Hierarchical Cluster Analysis

Goals: Learn how to perform hierarchical cluster analysis and manipulate with resulting trees

Assignments:

Work with the "iris" data set from R.

1) Describe the data set using R help and online resources.

2) Use hierarchical cluster analysis to identify the examined species (iris setosa, versicolor, and virginica) using clustering of their sepal length and width and petal length and width.
   a. Plot data in different dimensions, using different colors for different species.
   b. Discuss your observations and formulate hypotheses on how the species can be separated.
   c. Perform HCA using different distances and cluster methods.
   d. Select a version of analysis that gives the best species identification.

Extra Points: Define the quality of identification as the proportion of correctly identified plants (out of 150). The best identification quality is worth 5 extra points.

Reports: Printed reports are due on Tuesday, May 9, 2017

Report preparation: Consider each report as a mini-paper. It should NOT be long, but it should provide a reader with all background information about the data, problem, and methods you are using. Review the necessary theoretical material, describe the data. Do not insert the R-output in your report; instead, summarize it in tables or text in a nice readable form. If you feel some parts of the output should be included, put them in Appendix. Put your name on the title page. Illustrations should support your conclusions and make it easier to read a report.
#=================================================#
#                   STAT 755                      #
#          Hierarchical Cluster Analysis          #
#=================================================#

# Install libraries ...
#=======================================
library(MASS)  # ... for Multivariate Normal Distribution
#======================================================
# Synthetic example
#=======================================
# generate points
#-----------------------------------
X = mvrnorm(n=6,c(0,0),diag(c(1,1)))
# plot points
#-----------------------------------------------
par(bg='yellow')
plot(X[,1],X[,2],pch=19,col='blue',
xlab='X',ylab='Y',main='Points in (X,Y) plane')
text(X[,1]+.05,X[,2]+.05)
windows()  # open new window
# Cluster analysis
#=======================================
# try 'euclidean', 'manhatten', 'minkowski',
d<-dist(X,method='minkowski',p=10)
# try 'complete', 'single', 'average', 'median'
h<-hclust(d,method='single')
par(bg='yellow')
plot(h,hang=1,xlab='',sub='',main='')

#=======================================
# Eurodistance example
#=======================================
h<-hclust(eurodist,method='single')
par(bg='yellow')
plot(h,hang=-1,xlab='',ylab='Distance, km',sub='',main='')
i<-identify(h)
i
h$merge
plot(h$h$height)
# Synthetic example: single group
X = mvrnorm(n=100,c(0,0),diag(c(1,1)))
d<-dist(X,method='euclidean')
h<-hclust(d,method='com')
par(bg='yellow')
plot(h, hang=-.01,xlab='',sub='',main='')
c<-cutree(h,k=3)
I1<-c==1
I2<-c==2
plot(X[,1],X[,2],pch=19)
points(X[I1,1],X[I1,2],col='red',pch=19)
points(X[I2,1],X[I2,2],col='blue',pch=19)

# Synthetic example: two groups
X = mvrnorm(n=50,c(0,0),diag(c(1,1)))
Y = mvrnorm(n=50,c(10,10),diag(c(1,1)))
X = rbind(X,Y)
d<-dist(X,method='euclidean')
h<-hclust(d,method='sin')
par(bg='yellow')
plot(h, hang=-.01,xlab='',sub='',main='')
c<-cutree(h,k=2)
I1<-c==1
I2<-c==2
plot(X[,1],X[,2],pch=19)
points(X[I1,1],X[I1,2],col='red',pch=19)
points(X[I2,1],X[I2,2],col='blue',pch=19)

# Stock market example
# (Closing prices on 10 stocks)

# read the data table
T<-read.table('Lab3_close.csv',sep=',',header=TRUE)
len<dim(T)[1] # length
Itime<-seq(len,1,by=-1) # inverse index for plots

time<-1989+(31+28+31+30+4)/365.25+seq(1,len)/250

names(T) # names of variables
P<-T[,seq(2,11)]               # remove the dates
P<-log10(P)

# Remove long-term trend
#========================
PD<-P*0
for (i in seq(1,10))
{
  t<-ts(P[,i],frequency=365)
  s<-stl(t,s.window=250)
  PD[,i]=t-s$time.series[,2]
}

d<-as.dist(cor(PD))
h<-hclust(1-d,method='average')
plot(h,hang=.1,xlab='',sub='',main='Average link')