Load ncidata100genes.txt.

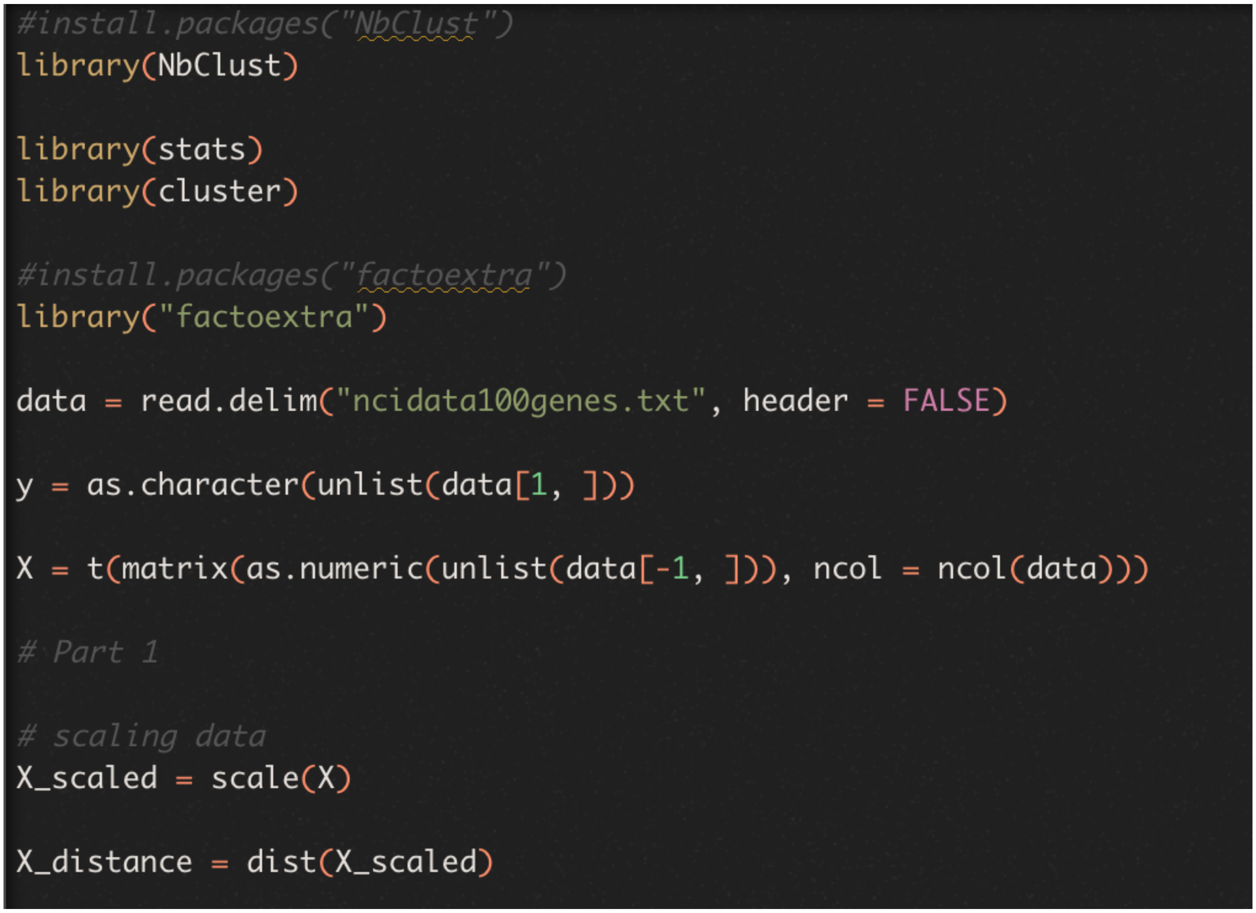
Data contains gene expression of 100 genes from 64 cell lines (samples) from diverse cancer types.

The first row indicates cell lines.

Starting from the second row, each row contains gene expression values for each gene.

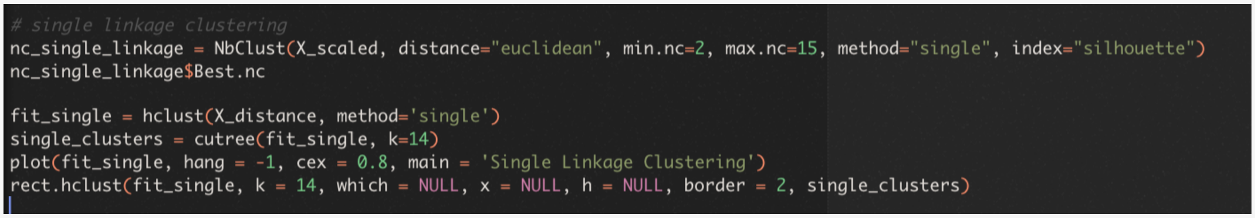
Each column indicates one sample.

Omit the first row, obtain the sample clusters from gene expression values using the cluster methods as below:

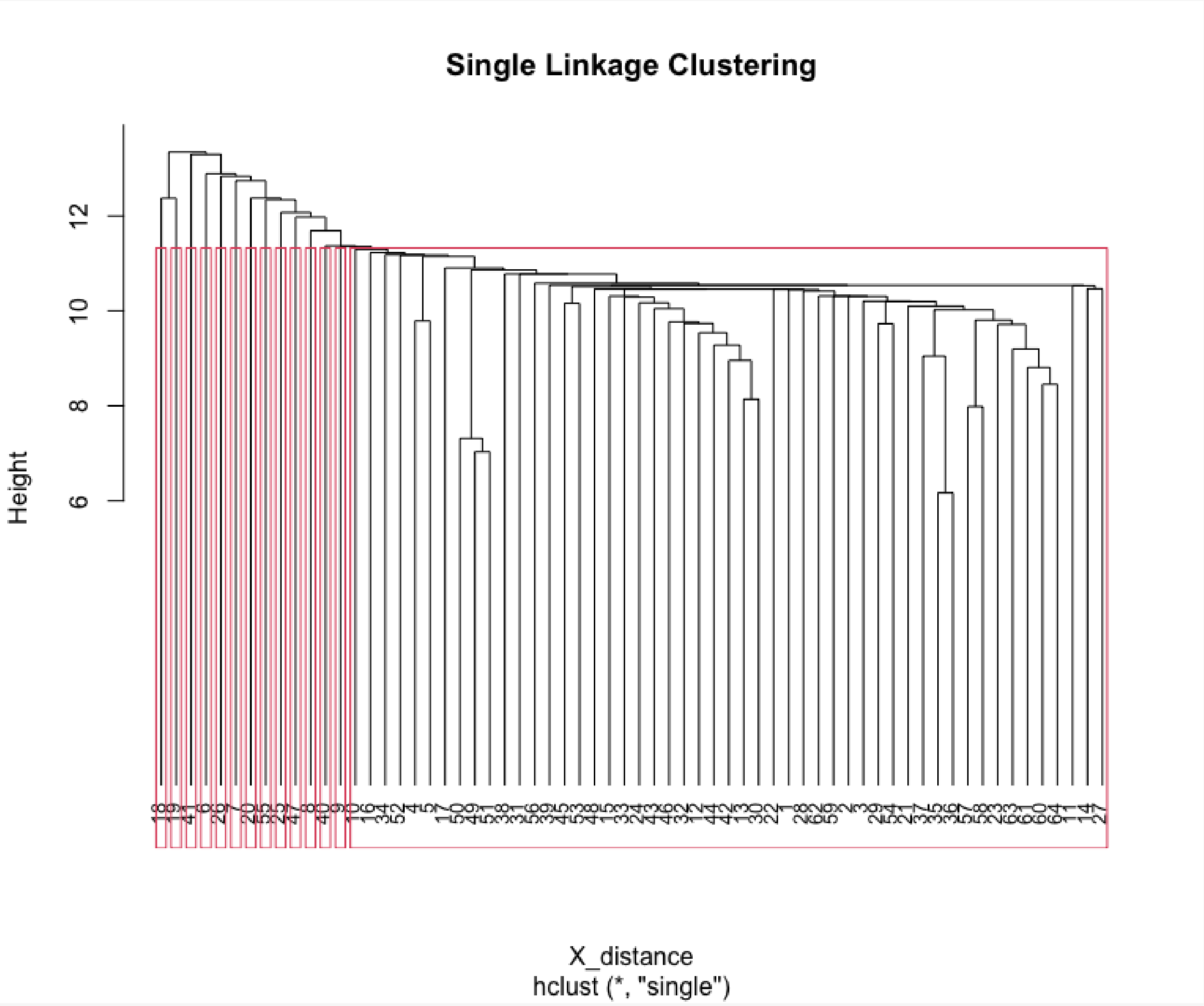


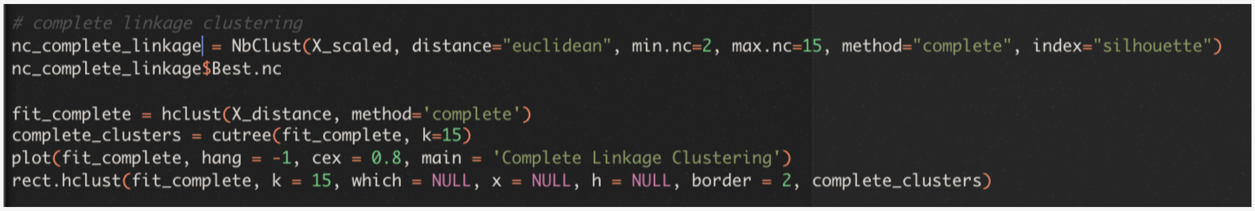
```
#install.packages("NbClust")
library(NbClust)

library(stats)
library(cluster)

#install.packages("factoextra")
library("factoextra")

data = read.delim("ncidata100genes.txt", header = FALSE)

y = as.character(unlist(data[1, ]))

X = t(matrix(as.numeric(unlist(data[-1, ])), ncol = ncol(data)))

# Part 1

# scaling data
X_scaled = scale(X)

X_distance = dist(X_scaled)
```

1. Perform Hierarchical clustering on this data set with linkage methods: Single linkage, Complete linkage, Average linkage, Centroid, Ward.

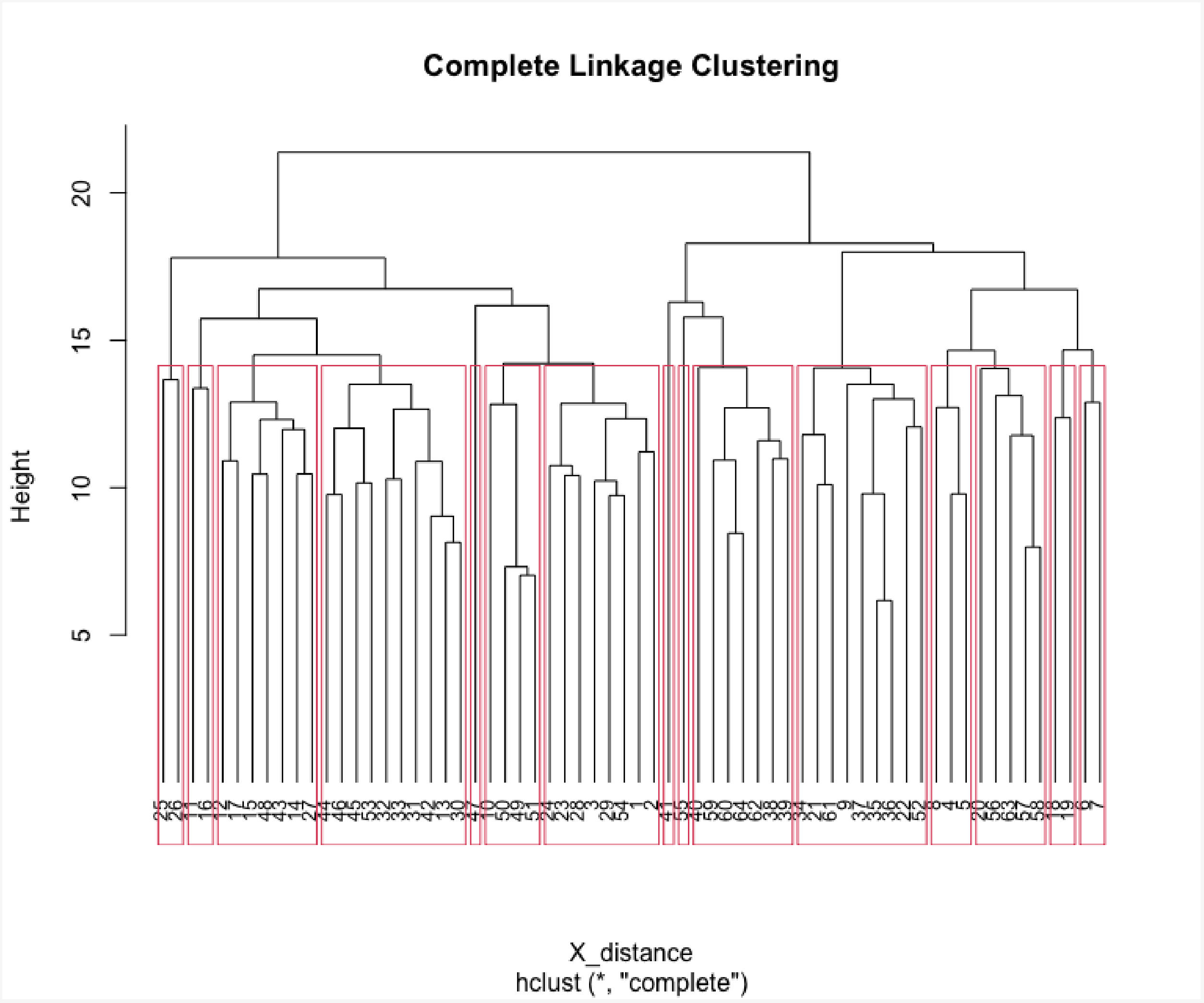


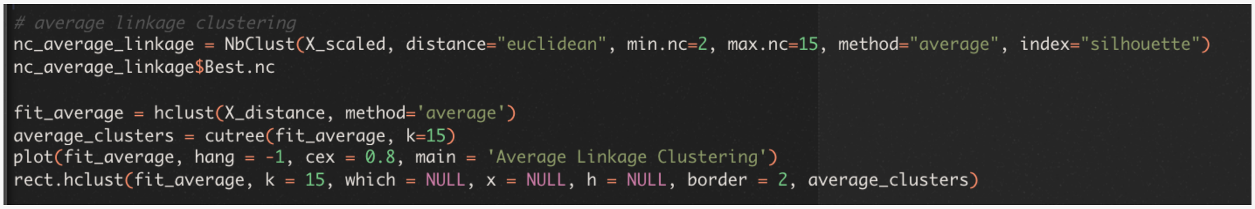
```
# single linkage clustering
nc_single_linkage = NbClust(X_scaled, distance="euclidean", min.nc=2, max.nc=15, method="single", index="silhouette")
nc_single_linkage$Best.nc

fit_single = hclust(X_distance, method='single')
single_clusters = cutree(fit_single, k=14)
plot(fit_single, hang = -1, cex = 0.8, main = 'Single Linkage Clustering')
rect.hclust(fit_single, k = 14, which = NULL, x = NULL, h = NULL, border = 2, single_clusters)
```



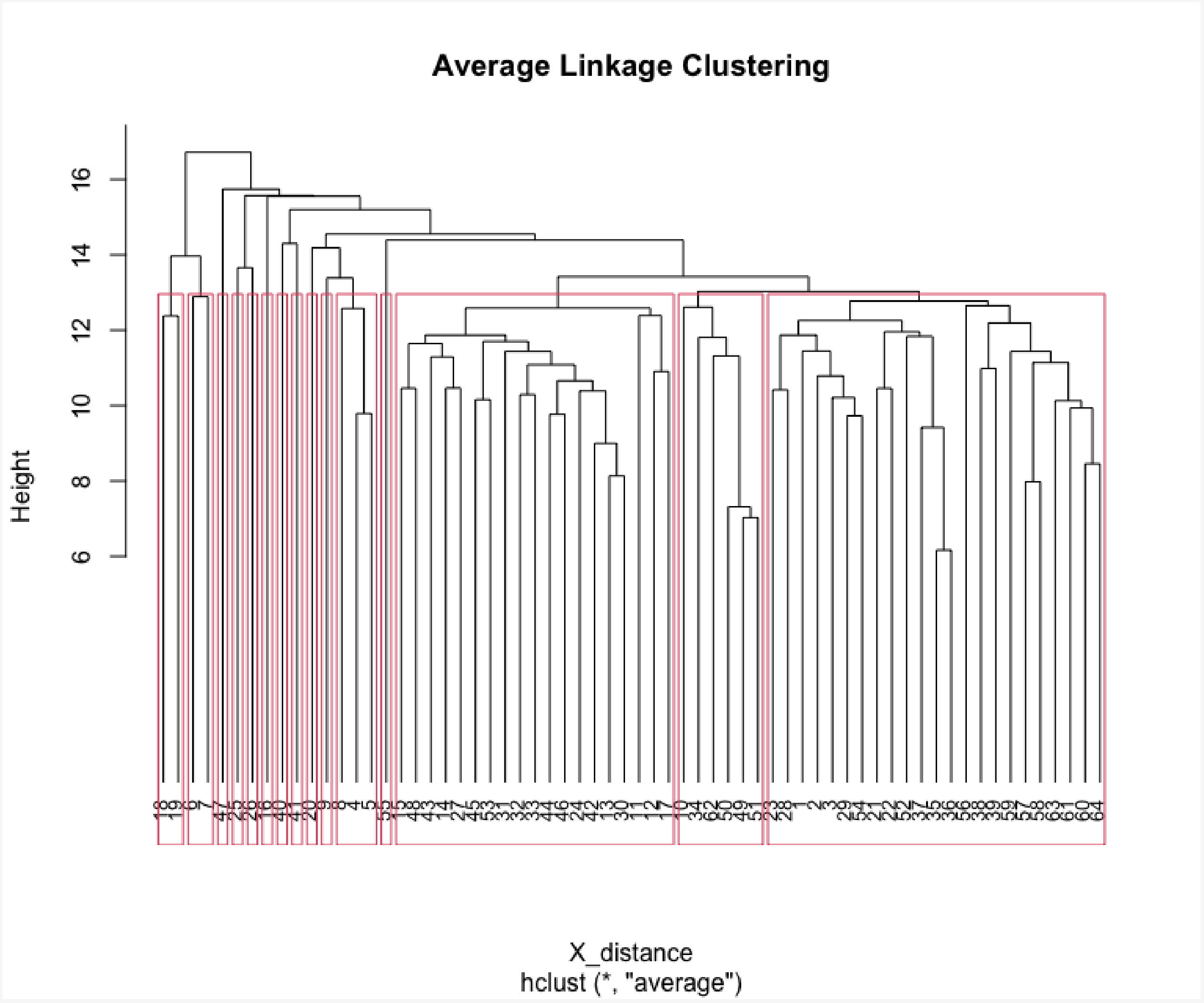


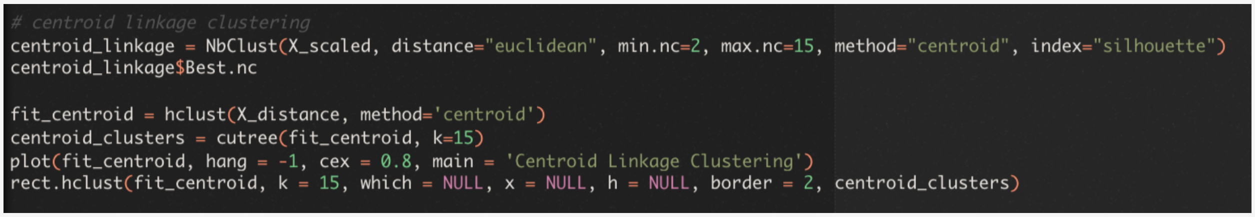
```
# complete linkage clustering
nc_complete_linkage = NbClust(X_scaled, distance="euclidean", min.nc=2, max.nc=15, method="complete", index="silhouette")
nc_complete_linkage$Best.nc

fit_complete = hclust(X_distance, method='complete')
complete_clusters = cutree(fit_complete, k=15)
plot(fit_complete, hang = -1, cex = 0.8, main = 'Complete Linkage Clustering')
rect.hclust(fit_complete, k = 15, which = NULL, x = NULL, h = NULL, border = 2, complete_clusters)
```



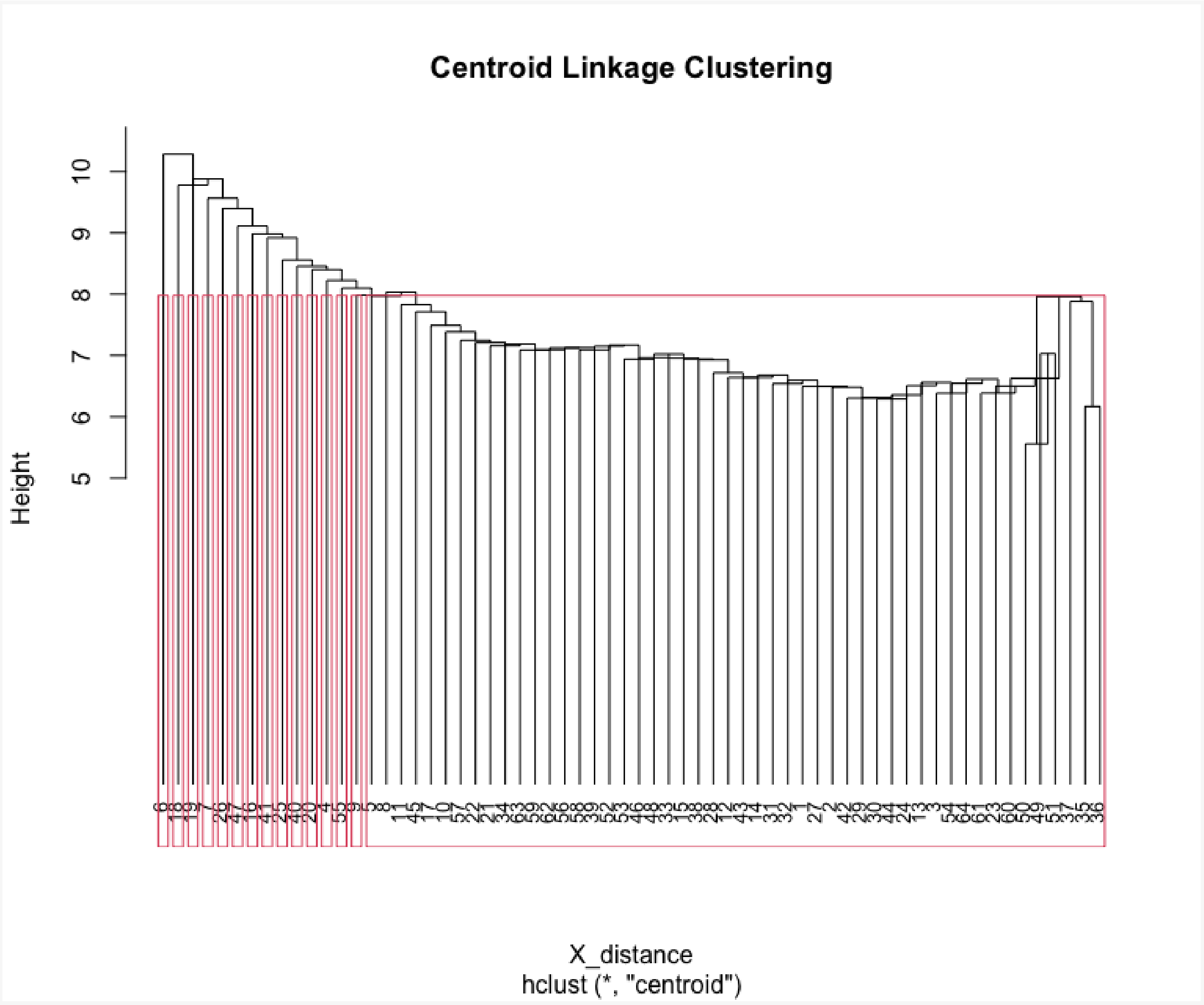


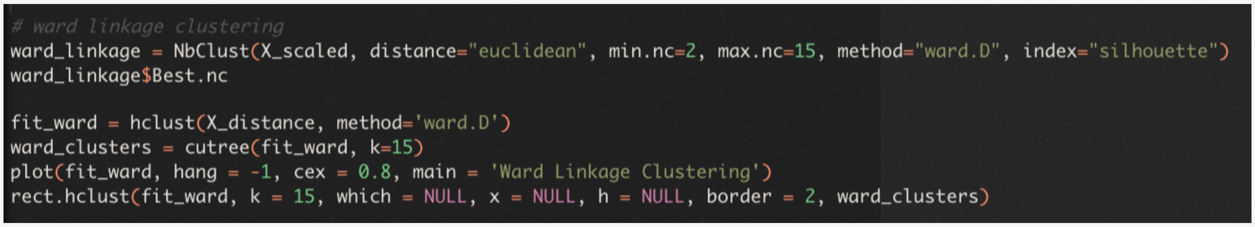
```
# average linkage clustering
nc_average_linkage = NbClust(X_scaled, distance="euclidean", min.nc=2, max.nc=15, method="average", index="silhouette")
nc_average_linkage$Best.nc

fit_average = hclust(X_distance, method='average')
average_clusters = cutree(fit_average, k=15)
plot(fit_average, hang = -1, cex = 0.8, main = 'Average Linkage Clustering')
rect.hclust(fit_average, k = 15, which = NULL, x = NULL, h = NULL, border = 2, average_clusters)
```



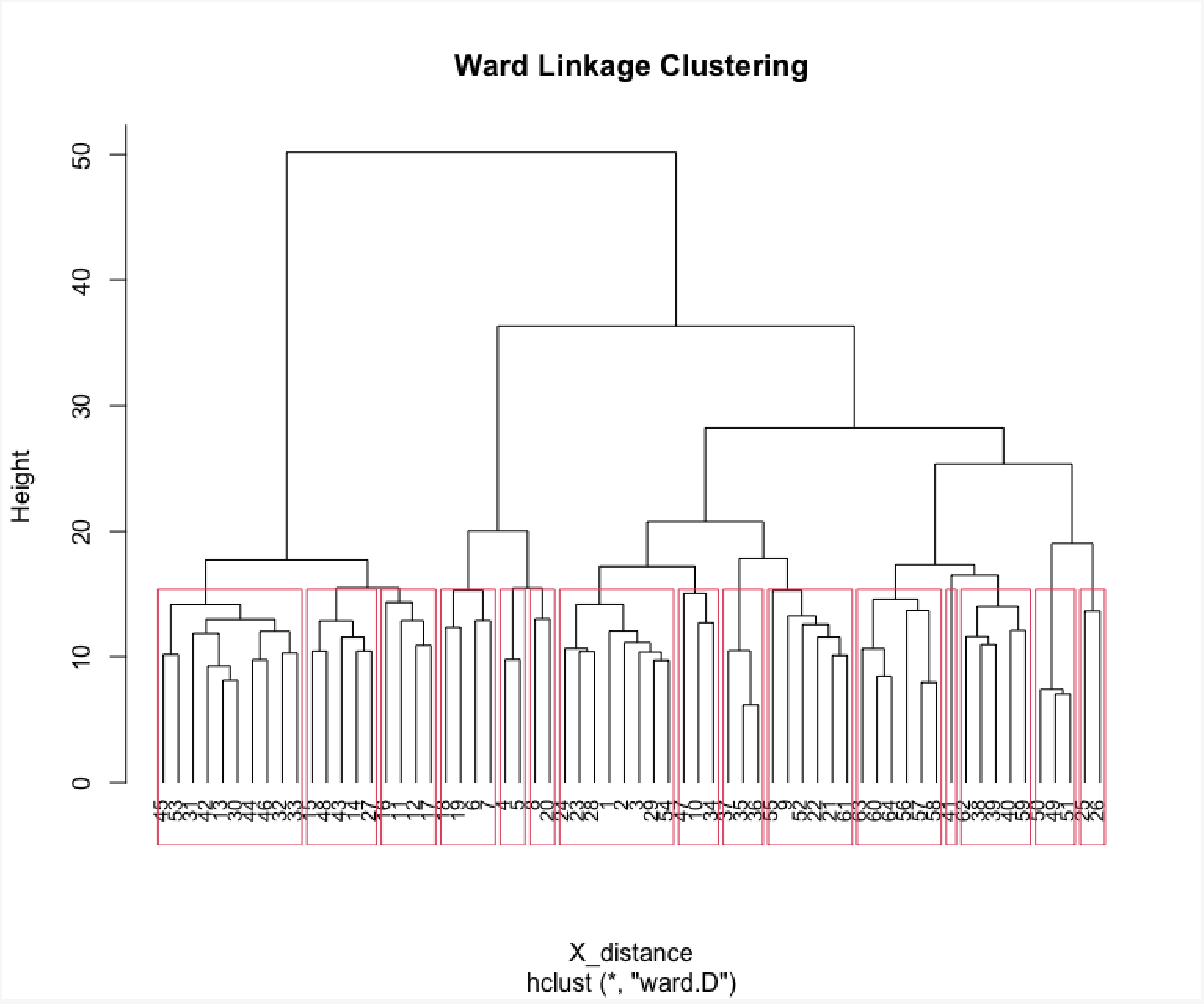


```
# centroid linkage clustering
centroid_linkage = NbClust(X_scaled, distance="euclidean", min.nc=2, max.nc=15, method="centroid", index="silhouette")
centroid_linkage$Best.nc

fit_centroid = hclust(X_distance, method='centroid')
centroid_clusters = cutree(fit_centroid, k=15)
plot(fit_centroid, hang = -1, cex = 0.8, main = 'Centroid Linkage Clustering')
rect.hclust(fit_centroid, k = 15, which = NULL, x = NULL, h = NULL, border = 2, centroid_clusters)
```

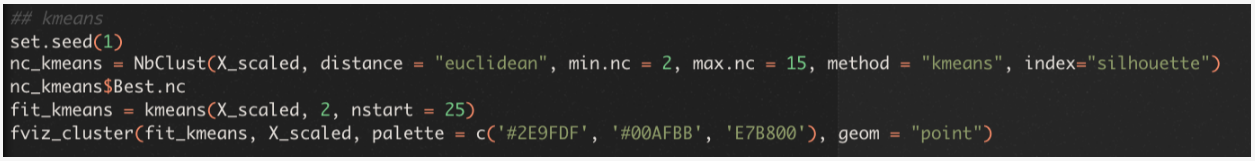


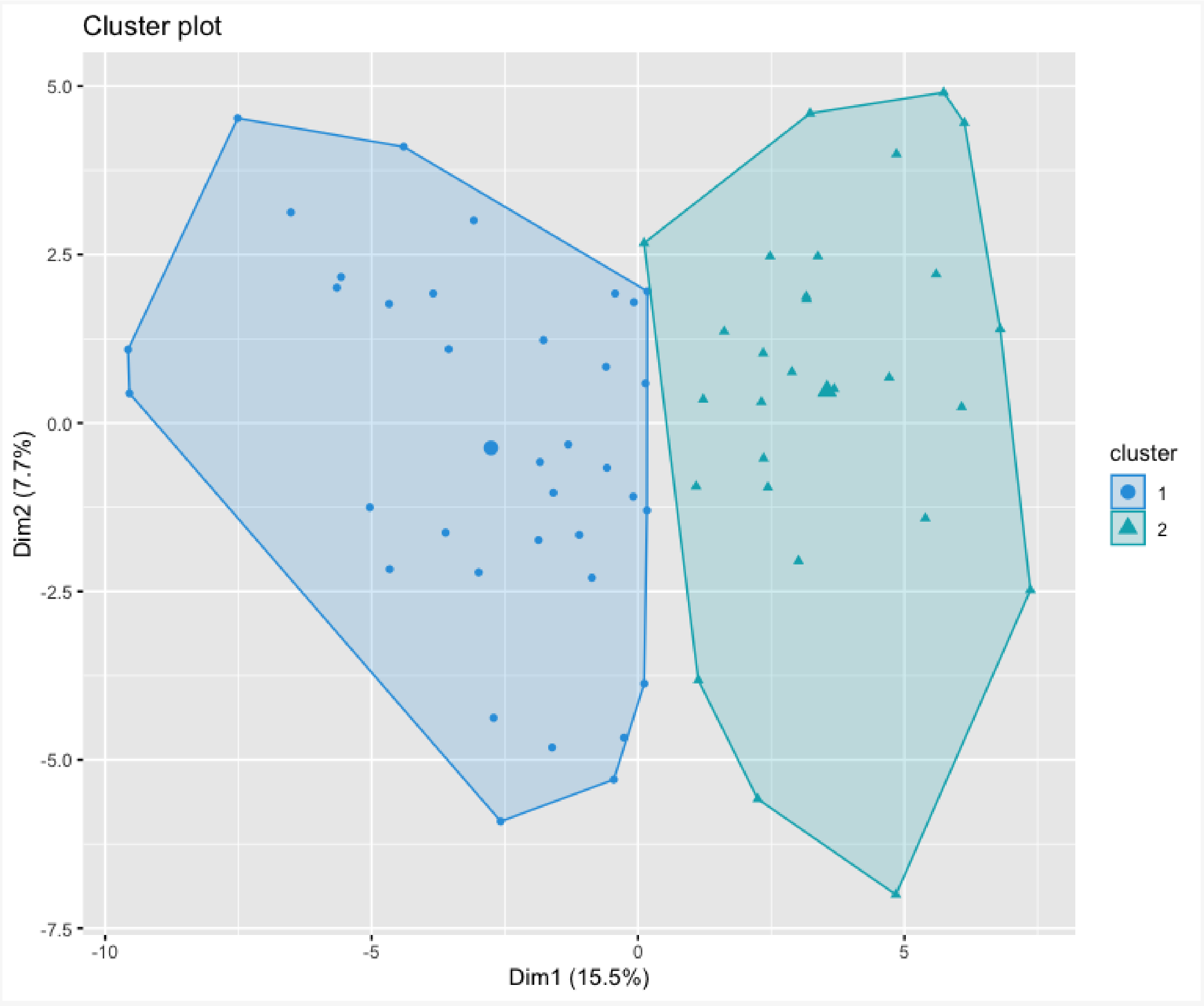


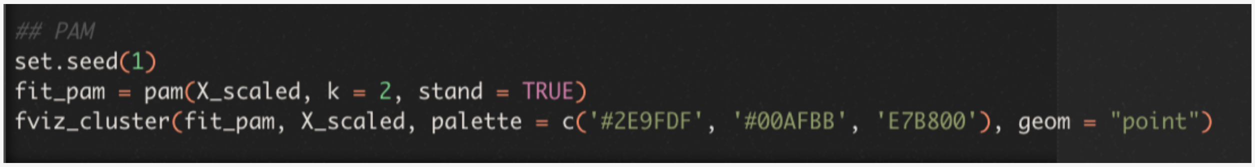
```
# ward linkage clustering
ward_linkage = NbClust(X_scaled, distance="euclidean", min.nc=2, max.nc=15, method="ward.D", index="silhouette")
ward_linkage$Best.nc

fit_ward = hclust(X_distance, method='ward.D')
ward_clusters = cutree(fit_ward, k=15)
plot(fit_ward, hang = -1, cex = 0.8, main = 'Ward Linkage Clustering')
rect.hclust(fit_ward, k = 15, which = NULL, x = NULL, h = NULL, border = 2, ward_clusters)
```



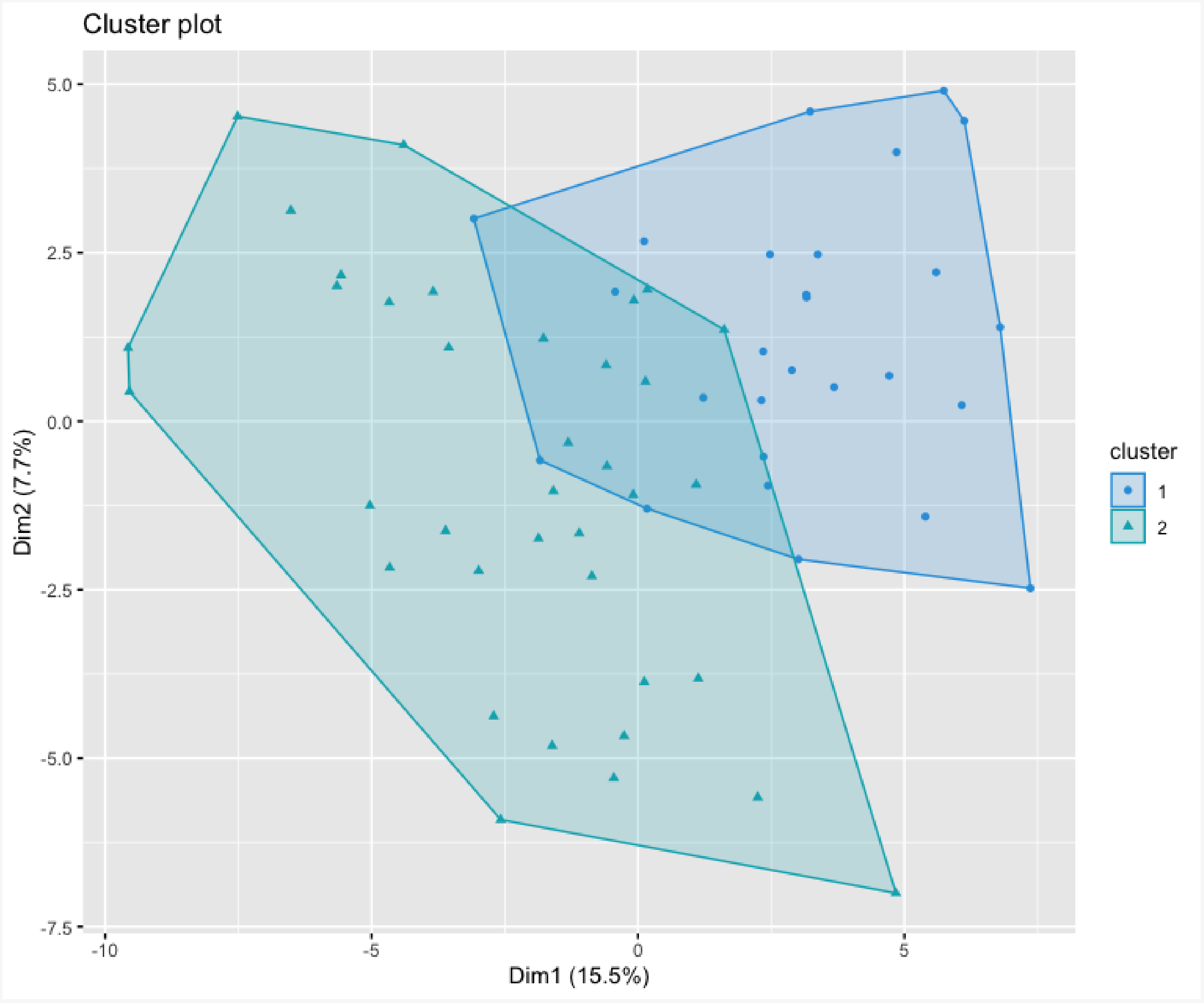
2. Perform k-means and partitioning around medoids (PAM) cluster methods on this data set.


```
## kmeans
set.seed(1)
nc_kmeans = NbClust(X_scaled, distance = "euclidean", min.nc = 2, max.nc = 15, method = "kmeans", index="silhouette")
nc_kmeans$Best.nc
fit_kmeans = kmeans(X_scaled, 2, nstart = 25)
fviz_cluster(fit_kmeans, X_scaled, palette = c('#2E9FDF', '#00AFBB', 'E7B800'), geom = "point")
```





```
## PAM
set.seed(1)
fit_pam = pam(X_scaled, k = 2, stand = TRUE)
fviz_cluster(fit_pam, X_scaled, palette = c('#2E9FDF', '#00AFBB', 'E7B800'), geom = "point")
```


Cluster plot
5.0
2.5
0.0
-2.5
-5.0
-7.5
-10
-5
0
5
Dim1 (15.5%)
Dim2 (7.7%)
cluster
1
2