Statistical Learning Assignment 3

12111603 谭致恒

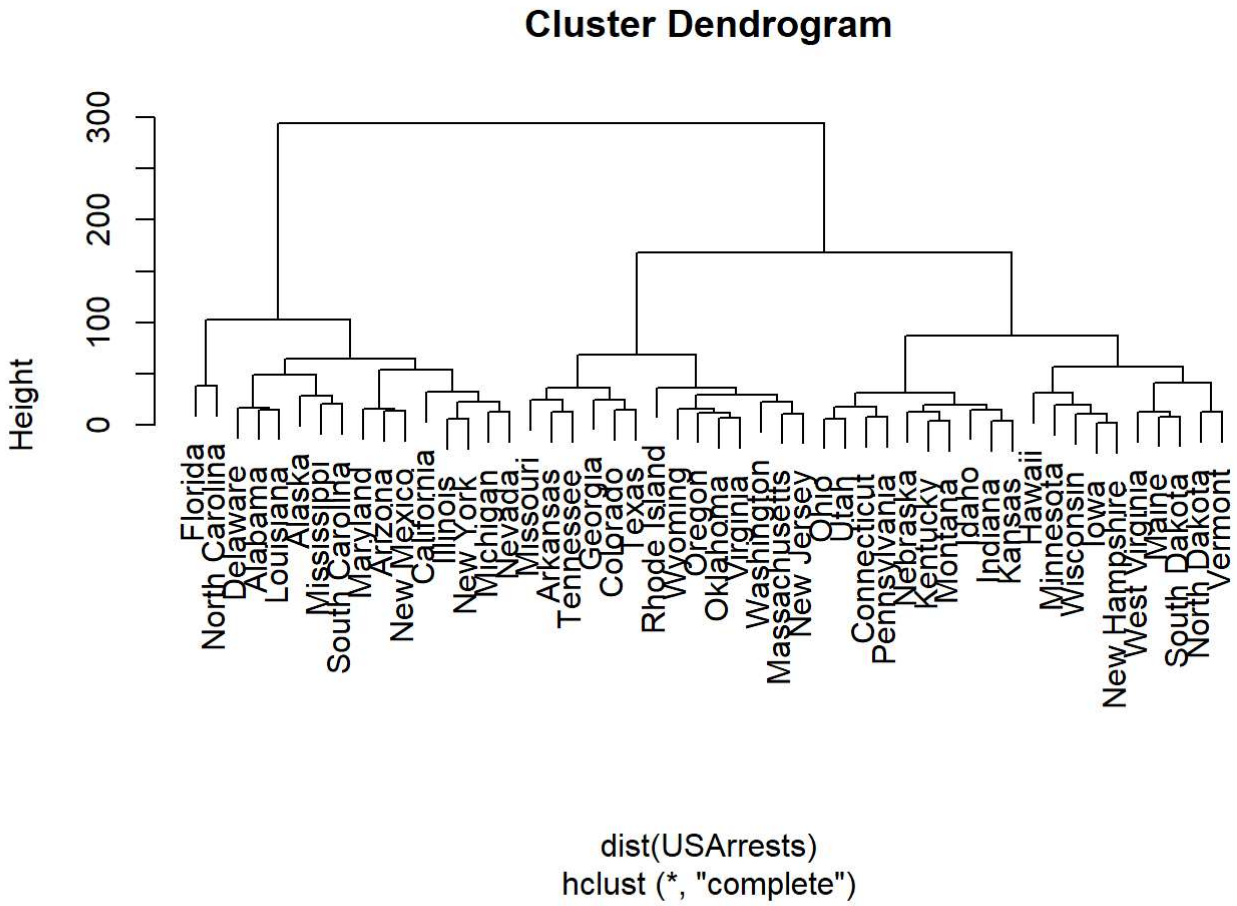
2023-12-07

ISLR Question 9

```
library(ISLR)
data("USArrests")
```

(a)

```
complete <- hclust(dist(USArrests), method = "complete")
plot(complete)
```



(b)

```
cut <- cutree(complete, k = 3)
library(tibble)
```

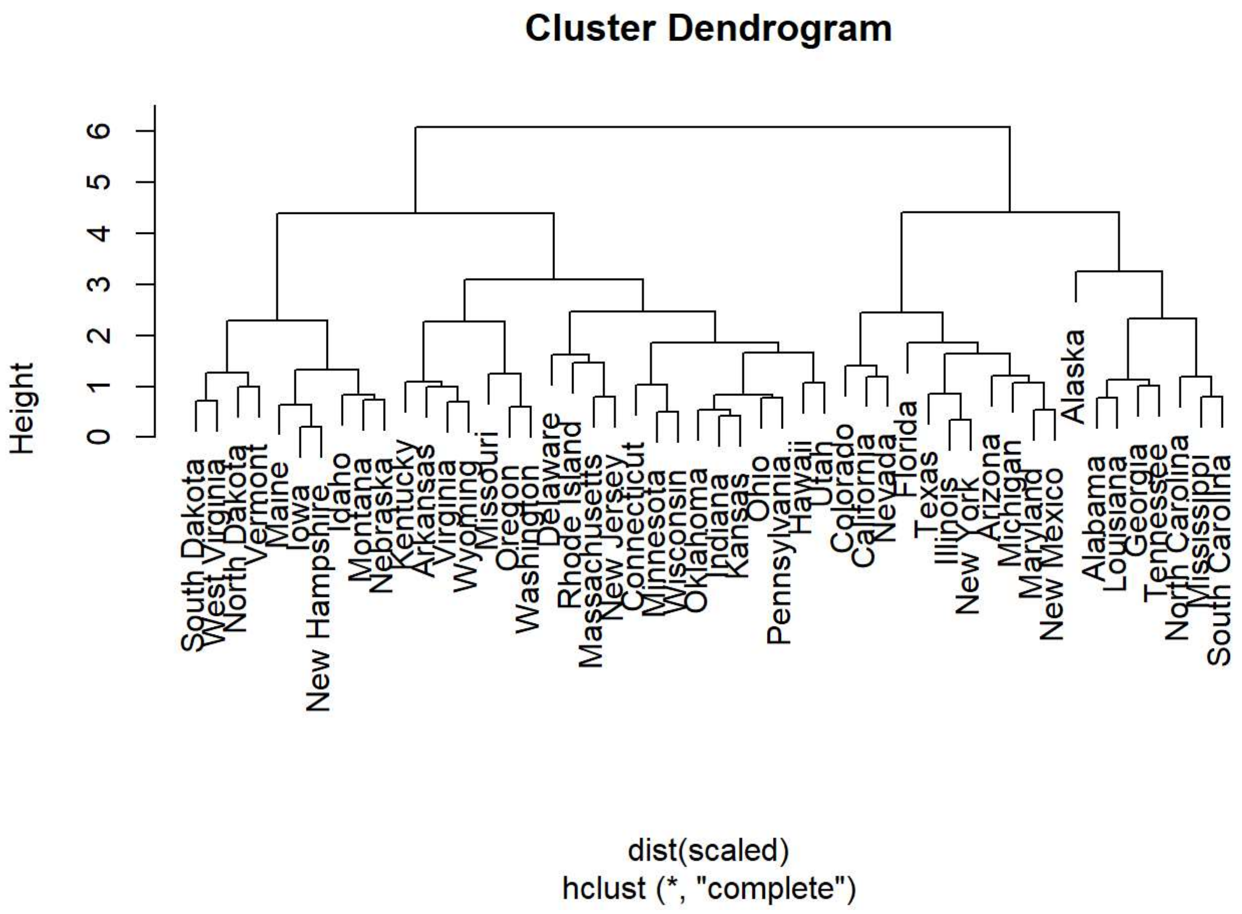
```
## Warning: 程辑包'tibble'是用R版本4.3.2 来建造的
```

```
tibble(states = rownames(USArrests), cluster = cut)
```

```
## # A tibble: 50 × 2
##    states      cluster
##    <chr>         <int>
##  1 Alabama           1
##  2 Alaska            1
##  3 Arizona           1
##  4 Arkansas          2
##  5 California        1
##  6 Colorado          2
##  7 Connecticut       3
##  8 Delaware          1
##  9 Florida           1
## 10 Georgia           2
## # ℹ 40 more rows
```

(c)

```
scaled <- scale(USArrests)
complete_scaled <- hclust(dist(scaled), method = "complete")
plot(complete_scaled)
```



(d)

- The scaled dendogram has greatly reduced height (the height is 300 before scaling while it is merely 6 after scaling), and the clusters obtained are somewhat different. However, the bushiness of the tree doesn’t appear to be affected.
- As a general rule, variables with different measurement units should be scaled before computing the inter-observation dissimilarities, which applies to this dataset.

ISLR Question 10

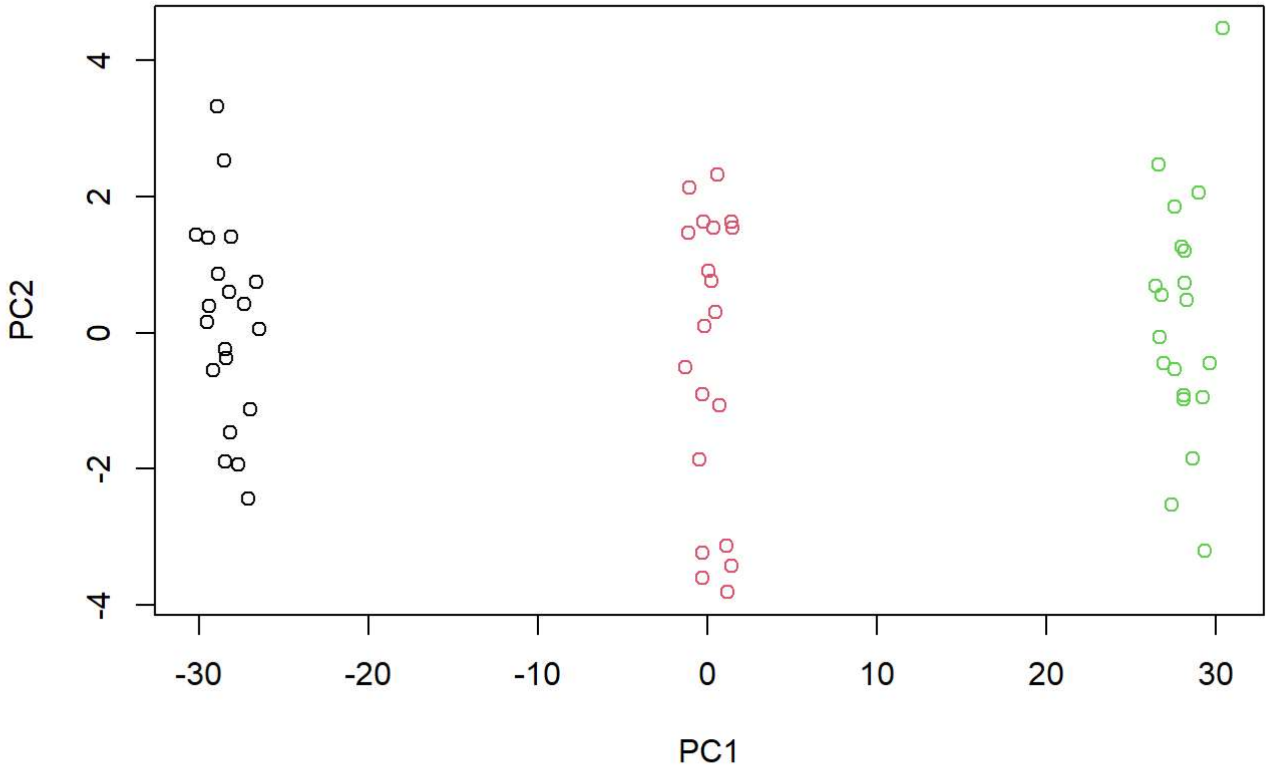
(a)

```
set.seed(123)
simulated_data <- matrix(c(
  rnorm(1000, mean = 1),
  rnorm(1000, mean = 5),
  rnorm(1000, mean = 9)
), ncol = 50, byrow = TRUE)
class = unlist(lapply(1:3,function(x){rep(x,20)}))
```

(b)

```
pca <- prcomp(simulated_data)

plot(pca$x[,1:2], col = class)
```

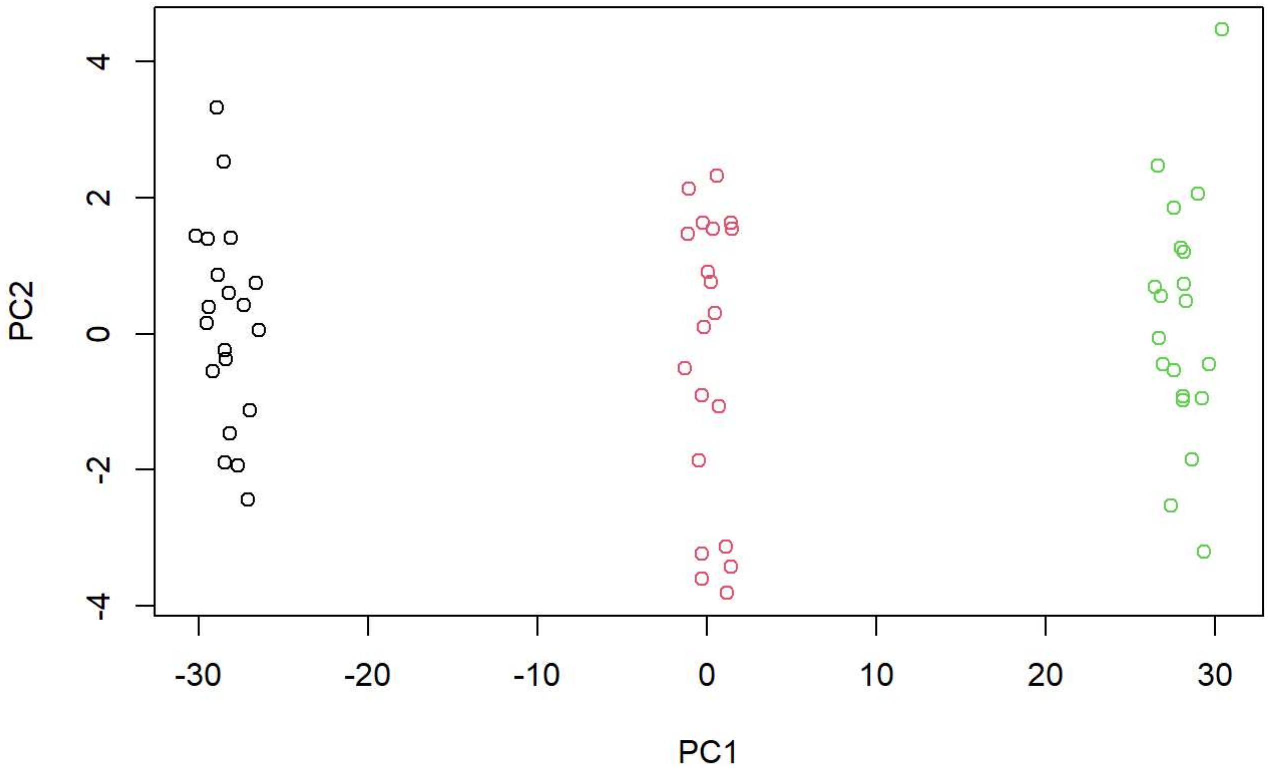


(c)

```
set.seed(124)
kmeans3 <- kmeans(simulated_data, centers = 3)
table(class,kmeans3$cluster)
```

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

```
plot(pca$x[,1:2], col = kmeans3$cluster)
```



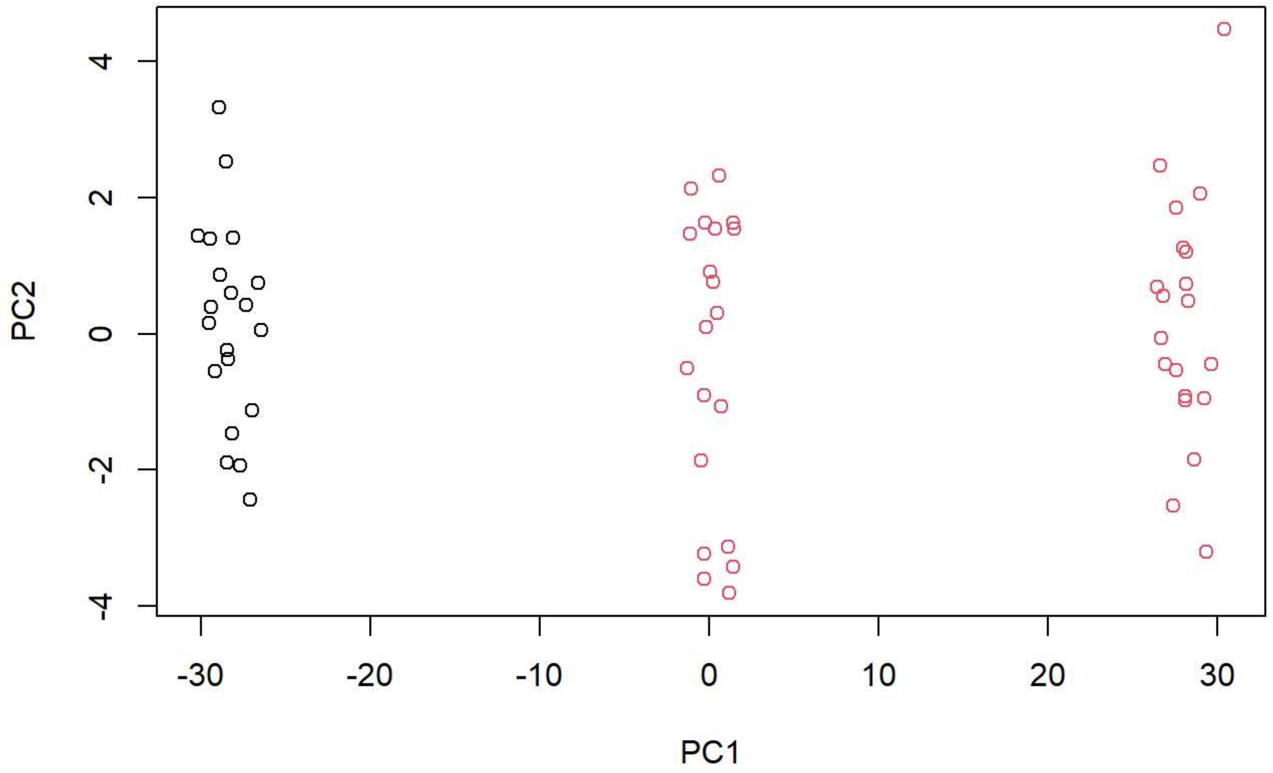
All the points are classified correctly.

(d)

```
set.seed(12)
kmeans2 <- kmeans(simulated_data, centers = 2)
table(class,kmeans2$cluster)
```

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

```
plot(pca$x[,1:2], col = kmeans2$cluster)
```



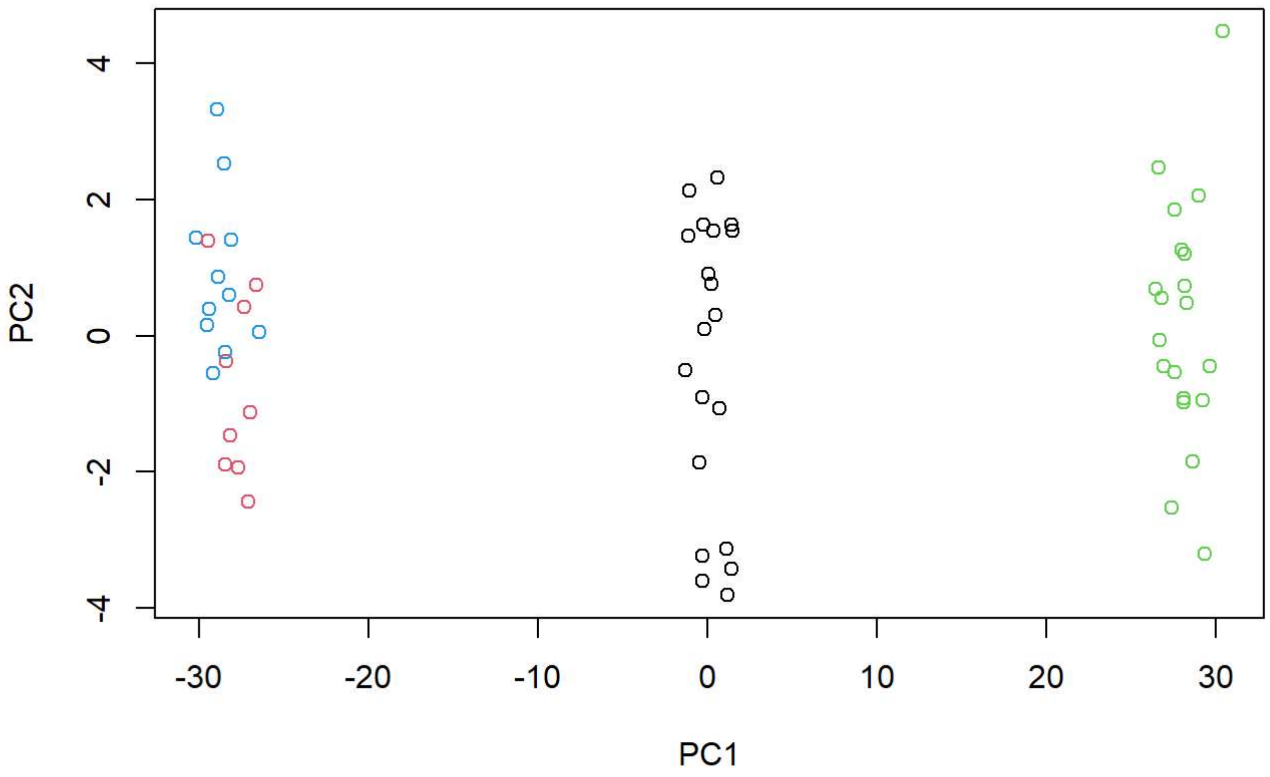
The points from class 1 are classified into one group while the points from class 2 and 3 are classified together into the other group.

(e)

```
set.seed(123)
kmeans4 <- kmeans(simulated_data, centers = 4)
table(class, kmeans4$cluster)
```

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

```
plot(pca$x[,1:2], col = kmeans4$cluster)
```



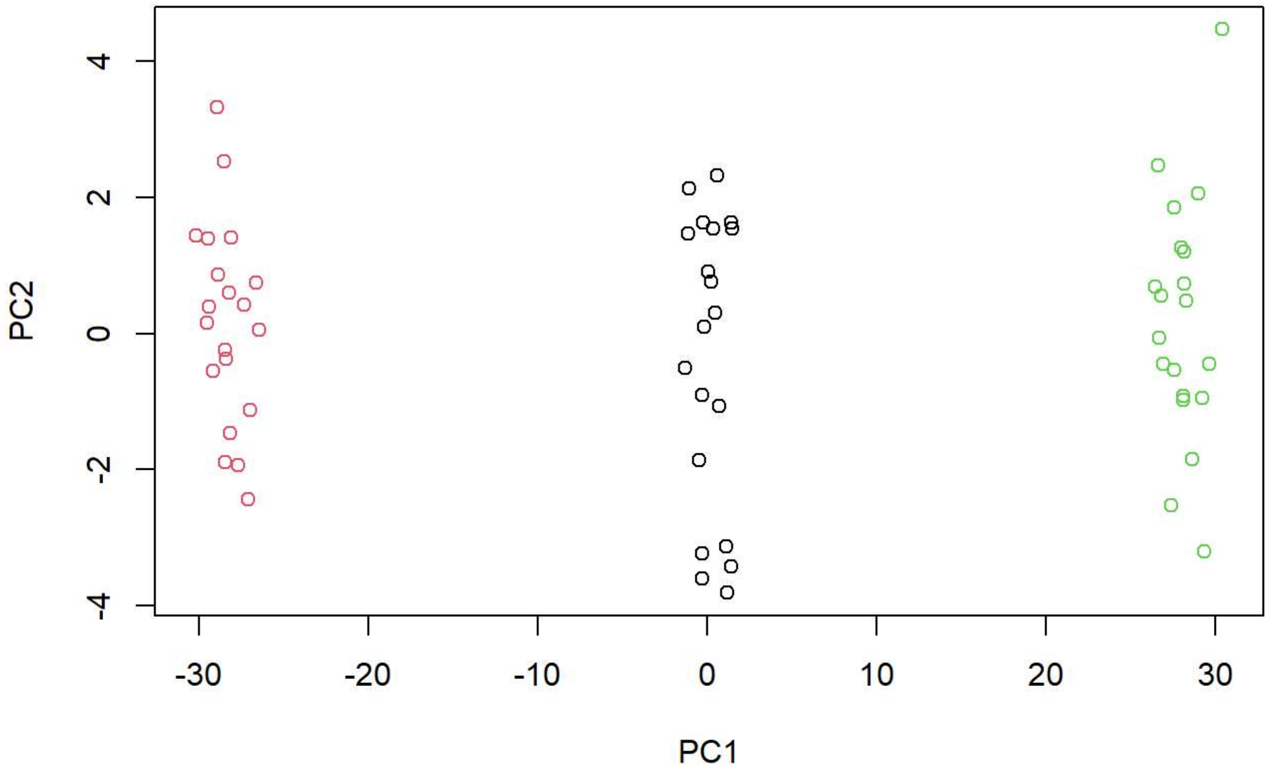
The points from class 2 are classified into one group seperately, points from class 3 are also classified into another group seperately while points from class 1 are divided into two groups.

(f)

```
set.seed(123)
kmeans_pca <- kmeans(pca$x[ ,1:2], centers = 3)
table(class, kmeans_pca$cluster)
```

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

```
plot(pca$x[ ,1:2], col = kmeans_pca$cluster)
```



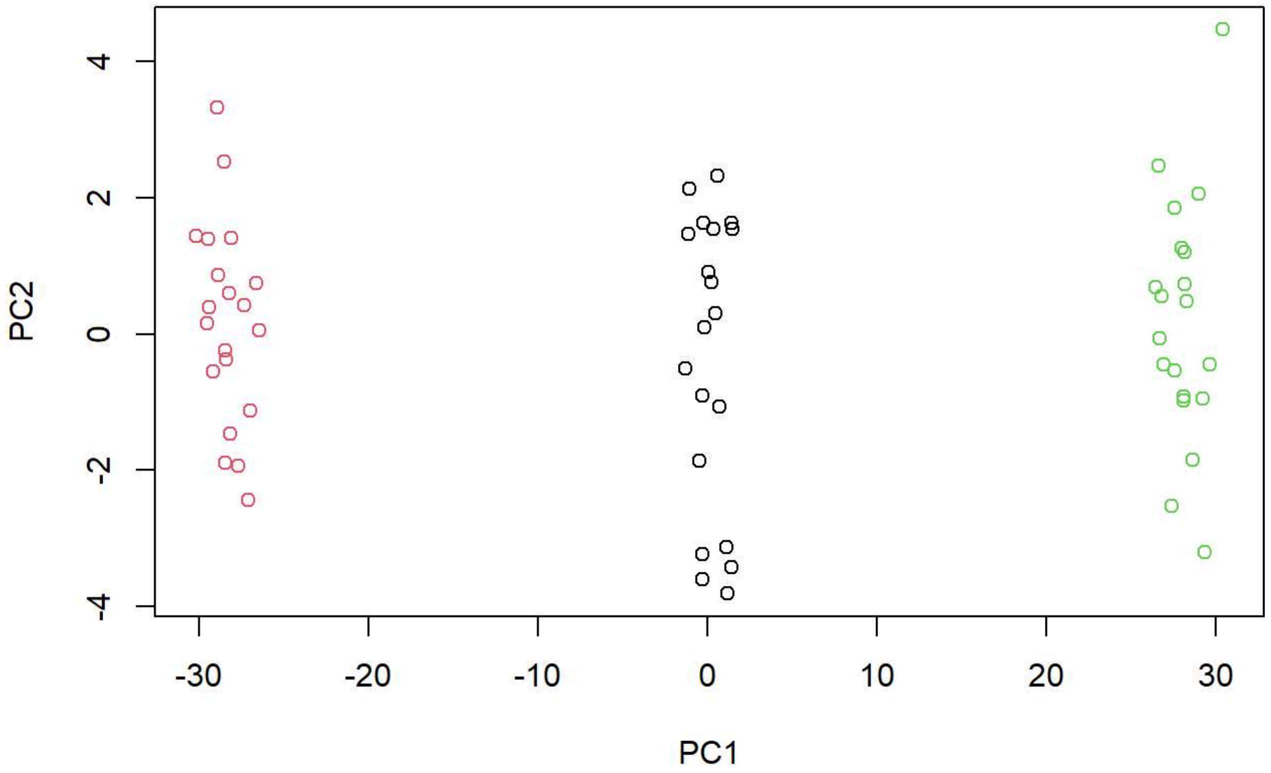
All the points are correctly classified.

(g)

```
set.seed(123)
scaled <- scale(simulated_data)
kmeans_scaled <- kmeans(scaled, centers = 3)
table(class, kmeans_scaled$cluster)
```

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

```
plot(pca$x[ ,1:2], col = kmeans_scaled$cluster)
```



The result is totally the same as the result in part (b), all the points are classified correctly. Likely because the simulated dataset created was very well separated. Datasets with overlapping observations would likely result in a different outcome.