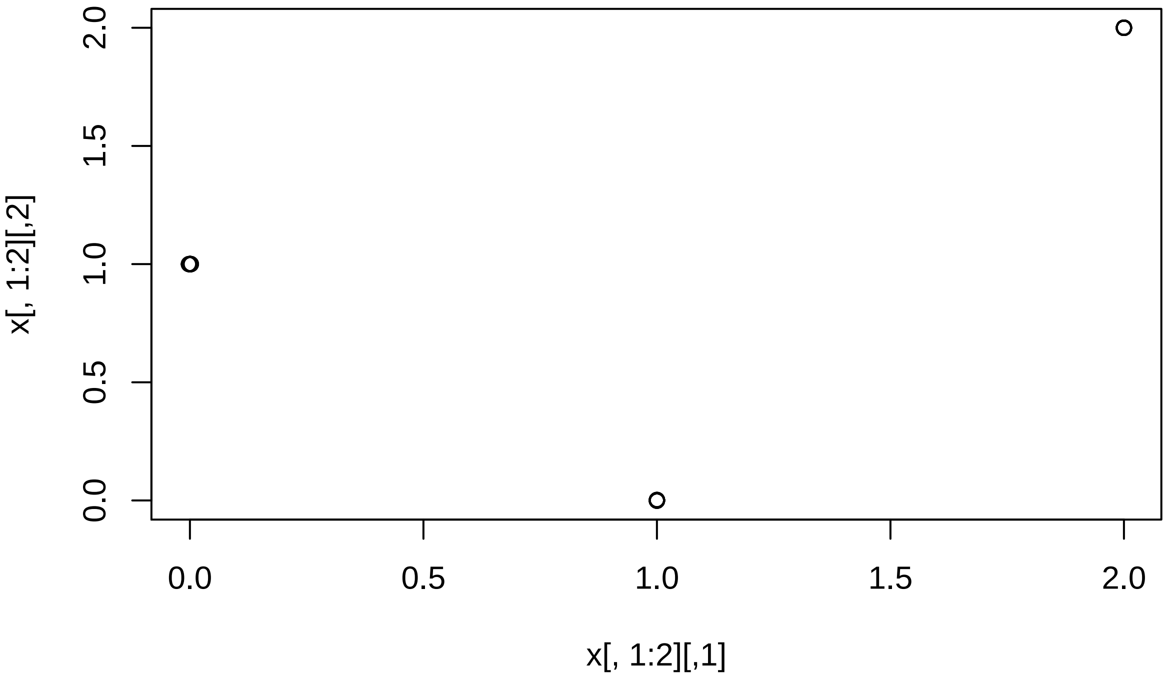
g)

```
km.out.scale <- kmeans(scale(x), 3, nstart = 20)
table(km.out.scale$cluster, c(rep(1,20), rep(2,20), rep(3,20)))
```

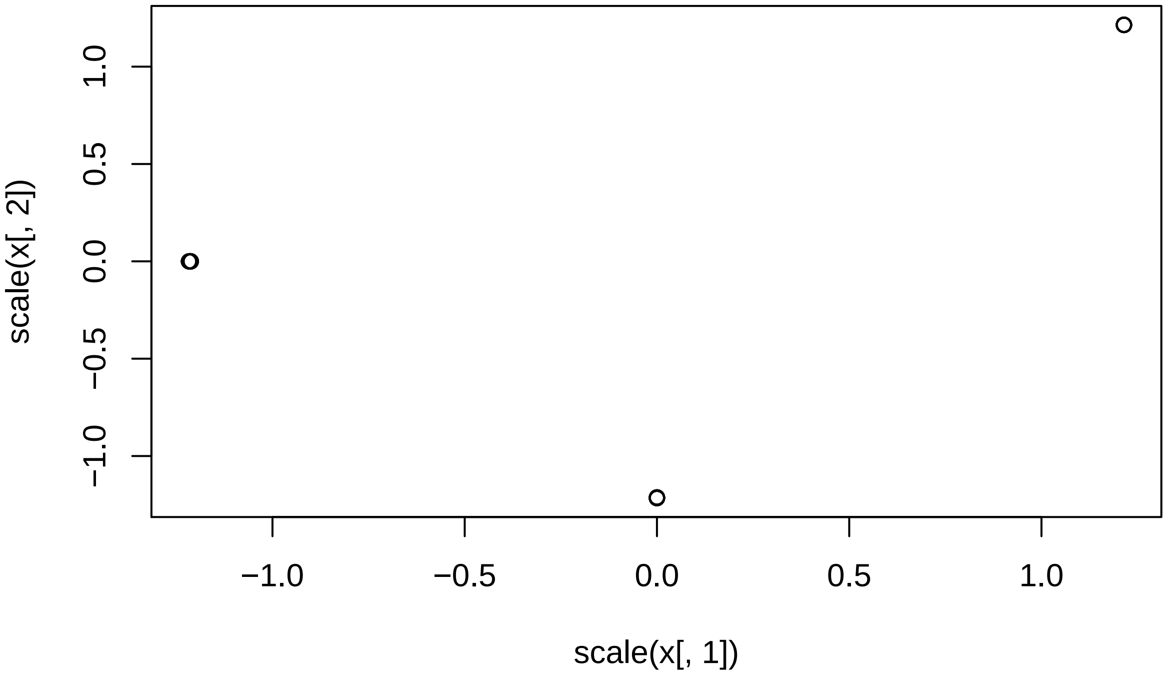
```
##
##      1  2  3
##   1  9  2  7
##   2  2 18  1
##   3  9  0 12
```

From the table above, we see many misclassifications. The data does not seem to be split into the three groups we made, and the three groups seem to have a mixture of each class. This makes sense since when we scale our observations, we bring everything closer together, making it harder for the centroids to split away from each other.

```
plot(x[, 1:2])
```



```
plot(scale(x[, 1]), scale(x[, 2]))
```


scale(x[, 2])
1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0
scale(x[, 1])