Clustering Exercises 0 - 6

# Exercise 0: Install these packages if you don't have them already

# install.packages(c("cluster", "rattle.data","NbClust"))

library(cluster)

library(NbClust)

# Exercise 1: Remove the first column from the data and scale

# it using the scale() function

df <- scale(wine[-1])

head (df)

# Exercise 2:

# Method 1: A plot of the total within-groups sums of squares against the

# number of clusters in a K-means solution can be helpful. A bend in the

# graph can suggest the appropriate number of clusters.

wssplot <- function(data, nc=15, seed=1234){

wss <- (nrow(data)-1)\*sum(apply(data,2,var))

for (i in 2:nc){

set.seed(seed)

wss[i] <- sum(kmeans(data, centers=i)$withinss)}

plot(1:nc, wss, type="b", xlab="Number of Clusters",

ylab="Within groups sum of squares")

}

wssplot(df)

# \* How many clusters does this method suggest? 3

# \* Why does this method work? What's the intuition behind it?

# \* Look at the code for wssplot() and figure out how it works - The number of clusters is determined using the wwsplot() function. The data parameter is the numeric dataset to be analyzed, nc is the maximum number of clusters to consider, and seed is a random number seed.

# Exercise 3: How many clusters does this method suggest? 3

# Method 2: Use the NbClust library, which runs many experiments

# and gives a distribution of potential number of clusters.

wssplot(df)

library(NbClust)

set.seed(1234)

nc <- NbClust(df, min.nc=2, max.nc=15, method="kmeans")

barplot(table(nc$Best.n[1,]),

xlab="Numer of Clusters", ylab="Number of Criteria",

main="Number of Clusters Chosen by 26 Criteria")

# Exercise 4: Once you've picked the number of clusters, run k-means

# using this number of clusters. Output the result of calling kmeans()

# into a variable fit.km

fit.km <- kmeans(df, 3, nstart=25)

# Exercise 5: using the table() function, show how the clusters in fit.km$clusters

# compares to the actual wine types in wine$Type. Would you consider this a good

# clustering?

table(wine[-1], by=list(cluster=fit.km$cluster), mean)

# Exercise 6:

# \* Visualize these clusters using function clusplot() from the cluster library

# \* Would you consider this a good clustering?

clusplot(df, fit.km$cluster, main='2D',

color=TRUE, shade=TRUE,

labels=2, lines=0)