R CODE

* **Aggregate(**.~ groups, data=data,Fun=mean**)** <-aggdata : creates a summary statistic by groups for all vars; can be used to create averages for each cluster/group
  + proptemp=aggregate(S~ groups, data=data, FUN=length) : here length is counts; an arbitrary variable called S is used to name the count var
  + aggregate(left~ TIC, data=tempdata, FUN=mean) : compute the average attrition rate for each value of TIC
  + cntbTimeRank=aggregate(left~ TIC, data=tempdata, FUN=length) # We compute the number of employees for each value of TIC
* **apply**( data, 2, mean) : applies the function mean function across all columns on data 1=across rows
* **as.numeric**(as.factor(dat$Student\_ID)) <- dat$id : changes a variables name and its type
* **cbind**(dat$weight, dat$hieght) <- dat2 : creates a new data frame by joining variables
* **cor(data[,c(1:5,10)]) :** produces correlations for specified variables in columns 1 – 5, and 10
* **dim**(data) ; gives the number of rows and the number of variables
* **head**(data, n=20) : display first 20 rows of data
* **hclust(**d, method = “ward.D”**)** <- cluster
  + d = **dist**(data, method =”euclidean”): computes distances prior to using hclust
  + **cutree**(cluster, k=3)<- data$groups : determines how many clusters to use after using hclust
* **inspect**(data): gives look at data set up
* **table**(is.na(dat)) : gives counts for each variables and how many na’s
  + **table(data$var);**
* **na.omit**(data) : removes all rows that have a na
* **order:** data = data[order(data$var1,decreasing=T): order rows by var1 in decreasing order
* **predict**(m2, missing=NA, dtest) <- out; takes a model m2 and predicts on a dtest, tells command there are missing values labeled NA
* **read:**
* **read.table**(‘title.csv’, header=TRUE, sep=',') : to read in in a excel spreadsheet in csv form as a data frame
* **read.transactions** **("course\_combos.csv", format ="basket", sep=",")**
* data needs to be in market basket style, no id’s, no headers
* **rm**(list=ls(all=TRUE)) : to clean out all defined variables and results
* **sapply**(data, class) – determine the class of every variable in data
* **scale(**data**) : used to normalize the data**
* **sink: puts plots in folder**
* sink("enc1101 rules")

print(inspect(enc.rules))

sink()

* **str**(data) : gives what type of structure, variables and their values
* **subset**(data, Successful == 0 , select = c(Student\_ID, hs\_gpa)) ; creates a subset of data and filters as well as determines what variables to include
* **summary(data) :** provides min, max, mean and median plus quartiles
* **write.csv**(data, file = "data.csv", row.names = F) ; writes a data frame to a file, gives it a name, tells whether to include row names

**Libraries**

Library(dplyr)

library(Matrix): cbind, rbind, colSums, Matrix, sparseMatrix

library(lattice): to produce advanced plots; xyplot

library(arules): apriori data minning

**Machine Learning Tools**

* **apriori**
* **apriori(MB, parameter = list(support = 0.01, confidence = 0.05)) <- first.rules**
* **plot(first.rules,control = list(jitter=2, col = rev(brewer.pal(9,"Greens")[4:9])), shading = "lift") - plot of rules**
* **enc.rules<- subset(first.rules, subset=rhs %pin% "ENC1101")**
* **inspect(enc.rules)**
* **chustering**
  + **hclust(**d, method = “ward.D”**)** <- cluster
    - d = **dist**(data, method =”euclidean”): computes distances prior to using hclust
    - **cutree**(cluster, k=3)<- data$groups : determines how many clusters to use after using hclust
    - **example module 2: introduction to data science EESEC**
* **regression**
  + **linear regression:** linreg <- lm(y var~., data = data) – produces general linear model for all vars as a function of y var
    - cor(linreg$fitted.values, data$y var) – produces correlation between fitted and actual
    - plot(data$ y var, linreg$fitted.values) – produce plot of actual verses fitted
    - summary(linreg) – to get coef, p value (var stat sig) and t value (var impact)
  + logistic: logreg<- glm(y var ~., family = binomial(logit), data = data)

**Visuals**

* **Stars(**aggdata[,2:(ncol(data))], len = 0.6, key.loc = c(11,6), xlim=c(2,12), main = “Title”, draw.segments = TRUE, nrow = 2, cex =.75, labels=aggdata$groups**) :** radar plot of clustered data
  + **Added prior: palette(rainbow(12, s= 0.6, v = 0.75)**
    - **example module 2: introduction to data science EESEC**
* **xyplot(**yavr~xvar,main = “Title”, type=“p”, group=groups, data=data, …defines groups to color auto.key=list(title=”Group”,space=”left”,cex=1.0,jus=.95), … to create legend par.settings=list(superpose.line=list(pch=0:18,cex=1)), … set display settings col=c(‘blue’,’gree’,’red’)) …sets colors of points
  + **requires library(lattice)**
    - **example module 2: introduction to data science EESEC**
* **plot(**data$column, data$column, main = “Title”, ylab = “Y axis Title”, xlab = “X axis Title**”)**
  + **abline(v=0.2, col =”red’);** h = for horizontal: to add horizontal or vertical lines
  + **text(0.15,9.7, “Title”, col= “red”)** : to add text into plot
* **hist(data$var):** produces a histogram of a variable
* **symbols :** to create a bubble plot
  + cntbTimeRank=aggregate(left~ TIC, data=tempdata, FUN=length)
  + size = cntbTimeRank$left
  + radius = sqrt(size / pi)
  + symbols(x = aggbTimeRank$TIC, y = aggbTimeRank$left, circles = radius, inches = .75, fg = "white", bg = "red, main = "Time and Employee Attrition", ylab = "Average Attrition Rate", xlab = "Time spent")
* **itemFrequencyPlot**(MB, support=.03, cex.names=0.75, xlim = c(0,0.5), type="relative”, horiz=TRUE, col="blue", las=1, xlab = paste("Proportion of Market baskets containing Course", "\n(Item Relative Frequency)")) - bar chats

**misc**

* **validation: accuracy of model**
  + sum((logreg$fitted.values<=cutoff)&(data$left==0))/sum(datatot$left==0) - where cutoff =.3 (can experiment with)
  + sum((logreg$fitted.values>cutoff)&(datatot$left==1))/sum(datatot$left==1)
  + mean((logreg$fitted.values>cutoff)==(datatot$left==1))