Generalized variance
Multivariate Normal Distribution

Goals: 1) Illustrate generalized variance properties;
       2) Learn how to generate Multivariate Normal rvs;
       3) Learn how to test for multi-normality;
       4) Learn how to write functions in R.

Assignments:

1. Generate a 1000x2 matrix N of iid standard Normal rvs; denote its columns by X and Y.
2. Create a 1000x3 matrix C with columns
   \[ C_1 = X + Y \]
   \[ C_2 = X - Y \]
   \[ C_3 = 2X + 3Y \]
3. Find the generalized variance of C; discuss.
4. Find the linearly dependent columns in C using the spectral decomposition approach.

5. Generate 1000 multivariate Normal rvs with zero mean and variance-covariance matrix
   \[ \Sigma = \begin{bmatrix} 12 & 4 \\ 4 & 5 \end{bmatrix} \]
6. Find the linear combination that transforms your rv to a standard 2-D Normal rv.

7. Generate 500 multivariate Normal rvs \( X_i \) with zero mean and variance-covariance matrix
   \[ \Sigma = \begin{bmatrix} 10 & 4 & 1 \\ 4 & 5 & 4 \\ 1 & 4 & 10 \end{bmatrix} \]
8. Test multivariate Normality for the sample \( X_i \) using a \( \chi^2 \) test based on statistical distances.
9. Test multivariate Normality for the sample \( (X_i)^T \).

Reports: Printed reports are due on Thursday, March 30, 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 problem and methods you are using. Review the necessary theoretical material and 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 still feel some parts of the R-output should be reported, put them in Appendix. Put your name on the title page.
1. **Generation of Multivariate Normal (MVN) rvs**
   1. Linear combination of iid standard Normal rvs
   2. $\mathbf{R}$-operator

2. **Generalized variance**
   1. Volume occupied by data points
   2. Linear dependence of data with zero generalized variance

3. **Properties of Multivariate Normal distribution**
   1. How to create a MVN rv with given variance matrix from iid standard Normal rvs
   2. How to create iid standard Normal rvs from a MVN rv with given variance matrix
   3. How to test for Multi-normality using the statistical distances

4. **How to write functions in $\mathbf{R}$**
# Install libraries ...
#=================================
library(Matrix)  # ... for matrix operations
library(car)  # ... for ellipse plots
library(stats)  # ... for statistical operations
library(MASS)  # ... for Multivariate Normal Distribution
library(graphics)  # ... for arrows

# Multivariate Normal Sample ...
# ... as a linear combination of iid standard normal rvs
len<5
N<matrix(rnorm(len*2),len,2)  # 5x2 iid N(0,1) rvs
A<matrix(c(1,1,1,-1),2,2)  # 2x2 matrix of coefficients
X<N%*%A  # 5x2 linear combination

# Multivariate Normal Sample ...
# ... using an R operator
#=================================
Sigma <- matrix(c(10,4,4,2),2,2)
mvrnorm(n=1,c(0,0),Sigma)  # sample 1x2 with mean [0,0]
mvrnorm(n=5,c(0,0),Sigma)  # sample 5x2 with mean [0,0]
mvrnorm(n=5,c(-100,100),Sigma)  # sample 5x2 with mean [-100,100]

var(mvrnorm(n=1000, rep(c(0, 2), Sigma))  # Sigma is the population variance
var(mvrnorm(n=1000, rep(c(0, 2), Sigma, empirical = TRUE))  # Sigma is the sample variance

# Correlation and covariance matrices
#-------------------------------------
cor(N)  # correlation matrix
cor(X)  # correlation matrix
cov(N)  # variance-covariance matrix
cov(X)  # variance-covariance matrix
var(N)  # the same as cov(N)
var(X)  # the same as cov(X)
# Generalized variance I: Volume occupied by data
# This example illustrates that generalized variance is related to the volume occupied by data scatter

len<-1000
N<-matrix(rnorm(len*2),len,2)  # 1000x2 iid N(0,1) rvs
A<-matrix(c(2,1,1,2,2,2),2)    # 2x2 matrix of coefficients
X<-N%*%A                      # 1000x2 linear combination
X[,1]=X[,1]+5                 # shift first column
N[,2]=N[,2]+5                 # gen. var for N
det(cov(N))                   # gen. var for X
e1<-SA(X)                     # ellipses for X
e2<-SA(N,add=T)               # ellipses for N

det(cov(X))                   # gen. var for X
eigen(Sigma)                  # eigenvalues, eigenvectors
Sigma<-cov(X)                 # covariance matrix
e plot(X%*%e$vectors[,1],col='blue') # lin. comb. for max. eigenvalue
points(X%*%e$vectors[,3],col='red') # lin. comb. for 0-eigenvalue
e$vectors[,3]/e$vectors[2,3] # "good" form of linear dependence
# Multivariate Normal (MVN) Distribution
# This example shows how to
# a) create Normal rvs with given variance matrix from iid N(0,1)
# b) create iid N(0,1) from Normal rvs with given covariance matrix

Sigma <- matrix(c(10,4,4,2),2,2)  # variance matrix
I <- diag(c(1,1))  # identity matrix
N <- mvrnorm(n=100000,c(0,0),I)  # MVN with variance I
X <- mvrnorm(n=100000,c(0,0),Sigma)  # MVN with variance Sigma

e <- eigen(Sigma)  # spectral decomposition
P <- e$vectors  # eigenvectors
L <- e$values  # eigenvalues

Sm05 <- P%*%sqrt(diag(1/L))%*%t(P)  # inverse square-root matrix
Sp05 <- P%*%sqrt(diag(L))%*%t(P)  # square-root matrix

Z <- t(Sm05%*%t(X))  # vector of iid N(0,1) rvs
X1 <- t(Sp05%*%t(N))  # MVN rv with variance Sigma

var(Z)
var(X1)
Sigma

# Chi-square distribution of statistical distances
# This example shows how to test for multi-normality using the chi-square distribution

Sigma <- matrix(c(10,4,4,2),2,2)  # variance matrix

# (A) True Multivariate Normal
len=1000
X <- mvrnorm(n=len,c(0,0),Sigma)  # 1000x2 MVN rv
S1 <- solve(cov(X))  # inverse of estimated covariance
d <- rep(0,len)
for (i in 1:len)
  d[i] <- t(X[i,]%*%S1%*%X[i,])  # distance from i-th point

qqplot(qchisq(seq(1,len)/len,2),d)  # qqplot with chi-sq quantiles
segments(0,0,10,10,col='red',lwd=2)
grid()

ks.test(d,"pchisq",2)  # formal KS test
# (B) Not Multivariate Normal

len = 1000
X <- mvrnorm(n=len, c(0,0), Sigma)   # 1000x2 MVN rv
X <- X^2

S1 <- solve(cov(X))          # inverse of estimated covariance
d <- rep(0, len)
for (i in 1:len)
  d[i] <- t(X[i,]) %*% S1 %*% X[i,]  # distance from i-th point

qqplot(qchisq(seq(1, len)/len, 2), d)  # qqplot with chi-sq quantiles
segments(0, 0, 10, 10, col='red', lwd=2)
grid()

ks.test(d, "pchisq", 2)  # formal KS test

# Function that illustrates spectral decomposition
# and statistical distance ellipses
#===================================================
# Function that illustrates spectral decomposition
# and statistical distance ellipses
#===================================================

SA <- function(X, add=FALSE, data.plot=TRUE)
{
  # Vector of means
  #===================================================
  n <- dim(X)[1]
ones <- matrix(rep(1,n),ncol=1)
mu <- as.vector(t(X) %*% ones / n)

  # Variance
  #===================================================
  Sigma <- var(X)

  e <- eigen(Sigma)
  par(bg='yellow')
  ellipse(mu, Sigma, 3, add=add, xlim=range(X), ylim=range(X))
  ellipse(mu, Sigma, 2, add=TRUE)
  ellipse(mu, Sigma, 1, add=TRUE)
  if (data.plot)
    points(X[,1], X[,2], pch=20, col=4)
    arrows(mu[1], mu[2], mu[1]+e$vectors[1,1]*sqrt(e$values[1]),
           mu[2]+e$vectors[2,1]*sqrt(e$values[1]), length=.1, col='green', lwd=2)
    arrows(mu[1], mu[2], mu[1]+e$vectors[1,2]*sqrt(e$values[2]),
           mu[2]+e$vectors[2,2]*sqrt(e$values[2]), length=.1, col='green', lwd=2)
    e
}