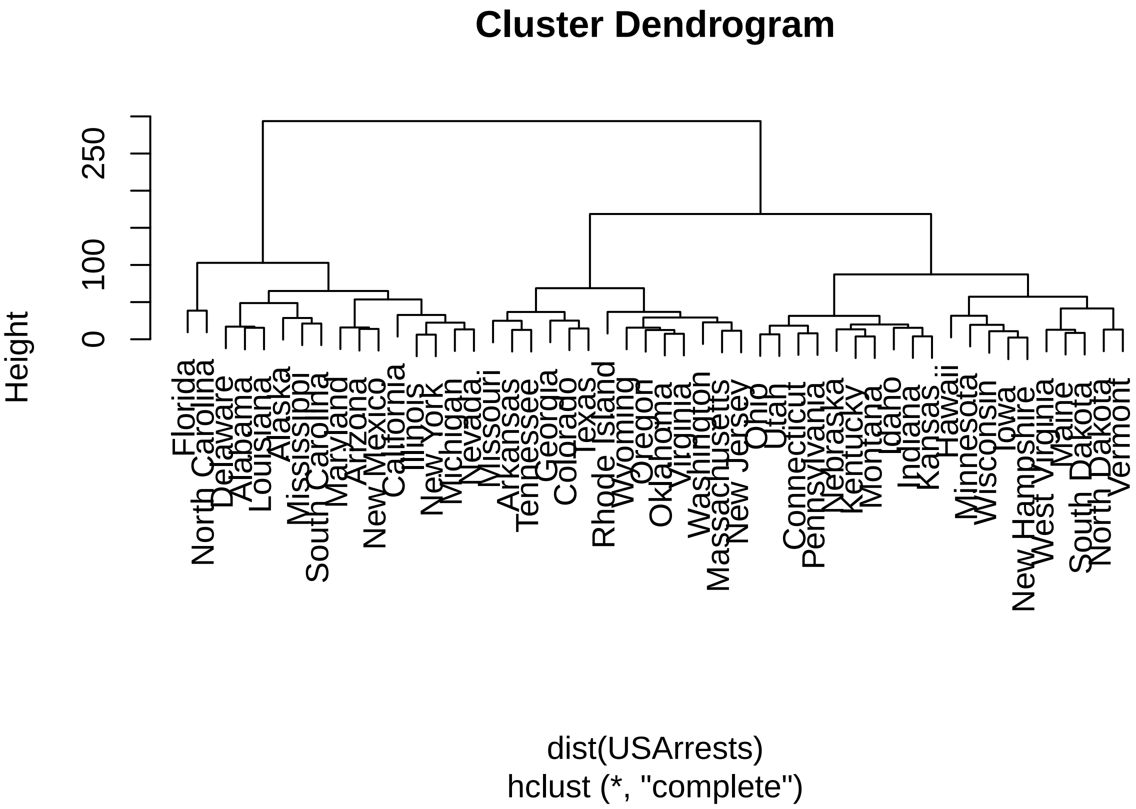
Problem 3

a)

```
hclust.complete=hclust(dist(USArrests),method='complete')
plot(hclust.complete)
```



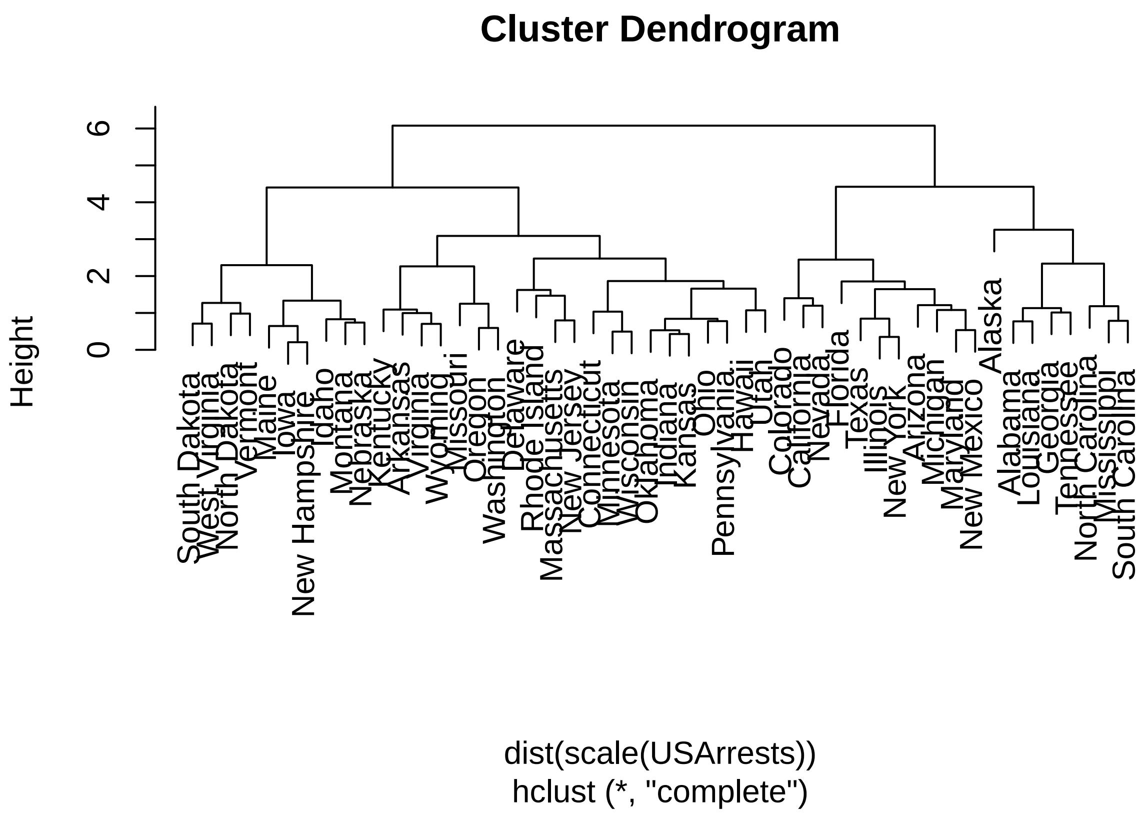
b)

```
re1=cutree(hclust.complete,3)
re1
```

```
##        Alabama         Alaska        Arizona       Arkansas     California
##              1              1              1              2              1
##       Colorado    Connecticut       Delaware        Florida        Georgia
##              2              3              1              1              2
##         Hawaii          Idaho       Illinois        Indiana           Iowa
##              3              3              1              3              3
##         Kansas       Kentucky      Louisiana          Maine       Maryland
##              3              3              1              3              1
##  Massachusetts       Michigan      Minnesota    Mississippi       Missouri
##              2              1              3              1              2
##        Montana       Nebraska         Nevada  New Hampshire     New Jersey
##              3              3              1              3              2
##     New Mexico       New York North Carolina   North Dakota           Ohio
##              1              1              1              3              3
##       Oklahoma         Oregon   Pennsylvania   Rhode Island South Carolina
##              2              2              3              2              1
##   South Dakota      Tennessee          Texas           Utah        Vermont
##              3              2              2              3              3
##       Virginia     Washington  West Virginia      Wisconsin        Wyoming
##              2              2              3              3              2
```

c)

```
hclust.complete.sc=hclust(dist(scale(USArrests)),method='complete')
plot(hclust.complete.sc)
```



```
re2=cutree(hclust.complete.sc,3)
re2
```

```
##        Alabama         Alaska        Arizona       Arkansas     California
##              1              1              2              3              2
##       Colorado    Connecticut       Delaware        Florida        Georgia
##              2              3              3              2              1
##         Hawaii          Idaho       Illinois        Indiana           Iowa
##              3              3              2              3              3
##         Kansas       Kentucky      Louisiana          Maine       Maryland
##              3              3              1              3              2
##  Massachusetts       Michigan      Minnesota    Mississippi       Missouri
##              3              2              3              1              3
##        Montana       Nebraska         Nevada  New Hampshire     New Jersey
##              3              3              2              3              3
##     New Mexico       New York North Carolina   North Dakota           Ohio
##              2              2              1              3              3
##       Oklahoma         Oregon   Pennsylvania   Rhode Island South Carolina
##              3              3              3              3              1
##   South Dakota      Tennessee          Texas           Utah        Vermont
##              3              1              2              3              3
##       Virginia     Washington  West Virginia      Wisconsin        Wyoming
##              3              3              3              3              3
```

d)

```
table(re1,re2)
```

```
##    re2
## re1  1  2  3
##   1  6  9  1
##   2  2  2 10
##   3  0  0 20
```

Scaling the variables indeed has effect on the output clusters. But the trees are still similar. We should scale the varibles in order to unify the data's measure.

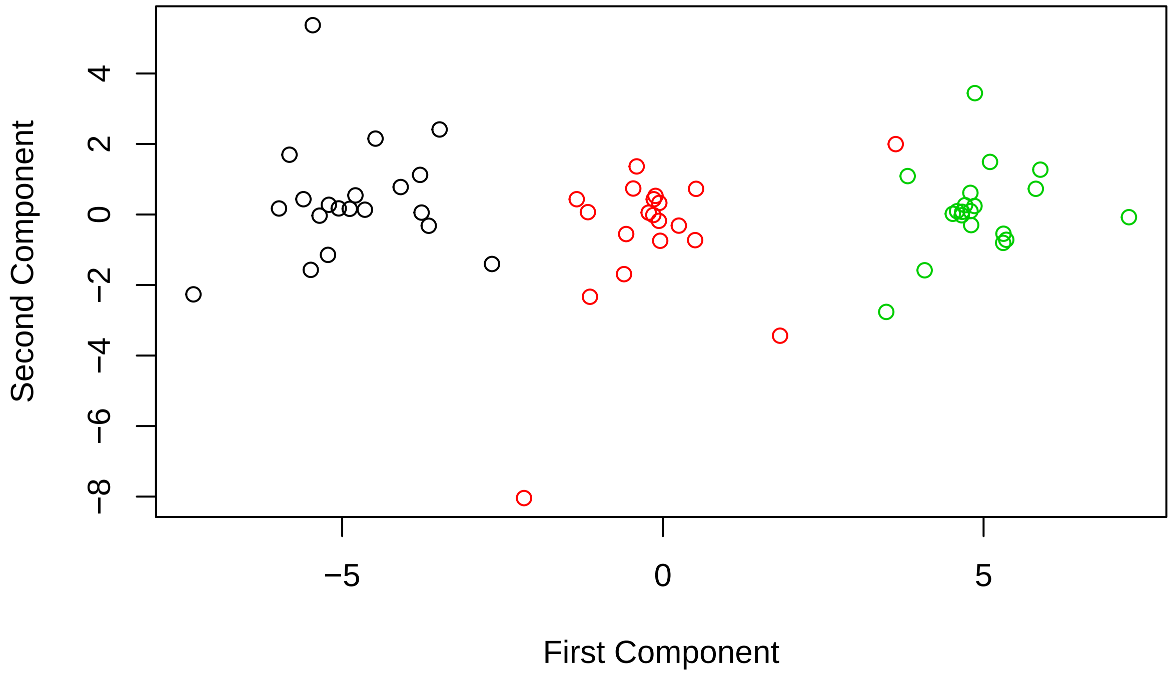
Problem 4

a)

```
set.seed(2)
data=matrix(0,ncol=50,nrow=60)
for(i in 1:20)
{
  data[i,]=rnorm(50,mean=1,sd=i/10)
}
for(i in 21:40)
{
  data[i,]=rnorm(50,mean=2,sd=(i-20)/10)
}
for(i in 41:60)
{
  data[i,]=rnorm(50,mean=3,sd=(i-40)/10 )
}
```

b)

```
pca=prcomp(data,scale=T)
first_comp=(pca$x)[,1]
second_comp=(pca$x)[,2]
plot(first_comp,second_comp,col=c(rep(1,20),rep(2,20),rep(3,20)),xlab='First Component',ylab='Second Co
```



c)

```r
set.seed(2)
km=kmeans(data,3,nstart=30)
vec_true_label=c(rep(1,20),rep(2,20),rep(3,20))
table(km$cluster,as.factor(vec_true_label) )
```

```
##
##      1  2  3
##   1 19  0  0
##   2  0  1 20
##   3  1 19  0
```

K-means is doing a really nice job in clustering the observations with only two wrong label.

d)

```r
set.seed(2)
km=kmeans(data,2, nstart=30 )
vec_true_label=c(rep(1,20),rep(2,20),rep(3,20))
table(km$cluster,as.factor(vec_true_label) )
```

```
##
##      1  2  3
##   1  0  2 20
##   2 20 18  0
```

K-means successfully seperates the true class 3 from the others while failing to seperate class 2 and 3 . That is it forms a cluster that consists of all 20 observations from class 3 and 2 observation from class 2. And all 20 observations in class 3 and 18 observations in class 2 got clustered together.

e)

```
set.seed(2)
km=kmeans(data,4, nstart=30 )
vec_true_label=c(rep(1,20),rep(2,20),rep(3,20))
table(km$cluster,as.factor(vec_true_label) )
```

```
##
##      1  2  3
##   1  1 19  0
##   2  0  1 19
##   3  0  0  1
##   4 19  0  0
```

K-means almost successfully seperates the 3 true class. And it also constructs a cluster with only 1 observation in it.

f)

```
set.seed(2)
km=kmeans(cbind(first_comp,second_comp),3, nstart=30 )
vec_true_label=c(rep(1,20),rep(2,20),rep(3,20))
table(km$cluster,as.factor(vec_true_label) )
```

```
##
##      1  2  3
##   1 19  0  0
##   2  0  1 20
##   3  1 19  0
```

Even with only 2 principle components, K-means is doing a really nice job in clustering the observations with only two wrong label. This shows that the first 2 principle components capture most of the information in the raw data.

g)

```
set.seed(2)
km=kmeans(scale(data),3,nstart=30)
vec_true_label=c(rep(1,20),rep(2,20),rep(3,20))
table(km$cluster,as.factor(vec_true_label) )
```

```
##
##      1  2  3
##   1 20  0  0
##   2  0  1 20
##   3  0 19  0
```

The result is slightly better with only 1 wrong label. This is because scaling the data gives each variable equally impact on the output of the model. This enhances the model's robustness against rare randomly-generated outliers in the observations.