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

#### EXAMPLE - HUMAN RESSOURCES ANALYTICS ####

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

# Set your directory to the folder where you have downloaded the HR dataset

# To clean up the memory of your current R session run the following line

rm(list=ls(all=TRUE))

# Let's load our dataset and call it data

data=read.table('8\_DATA\_2.02\_HR.csv',header = T,sep=',') # The function read.table enables us to read flat files such as .csv files

# Now let's have a look at our variables and see some summary statistics

str(data) # The str() function shows the structure of your dataset and details the type of variables that it contains

summary(data) # The summary() function provides for each variable in your dataset the minimum, mean, maximum and quartiles

# Now let's normalize our variables

testdata=data # To keep our dataset safe, let's create a copy of it called "testdata"

testdata = scale(testdata) # the scale function automatically performs data normalization on all your variables

d = dist(testdata, method = "euclidean") # the dist() function computes the distances of all the observations in our dataset

hcward = hclust(d, method="ward.D") # hclust() function performs hiearchical clustering, we pass it the distances, and we set the method argument to "ward.D"

data$groups = cutree(hcward,k=4) # assign our points to our k=4 clusters

aggdata = aggregate(.~ groups, data=data, FUN=mean) # The aggregate() function presents a summary of a statistic, broken down by one or more groups. Here we compute the mean of each variable for each group.

# One thing we would like to have is the proportion of our data that is in each cluster

proptemp=aggregate(S~ groups, data=data, FUN=length) # we create a variable called proptemp which computes the number of observations in each group (using the S variable, but you can take any.)

aggdata$proportion=(proptemp$S)/sum(proptemp$S) # proportion of observations in each group we compute the ratio between proptemp and the total number of observations

aggdata=aggdata[order(aggdata$proportion,decreasing=T),] # Let's order the groups from the larger to the smaller

# Let's see the output by calling our aggdata variable

aggdata

# As discussed in the videos, let's remove the Newborn variable, which is not really relevant and by being a dummy drives the clustering too much...

testdata=data[,1:5] # we create a new dataset, called "testdata" includes all the rows and the 5 first columns of our original dataset

# We then rerun the code used above

testdata = scale(testdata) # We normalize again our original variables

d = dist(testdata, method = "euclidean") # We compute the distances between observations

hcward = hclust(d, method="ward.D") # Hiearchical Clustering using Ward criterion

data$groups = cutree(hcward,k=4) # Create segments for k=4

# Note that we re-use the original dataset "data" (where the variable Newborn is still present) and not "testdata" (where the variable Newborn has been removed)

# Hence we'll be able to produce summary statistics also for the Newborn variable regardless it wasn't included when doing the second version of the clustering

aggdata = aggregate(.~ groups, data=data, FUN=mean) # Aggregate the values again

proptemp=aggregate(S~ groups, data=data, FUN=length) # Compute the number of observations per group

aggdata$proportion=(proptemp$S)/sum(proptemp$S) # Compute the proportion

aggdata=aggdata[order(aggdata$proportion,decreasing=T),] # Let's order the groups from the larger to the smaller

# Let's see the output by calling our aggdata variable

aggdata

# To export the output of our result, we execute the following line using the write.table() function

# write.csv(aggdata, "HR\_example\_Numerical\_Output.csv", row.names=FALSE)

# This allows to import the data in Excel for instance where we can prepare it for the presentation. E.g. change the name of the columns, use colours, etc.

# Instead of write.csv, you can also use write.csv2() if you encounter an error due to regional settings for separators