```
# Script file developed by

# M. Concepción Ausín
# Cristina G. de la Fuente
# Aitor J. Farragut

# This document was created within the framework of a final degree project
# presented by Aitor Juan Farragut in order to obtain the Bachelor's degree
# in Economics at Universidad Carlos III de Madrid.

# The name of the project is
# "Estimación de series temporales financieras multivariantes con modelos de
# cópulas."

# May, 2017

# Clean workspace
rm(list = ls())

# Load the necessary libraries
library(copula)
library(parallel)
library(rugarch)
library(CDVine)

# Estabish directory
# setwd("_____")

# Read data.

v1 = read.csv("____.csv", header = TRUE, sep = ",", dec = ".")
t1 = dim(v1)[1]

v2 = read.csv("____.csv", header = TRUE, sep = ",", dec = ".")
t2 = dim(v2)[1]

v3 = read.csv("____.csv", header = TRUE, sep = ",", dec = ".")
t3 = dim(v3)[1]

v4 = read.csv("____.csv", header = TRUE, sep = ",", dec = ".")
t4 = dim(v4)[1]

# We are going to work with 4 financial time series.
d = 4

# First date in which we observe any data.
f = min(v1[t1, 1], v2[t2, 1], v3[t3, 1], v4[t4, 1])
# Last date in which we observe data in all the series.
F = min(v1[1, 1], v2[1, 1], v3[1, 1], v4[1, 1])

Data = NULL
```

```
# We are only going to keep the adjusted closing prices for the days in which
# we have observations for all d time series.
while(f <= F)
{
	newObs = NULL
	NoV = 0

	if(v1[t1,1] == f)
	{
		newObs = cbind(newObs, v1[t1,7], v2[t2,7])
		NoV = NoV + 2
		v1 = v1[1:(t1-1),]
		t1 = t1 - 1
		v2 = v2[1:(t2-1),]
		t2 = t2 - 1
	}

	if(v3[t3,1] == f)
	{
		newObs = cbind(newObs, v3[t3,7])
		NoV = NoV + 1
		v3 = v3[1:(t3-1),]
		t3 = t3 - 1
	}

	if(v4[t4,1] == f)
	{
		newObs = cbind(newObs, v4[t4,7])
		NoV = NoV + 1
		v4 = v4[1:(t4-1),]
		t4 = t4 - 1
	}

	if(NoV == d)
	{
		Data = rbind(Data, c(f, newObs))
	}

	f = f+1
}

T = dim(Data)[1] - 1

# Diclare the column names according to the variables and the row names accor-
# ding to the date.
col_names = c("NASDAQ", "Dow Jones", "BMV", "IBEX35")
row_names = Data[,1]
row_names = as.Date(as.character(row_names), format = "%Y%m%d")

# In Y, we will have the log-returns of the d assets that we will analyze:
Y = matrix(nrow = T, ncol = d)
```

```
for(t in 1:T)
{
	for(i in 1:d)
	{
		Y[t,i] = 100 * log( Data[(t+1),(i+1)] / Data[t,(i+1)] )
	}
}

Data = data.frame(Y)
names(Data) = col_names
row.names(Data) = row_names[2:(T+1)]
attach(Data)
# Now we have a data frame with the log-returns of all the studied assets.

par(mfrow = c(d, 1))
for (i in 1:d)
{
	plot(row_names[2:(T+1)], Data[,i], "l", col = "blue", ylab = 
col_names[i], xlab = "t")
}

for (i in 1:d)
{
	plot(row_names[2:(T+1)], Data[,i], "l", col = "blue", ylab = 
col_names[i], xlab = "t")
	readline(prompt="Press [enter] to continue")
}

############################################################################
# We will fit the data using a copula-GARCH(1,1) model. We will use vine-
# copulas to characterize the joint behavior of the series.
############################################################################

# We consider a Student's-t distribution for the innovations.

# Constant mean.
meanModel = list(armaOrder = c(0,0), include.mean=TRUE)
# standard GARCH(1,1).
varModel = list(model = "sGARCH", garchOrder = c(1,1))

uspec = ugarchspec(varModel, mean.model = meanModel, distribution.model = "std")

fit = NULL

for(i in 1:d)
{
	fit = cbind(fit, ugarchfit(data = Data[,i], spec = uspec))
}

mu_hat = matrix(nrow = d, ncol = 1)
omega_hat = matrix(nrow = d, ncol = 1)
alpha_hat = matrix(nrow = d, ncol = 1)
```

```
beta_hat = matrix(nrow = d, ncol = 1)
nu_hat = matrix(nrow = d, ncol = 1)

for(i in 1:d)
{
        mu_hat[i] = fit[[i]]@fit$coef[1]
        omega_hat[i] = fit[[i]]@fit$coef[2]
        alpha_hat[i] = fit[[i]]@fit$coef[3]
        beta_hat[i] = fit[[i]]@fit$coef[4]
        nu_hat[i] = fit[[i]]@fit$coef[5]
}

epsilon_hat = sapply(fit, residuals, standardize = TRUE)
U_hat = pobs(epsilon_hat)
pairs(U_hat, labels = c("U_1", "U_2", "U_3", "U_4"), col = "royalblue4")

tau_pairs = cor(U_hat, method = "k")
tau_pairs

# The relation between the NASDAQ and the Dow Jones is clearly very strong
# The relation between the American indices and the Mexican one is surprisingly
# low.
# There is a moderate relation between the American indices and the Spanish one.

# In order to estimate the parameters related to the joint behavior of the
# innovations, we start by estimating the parameters of all the copulas that
# result from all the possible variable orders.

fam_1234 = CDVineCopSelect(U_hat, type=1, familyset = 1:6, selectioncrit =
"AIC", indeptest = T)$family
Cop_hat_1234 = CDVineMLE(U_hat, type = 1, family = fam_1234)

fam_1324 = CDVineCopSelect(U_hat[,c(1,3,2,4)], type=1, familyset = 1:6,
selectioncrit = "AIC", indeptest = T)$family
Cop_hat_1324 = CDVineMLE(U_hat[,c(1,3,2,4)], type = 1, family = fam_1324)

fam_1432 = CDVineCopSelect(U_hat[,c(1,4,3,2)], type=1, familyset = 1:6,
selectioncrit = "AIC", indeptest = T)$family
Cop_hat_1432 = CDVineMLE(U_hat[,c(1,4,3,2)], type = 1, family = fam_1432)

fam_2134 = CDVineCopSelect(U_hat[,c(2,1,3,4)], type=1, familyset = 1:6,
selectioncrit = "AIC", indeptest = T)$family
Cop_hat_2134 = CDVineMLE(U_hat[,c(2,1,3,4)], type = 1, family = fam_2134)

fam_2314 = CDVineCopSelect(U_hat[,c(2,3,1,4)], type=1, familyset = 1:6,
selectioncrit = "AIC", indeptest = T)$family
Cop_hat_2314 = CDVineMLE(U_hat[,c(2,3,1,4)], type = 1, family = fam_2314)

fam_2413 = CDVineCopSelect(U_hat[,c(2,4,1,3)], type=1, familyset = 1:6,
selectioncrit = "AIC", indeptest = T)$family
Cop_hat_2413 = CDVineMLE(U_hat[,c(2,4,1,3)], type = 1, family = fam_2413)
```

```
fam_3124 = CDVineCopSelect(U_hat[,c(3,1,2,4)], type=1, familyset = 1:6,
selectioncrit = "AIC", indeptest = T)$family
Cop_hat_3124 = CDVineMLE(U_hat[,c(3,1,2,4)], type = 1, family = fam_3124)

fam_3214 = CDVineCopSelect(U_hat[,c(3,2,1,4)], type=1, familyset = 1:6,
selectioncrit = "AIC", indeptest = T)$family
Cop_hat_3214 = CDVineMLE(U_hat[,c(3,2,1,4)], type = 1, family = fam_3214)

fam_3412 = CDVineCopSelect(U_hat[,c(3,4,1,2)], type=1, familyset = 1:6,
selectioncrit = "AIC", indeptest = T)$family
Cop_hat_3412 = CDVineMLE(U_hat[,c(3,4,1,2)], type = 1, family = fam_3412)

fam_4123 = CDVineCopSelect(U_hat[,c(4,1,2,3)], type=1, familyset = 1:6,
selectioncrit = "AIC", indeptest = T)$family
Cop_hat_4123 = CDVineMLE(U_hat[,c(4,1,2,3)], type = 1, family = fam_4123)

fam_4231 = CDVineCopSelect(U_hat[,c(4,2,3,1)], type=1, familyset = 1:6,
selectioncrit = "AIC", indeptest = T)$family
Cop_hat_4231 = CDVineMLE(U_hat[,c(4,2,3,1)], type = 1, family = fam_4231)

fam_4312 = CDVineCopSelect(U_hat[,c(4,3,1,2)], type=1, familyset = 1:6,
selectioncrit = "AIC", indeptest = T)$family
Cop_hat_4312 = CDVineMLE(U_hat[,c(4,3,1,2)], type = 1, family = fam_4312)

# We check the value of the AIC for each order.

AIC_1234 = CDVineAIC(U_hat, family = fam_1234, par = Cop_hat_1234$par, par2 =
Cop_hat_1234$par2, type = 1)$AIC
AIC_1324 = CDVineAIC(U_hat[,c(1,3,2,4)], family = fam_1324, par =
Cop_hat_1324$par, par2 = Cop_hat_1324$par2, type = 1)$AIC
AIC_1432 = CDVineAIC(U_hat[,c(1,4,3,2)], family = fam_1432, par =
Cop_hat_1432$par, par2 = Cop_hat_1432$par2, type = 1)$AIC
AIC_2134 = CDVineAIC(U_hat[,c(2,1,3,4)], family = fam_2134, par =
Cop_hat_2134$par, par2 = Cop_hat_2134$par2, type = 1)$AIC
AIC_2314 = CDVineAIC(U_hat[,c(2,3,1,4)], family = fam_2314, par =
Cop_hat_2314$par, par2 = Cop_hat_2314$par2, type = 1)$AIC
AIC_2413 = CDVineAIC(U_hat[,c(2,4,1,3)], family = fam_2413, par =
Cop_hat_2413$par, par2 = Cop_hat_2413$par2, type = 1)$AIC
AIC_3124 = CDVineAIC(U_hat[,c(3,1,2,4)], family = fam_3124, par =
Cop_hat_3124$par, par2 = Cop_hat_3124$par2, type = 1)$AIC
AIC_3214 = CDVineAIC(U_hat[,c(3,2,1,4)], family = fam_3214, par =
Cop_hat_3214$par, par2 = Cop_hat_3214$par2, type = 1)$AIC
AIC_3412 = CDVineAIC(U_hat[,c(3,4,1,2)], family = fam_3412, par =
Cop_hat_3412$par, par2 = Cop_hat_3412$par2, type = 1)$AIC
AIC_4123 = CDVineAIC(U_hat[,c(4,1,2,3)], family = fam_4123, par =
Cop_hat_4123$par, par2 = Cop_hat_4123$par2, type = 1)$AIC
AIC_4231 = CDVineAIC(U_hat[,c(4,2,3,1)], family = fam_4231, par =
Cop_hat_4231$par, par2 = Cop_hat_4231$par2, type = 1)$AIC
AIC_4312 = CDVineAIC(U_hat[,c(4,3,1,2)], family = fam_4312, par =
Cop_hat_4312$par, par2 = Cop_hat_4312$par2, type = 1)$AIC

AIC_1234
```

```
AIC_1324
AIC_1432
AIC_2134
AIC_2314
AIC_2413
AIC_3124
AIC_3214
AIC_3412
AIC_4123
AIC_4231
AIC_4312

min(c(AIC_1234, AIC_1324, AIC_1432, AIC_2134, AIC_2314, AIC_2413, AIC_3124,
AIC_3214, AIC_3412, AIC_4123, AIC_4231, AIC_4312))

# We choose the appropriate order for the U_hat, according to the value of the
# corresponding AIC. In this case and with this criterion, we choose 1, 2, 3, 4.

Cop_hat = Cop_hat_1234
fam_copHat = fam_1234

# We save the estimated parameters.
par_hat = Cop_hat$par
par2_hat = Cop_hat$par2

tau_hat = matrix(nrow = (d*(d-1)/2), ncol = 1)
for(i in 1:(d*(d-1)/2))
{
	tau_hat[i] = BiCopPar2Tau(fam_copHat[i], par = par_hat[i], par2 =
par2_hat[i])
}

###########################################################################
# Once the estimations of all the parameters have been performed, we can
# organize and visualize all these estimations.
###########################################################################

# Fitted marginal parameters.
est_marginal = matrix(nrow = d, ncol = 5)
for(i in 1:d)
{
	est_marginal[i,] = c(mu_hat[i], omega_hat[i], alpha_hat[i], beta_hat[i],
nu_hat[i])
}
colnames(est_marginal) = c("mu_hat", "omega_hat", "alpha_hat", "beta_hat",
"nu_hat")
rownames(est_marginal) = col_names

est_marginal

# Fitted parameters of the copula.
est_cop = matrix(nrow = (d*(d-1)/2), ncol = 3)
```

```
est_cop[,1] = par_hat
est_cop[,2] = par2_hat
est_cop[,3] = tau_hat

colnames(est_cop) = c("par", "par2", "tau de Kendall")
# Watch out for the copula names!
rownames(est_cop) = c("copula NASDAQ-Dow Jones: t", "copula NASDAQ-BMV: Gauss",
"copula NASDAQ-IBEX35: t", "copula Dow Jones-BMV | NASDAQ: Frank", "copula Dow
Jones-IBEX35 | NASDAQ: Gauss", "copula BMV-IBEX35 | NASDAQ, Dow Jones: Frank")

est_cop

###########################################################################
# Now, we check the estimated volatilities using the estimated innovations
# and the estimated values of the GARCH parameters.
###########################################################################

h_hat = matrix(nrow = T, ncol = d)

h_hat[1,] = omega_hat

for(k in 2:T)
{
        for(i in 1:d)
        {
                h_hat[k,i] = omega_hat[i] + alpha_hat[i]*( Y[(k-1),i] -
mu_hat[i] )^2 + beta_hat[i]*h_hat[(k-1),i]
        }
}

# Now we plot, for each series, the estimated volatility.
# First, we do a separate plot for each series.
for(i in 1:d)
{
        plot(row_names[2:(T+1)], h_hat[,i], type = "l", xlab = "t", ylab =
"h_it", col = "blue", main = paste("Volatilidades estimadas", col_names[i]))
        readline(prompt="Press [enter] to continue")
}

# Now we put all d plots in a single window.
par(mfrow = c(d,1))
for(i in 1:d)
{
        plot(row_names[2:(T+1)], h_hat[,i], type = "l", xlab = "t", ylab =
"h_it", col = "blue", main = paste("Volatilidades estimadas", col_names[i]))
}

###########################################################################
# Now we compute the predictive VaR for time T+1
###########################################################################
```

```
alphaVaR = 0.05

# First, we compute the VaR with the estimated parameters and volatilities for
each
# individual series (just to get to know each series a bit).

VaR_hat = matrix(nrow = d, ncol = 1)

for(i in 1:d)
{
	VaR_hat[i] = -(mu_hat[i] + sqrt(h_hat[T,i])*qt(alphaVaR, df = nu_hat[i],
lower.tail = TRUE))
}

colnames(VaR_hat) = "VaR"
rownames(VaR_hat) = col_names
VaR_hat

############################################################################
# Given a vector of weights, we compute the portfolio VaR using simulation.
############################################################################

# Number of simulations for the prediction.
M = 5000

# Us for the innovations with the appropiate dependance.
U_VaR_hat = CDVineSim(M, family = fam_copHat, par = par_hat, par2 = par2_hat,
type = 1)

# "Observations" from the Student's-t distribution.
aux_hat = matrix(nrow = M, ncol = d)
for(j in 1:M)
{
	for(i in 1:d)
	{
		aux_hat[j,i] = qt(U_VaR_hat[j,i], df = nu_hat[i])
	}
}

# Simulation of the innovations for time T+1
sim_Eps_hat = matrix(nrow = M, ncol = d)
for(j in 1:M)
{
	for(i in 1:d)
	{
		sim_Eps_hat[j,i] = sqrt((nu_hat[i] - 2)/nu_hat[i]) *
aux_hat[j,i]
	}
}

# Computation of the simulated volatilities and simulation of
# the log-returns for time T+1
```

```
sim_h_hat = matrix(nrow = M, ncol = d)
sim_Data_hat = matrix(nrow = M, ncol = d)

for(j in 1:M)
{
        for(i in 1:d)
        {
                sim_h_hat[j,i] = omega_hat[i] + alpha_hat[i]*( Data[T,i] -
mu_hat[i] )^2 + beta_hat[i]*h_hat[T,i]
                sim_Data_hat[j,i] = mu_hat[i] + sqrt(sim_h_hat[j,i]) *
sim_Eps_hat[j,i]
        }
}

# Now, we build the portfolio, given a set of weights, where
# w[i] is the weight of the i-th asset.
a_w = c(0.26, 0.47, 0.11)
w = matrix(c(a_w, (1-sum(a_w))), ncol = d, nrow = 1)

if(sum(w) != 1)
{
        print("Careful! The portfolio weights don't add up to 1")
}

# With the established weights, we will build a vector that contains
# the simulated porfolio "log-return" at time T+1.

portfolio_hat = sim_Data_hat %*% t(w)
hist(portfolio_hat)

# With the simulations of portfolios at time T+1 (with the real parameters),
# we estimate the predictive VaR for time T+1.

VaR_hat = -quantile(portfolio_hat, alphaVaR)

# Once we have estimated the VaR, we can estimate the CVaR
# (Conditional VaR or expected shortfall)

subSample_hat = NULL

for(j in 1:M)
{
        if(portfolio_hat[j] <= (-VaR_hat))
        {
                subSample_hat = rbind(subSample_hat, portfolio_hat[j])
        }
}

CVaR_hat = -mean(subSample_hat)

# Estimated (by simulation) predictive VaR with estimated parameters.
portfolio_risk = cbind(VaR_hat, CVaR_hat)
```

```
colnames(portfolio_risk) = c("VaR", "CVaR")

portfolio_risk
```