```
# load library
library(fGarch)
library(boot)
library(MASS)
library(fpc)

### insight: create data which a clustering is clearly visible
##  subset 1: subset has normal distribution with given mean and
covariance matrix
set.seed(4)
D1 <-mvrnorm(n=150, mu=c(1,0,1), Sigma=matrix(rep(0.15, 9),3,3),
             empirical = T)%>%data.frame()

##  subset 2: subset has normal distribution with given mean and
covariance matrix
set.seed(4)
D2 <- mvrnorm(n=175, mu=c(4,3,3),
               Sigma=matrix(data = c(0.5, 0.35, 0.35, 0.35,
                                     0.5, 0.35, 0.35, 0.35, 0.5),
                            byrow=T, nrow=3),
              empirical = T) %>% data.frame()

##  subset 3: subset has normal distribution with given mean and
covariance matrix
set.seed(4)
D3 <- mvrnorm(n=175, mu=c(-3,-3,-2),
              Sigma=matrix(data = c(0.6, 0.5, 0.5, 0.5,
                                    0.6, 0.5, 0.5, 0.5, 0.6),
                           byrow=T, nrow=3),
              empirical = T) %>% data.frame()

## subset 4 (outliers)
set.seed(4)
D4 <- matrix(data = c(runif(n=10, min=-3, max = 1),
```

```
                        runif(n=10, min=-3, max = 1),
                        runif(n=10, min=-3, max = 1)),
              byrow=F, ncol =3)%>% data.frame()

## subset 5 (outliers)
set.seed(4)
D5 <- matrix(data = c(runif(n=10, min=-1, max = 4),
                        runif(n=10, min=-1, max = 4),
                        runif(n=10, min=-1, max = 4)),
              byrow=F, ncol =3)%>% data.frame()

# summary of data generated
total_data <- rbind(D1, D2, D3, D4, D5)
total_data <-scale(total_data)%>%data.frame()

# using MDS to visualize data total_data_dist <-
dist(total_data) total_data_cmd <-
cmdscale(total_data_dist) rownames(total_data_cmd) <-
c(rep(1,150), rep(2, 175),
                                rep(3,175), rep(4,10), rep(5,10))
plot(total_data_cmd, type='n')
text(total_data_cmd, rownames(total_data_cmd))

# using the function for k-mean clustering method
boot_kmean_3 <-clusterboot(data=total_data,bscompare=T,
                            multipleboot=F,bootmethod="boot",
                            B=100, clustermethod=kmeansCBI,
                            count=T, showplots=T, krange=3, seed = 4)

# using the function for hierarchical clustering with single linkage
boot_hclus_3_single <- clusterboot(data=total_data,bscompare=T,
                                    B=100, multipleboot=F,
                                    method="single", bootmethod="boot",
                                    clustermethod=hclustCBI, count=T,
                                    showplots=T,k=3,seed = 4,cut="number")

# using the function for hierarchical clustering with complete linkage
boot_hclus_3_comple <- clusterboot(data=total_data,bscompare=T,
                                    B=100, multipleboot=F,
```

```
                                        method="complete", bootmethod="boot",
                                        clustermethod=hclustCBI, count=T,
                                        showplots=T,k=3,seed = 4,cut="number")

# using the function for hierarchical clustering with average linkage
boot_hclus_3_average <- clusterboot(data=total_data,bscompare=T,
                                        B=100, multipleboot=F,
                                        method="average", bootmethod="boot",
                                        clustermethod=hclustCBI, count=T,
                                        showplots=T,k=3,seed = 4,cut="number")

# using the function for hierarchical clustering with centroid linkage
boot_hclus_3_centroid <- clusterboot(data=total_data,bscompare=T,
                                        B=100, multipleboot=F,
                                        method="centroid", bootmethod="boot",
                                        clustermethod=hclustCBI, count=T,
                                        showplots=T,k=3,seed =4,cut="number")

# compare clusters found by algorithm and real clusters
real_cluster <- factor(c(rep(1,150), rep(2, 175), rep(3,175)))

## K-means
set.seed(4)
k_mean <- kmeans(total_data,3)$cluster[1:500]%>%as.factor()

mean(real_cluster==k_mean)

## hierarchical clustering with average linkage
hc_average = hclust(dist(total_data), method = "average")

hc_average_ind <- cutree(hc_average, k=3)[1:500]

hc_average_ind

## visualize clusters
# visualize data for k=3
k_mean_3 <- kmeans(total_data,3)
fviz_cluster(k_mean_3, geom = "point",
             data = total_data) + ggtitle("k = 3")
clusplot(total_data, k_mean_3$cluster,
         main='2D representation of the Cluster solution',

         color=TRUE, shade=TRUE,
```

```
          labels=3, lines=0)

# visualize data for hierarchical clustering
hc_single = hclust(dist(total_data), method = "single")
hc_complete = hclust(dist(total_data), method = "complete")
hc_average = hclust(dist(total_data), method = "average")
hc_centroid = hclust(dist(total_data), method = "centroid")

plot(hc_single, main = "Single Linkage", xlab = "",sub = "",cex =.9)
plot(hc_complete, main = "Complete Linkage", xlab = "",sub = "",cex =.9)
plot(hc_average, main = "Average Linkage", xlab = "",sub = "",cex =.9)
plot(hc_centroid, main = "Centroid Linkage", xlab = "",sub = "",cex =.9)
```