# Assignment 21

# Load the libraries

library(arules)

library(arulesViz)

# Load the data set

data(Groceries)

summary(Groceries)

inspect(Groceries[1:3])

cbind(itemFrequency(Groceries[,1:10])\*100)

itemFrequencyPlot(Groceries, support=0.01, main="Relative ItemFreq Plot",

type="absolute")

itemFrequencyPlot(Groceries,topN=50,type="relative",main="Relative Freq Plot")

# Get the association rules based on apriori algo

rules <- apriori(Groceries, parameter = list(supp = 0.01, conf = 0.10))

summary(rules)

inspect(rules[1:8])

rules <- apriori(Groceries, parameter = list(supp = 0.01, conf = 0.10, minlen=2))

summary(rules)

support<-seq(0.01,0.1,0.01)

support

rules\_count<-c(435,128,46,26,14, 10, 10,8,8,8)

rules\_count

plot(support,rules\_count,type = "l",main="Number of rules at different support %",

col="darkred",lwd=3)

conf<-seq(0.10,1.0,0.10)

conf

rules\_count<-c(427,231,125,62,15,0,0,0,0,0)

rules\_count

plot(conf,rules\_count,type = "l",main="Number of rules at different confidence %",

col="darkred",lwd=3)

rules\_ec <- eclat(Groceries, parameter = list(supp = 0.05))

summary(rules\_ec)

#sorting out the most relevant rules

rules<-sort(rules, by="confidence", decreasing=TRUE)

inspect(rules[1:5])

rules<-sort(rules, by="lift", decreasing=TRUE)

inspect(rules[1:5])

########################################

library(factoextra)

library("factoextra")

data(decathlon2)

decathlon2.active <- decathlon2[1:23, 1:10]

head(decathlon2.active[, 1:6])

#Compute PCA in R using prcomp()

library(factoextra)

res.pca <- prcomp(decathlon2.active, scale = TRUE)

res.pca

summary(res.pca)

fviz\_eig(res.pca)

fviz\_pca\_ind(res.pca,

col.ind = "cos2", # Color by the quality of representation

gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),

repel = TRUE # Avoid text overlapping

)

fviz\_pca\_var(res.pca,

col.var = "contrib", # Color by contributions to the PC

gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),

repel = TRUE # Avoid text overlapping

)

fviz\_pca\_biplot(res.pca, repel = TRUE,

col.var = "#2E9FDF", # Variables color

col.ind = "#696969" # Individuals color

)

library(factoextra)

# Eigenvalues

eig.val <- get\_eigenvalue(res.pca)

eig.val

# Results for Variables

res.var <- get\_pca\_var(res.pca)

res.var$coord # Coordinates

res.var$contrib # Contributions to the PCs

res.var$cos2 # Quality of representation