Applied econometrics 5, Copulas for portfolio modeling

Евгений Орлов

21.11.2014

### Данные

В качестве входных данных взяты цены закрытия по обыкновенным акциям Газпрома и Норильского Никеля в период с 01.01.2010 по 07.11.2014 включительно.

Источником данных выступила ИТС QUIK.

gazp.df <- read.csv("ГАЗПРОМ ао [Price].txt", header=TRUE)  
gmkn.df <- read.csv("ГМКНорНик [Price].txt", header=TRUE)  
tail(gazp.df)

## X.TICKER. X.PER. X.DATE. X.TIME. X.OPEN. X.HIGH. X.LOW. X.CLOSE.  
## 2191 GAZP [TQBR] Daily 20141030 0 138.09 139.90 137.62 139.50  
## 2192 GAZP [TQBR] Daily 20141031 0 140.50 141.50 139.49 141.50  
## 2193 GAZP [TQBR] Daily 20141103 0 141.00 144.00 140.26 143.39  
## 2194 GAZP [TQBR] Daily 20141105 0 143.01 143.82 141.11 143.20  
## 2195 GAZP [TQBR] Daily 20141106 0 143.35 146.25 143.05 144.10  
## 2196 GAZP [TQBR] Daily 20141107 0 143.50 147.00 142.73 144.46  
## X.VOL.  
## 2191 5463327  
## 2192 7197166  
## 2193 1966065  
## 2194 4707525  
## 2195 5571670  
## 2196 8263385

# Цены закрытия начиная с 01.01.2010 по 07.11.2014 включительно  
gazp.close <- gazp.df[gazp.df$X.DATE. >= 20100101, "X.CLOSE."]  
gmkn.close <- gmkn.df[gmkn.df$X.DATE. >= 20100101, "X.CLOSE."]

Была произведена проверка соответствия дат цен активов:

# Даты  
gazp.dates <- gazp.df[gazp.df$X.DATE. >= 20100101, "X.DATE."]  
gazp.dates <- as.Date(as.character(gazp.dates), format="%Y%m%d")  
gmkn.dates <- gmkn.df[gmkn.df$X.DATE. >= 20100101, "X.DATE."]  
gmkn.dates <- as.Date(as.character(gmkn.dates), format="%Y%m%d")  
date.comp <- gazp.dates == gmkn.dates  
if (sum(date.comp) / length(date.comp) == TRUE) {  
 print("Даты совпадают!")  
} else {  
 print("Даты не совпадают!")  
}

## [1] "Даты совпадают!"

На основе цен закрытия получен вектор доходностей каждого актива:

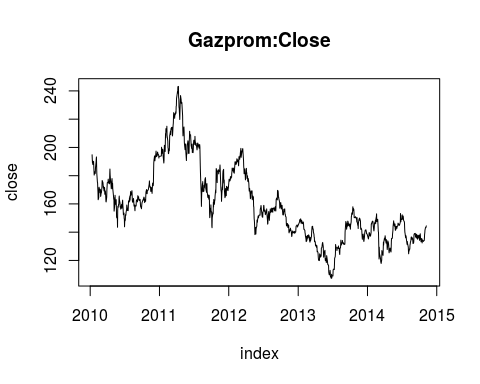
# Размер выборки  
samp.len <- length(gazp.close)  
print(paste0("Размер выборки: ", samp.len))

## [1] "Размер выборки: 1214"

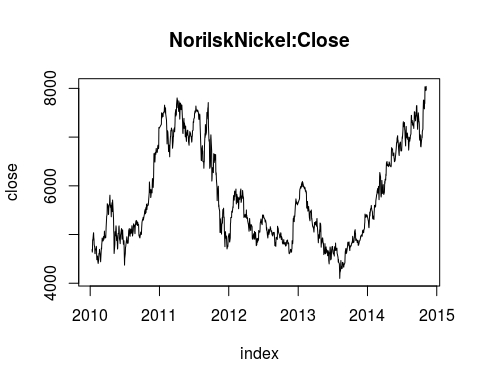
# Вектор доходностей  
gazp.ret <- gazp.close[2:samp.len] / gazp.close[1:(samp.len-1)] - 1  
gmkn.ret <- gmkn.close[2:samp.len] / gmkn.close[1:(samp.len-1)] - 1

Графики цен закрытия и доходностей:

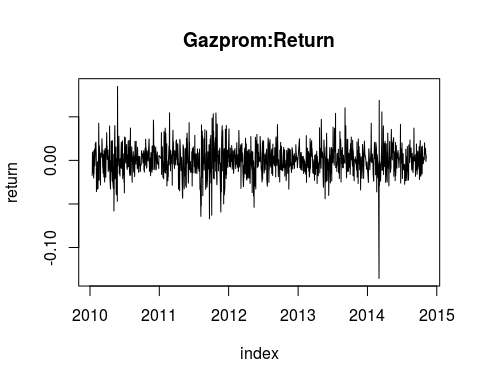
# Графики цен закрытия  
plot(gazp.dates, gazp.close,   
 type='l', xlab='index', ylab='close', main='Gazprom:Close')



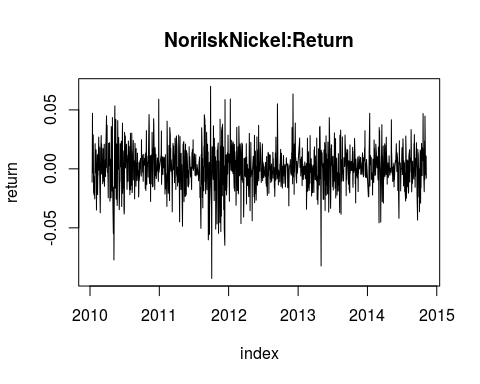
plot(gmkn.dates, gmkn.close,   
 type='l', xlab='index', ylab='close', main='NorilskNickel:Close')



# Графики доходностей  
plot(gazp.dates[-1], gazp.ret,   
 type='l', xlab='index', ylab='return', main='Gazprom:Return')



plot(gmkn.dates[-1], gmkn.ret,   
 type='l', xlab='index', ylab='return', main='NorilskNickel:Return')



### Моделирование копулы

Для того, чтобы подобрать копулу для портфеля из двух активов (50% Газпром, 50% НорНикель) подберем каждому активу распределение из семейства обобщенного гиперболического распределения и сгенерируем псевдо-наблюдения на основе подобранных функций распределения.

library(ghyp, quietly=FALSE)

## Loading required package: numDeriv  
## Loading required package: gplots  
##   
## Attaching package: 'gplots'  
##   
## The following object is masked from 'package:stats':  
##   
## lowess

# моделирование частных функций распределения  
gazp.fit <- stepAIC.ghyp(gazp.ret, dist=c("gauss", "t", "ghyp"),   
 symmetric=NULL, silent=TRUE)$best.model

## Currently fitting: asymmetric t   
## Currently fitting: asymmetric ghyp   
## Currently fitting: symmetric t   
## Currently fitting: symmetric ghyp   
## Currently fitting: gauss

gmkn.fit <- stepAIC.ghyp(gmkn.ret, dist=c("gauss", "t", "ghyp"),   
 symmetric=NULL, silent=TRUE)$best.model

## Currently fitting: asymmetric t   
## Currently fitting: asymmetric ghyp   
## Currently fitting: symmetric t   
## Currently fitting: symmetric ghyp   
## Currently fitting: gauss

# расчёт значений F1(u) и F2(u)  
gazp.cdf <- pghyp(gazp.ret, object=gazp.fit)  
gmkn.cdf <- pghyp(gmkn.ret, object=gmkn.fit)  
cdf <- array(c(gazp.cdf, gmkn.cdf), dim=c(length(gazp.ret), 2))

На основе сгенерированных наблюдений подберем параметры для ряда копул (для подбора параметров используем метод по умолчанию, квази-ММП).

library(copula, quietly=TRUE)   
# объявление копулы  
norm.cop <- normalCopula(dim=2, param=0.5, dispstr='un')   
stud5.cop <- tCopula(dim=2, param=0.5, df=5, df.fixed=TRUE, dispstr='un')   
stud10.cop <- tCopula(dim=2, param=0.5, df=10, df.fixed=TRUE, dispstr='un')   
stud20.cop <- tCopula(dim=2, param=0.5, df=20, df.fixed=TRUE, dispstr='un')   
gumb.cop <- gumbelCopula(dim=2, param=2)   
clay.cop <- claytonCopula(dim=2, param=2)   
# подгонка копулы  
norm.fit <- fitCopula(cdf, copula=norm.cop)   
stud5.fit <- fitCopula(cdf, copula=stud5.cop)   
stud10.fit <- fitCopula(cdf, copula=stud10.cop)   
stud20.fit <- fitCopula(cdf, copula=stud20.cop)   
gumb.fit <- fitCopula(cdf, copula=gumb.cop)   
clay.fit <- fitCopula(cdf, copula=clay.cop)

### Оценка финансового риска портфеля с помощью копулы

Из построенных копул выберем лучшую на основе информационного критерия Акаике и метода максимального правдоподобия (в данном случае оба подхода ведут к одному результату).

# выбор оптимальной копулы  
norm.fit@loglik;stud5.fit@loglik;stud10.fit@loglik

## [1] 214.3339

## [1] 233.3042

## [1] 231.5195

stud20.fit@loglik;gumb.fit@loglik;clay.fit@loglik

## [1] 225.8214

## [1] 194.3105

## [1] 200.0522

AIC(norm.fit);AIC(stud5.fit);AIC(stud10.fit)

## [1] -426.6679

## [1] -464.6083

## [1] -461.039

AIC(stud20.fit);AIC(gumb.fit);AIC(clay.fit)

## [1] -449.6429

## [1] -386.6209

## [1] -398.1044

Для расчета VaR и ES будем использовать копулу Стьюдента с 5 степенями свободы.

C помощью копулы сгенерируем значения частных функций распределения и оценим  
соответствующие квантили распределений для получения доходностей. На основе полученных векторов доходностей оценим значения VaR и ES.

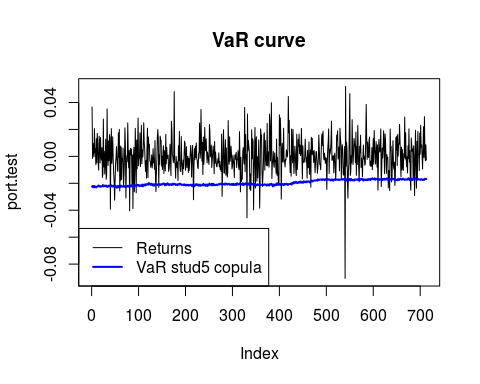
# значения частных функций распределения  
N <- 10^3  
stud5.sim <- rCopula(n=N, copula=stud5.fit@copula)   
  
# доходности активов  
gazp.sim <- qghyp(stud5.sim[, 1], object=gazp.fit)  
gmkn.sim <- qghyp(stud5.sim[, 2], object=gmkn.fit)  
w <- c(0.5, 0.5)  
port.sim <- w[1]\*gazp.sim + w[2]\*gmkn.sim  
  
# измерители риска  
alpha <- 0.1  
port.sim <- sort(port.sim)  
VaR <- port.sim[alpha\*N]  
ES <- mean(port.sim[1:(alpha\*N-1)])

### Кривая VaR с помощью копулы Стьюдента

Кривая VaR строилась для 95%-го VaR с использованием последних 500 известных значений доходностей активов. Ввиду ограниченной вычислительной мощности компьютера для распределения индивидуальных активов использовались симметричные распределения Гаусса и Стьюдента. Доходности портфеля генерировались на основе копулы Стьюдента с 5 степенями свододы, которая была признана оптимальной при подгоне по всей выборке на предыдущем шаге.

# Параметры для расчета VaR  
T1 <- 500  
T2 <- length(gazp.ret) - T1  
alpha <- 0.05  
N <- 10^4  
w <- c(0.5, 0.5) # Веса активов в портфеле  
VaR.copula <- numeric()  
  
h <- T1 # Длина обучающей выборки  
for (i in (T1+1):(T1+T2)) {  
 print((i-T1)/T2)  
 # Обучающая выборка  
 gazp.train <- gazp.ret[(i-h):(i-1)]   
 gmkn.train <- gmkn.ret[(i-h):(i-1)]  
 # Подгон частных функций распределения  
 gazp.fit <- stepAIC.ghyp(gazp.train,dist=c("gauss", "t"),   
 symmetric=TRUE, silent=TRUE)$best.model   
 gmkn.fit <- stepAIC.ghyp(gmkn.train,dist=c("gauss", "t"),   
 symmetric=TRUE, silent=TRUE)$best.model   
 gazp.cdf <- pghyp(gazp.train, object=gazp.fit)   
 gmkn.cdf <- pghyp(gmkn.train, object=gmkn.fit)   
 cdf <- array(c(gazp.cdf, gmkn.cdf), dim=c(length(gazp.ret), 2))  
 # Подгон копулы  
 stud5.fit <- fitCopula(cdf, copula=stud5.cop)  
 # Симуляция доходности портфеля  
 stud5.sim <- rCopula(n=N,copula=stud5.fit@copula)   
 gazp.sim <- qghyp(stud5.sim[, 1], object=gazp.fit)   
 gmkn.sim <- qghyp(stud5.sim[, 2], object=gmkn.fit)   
 port.sim <- w[1]\*gazp.sim + w[2]\*gmkn.sim   
 port.sim <- sort(port.sim)  
 # Расчет VaR  
 VaR.copula[i-T1] <- port.sim[alpha\*N]   
}

# График кривой VaR  
port.test <- w[1]\*gazp.ret[(T1+1):(T1+T2)] + w[2]\*gmkn.ret[(T1+1):(T1+T2)]  
plot(port.test, type="l", main="VaR curve")  
lines(VaR.copula, col="blue", lwd=2)  
legend('bottomleft', c("Returns", "VaR stud5 copula"), col=c("black", "blue"),   
 lty=c(1, 1), lwd=c(1, 2))



Для верификации кривой VaR определим функцию для теста Купика.

# Частота пробоев  
kupiec.test <- function(ret, VaR, alpha) {  
 # Тест Купика:  
 # H0: модельная и эмпирическая частоты пробоя VaR совпадают  
 K <- sum(ret < VaR)  
 T2 <- length(ret)  
 alpha0 <- K / T2  
 S <- -2\*log((1-alpha)^(T2-K) \* alpha^K) + 2\*log((1-alpha0)^(T2-K) \* alpha0^K)  
 p.value <- 1-pchisq(S, df=1)  
 return(c(alpha0, p.value))  
}

## [1] "Kupiec test, alpha = 0.05"

## [1] "GPD: alpha0 = 0.054698, p-value = 0.570426"

p-value теста Купика равно 0.57, что неплохо, учитывая наложенные малой мощностью машины ограничения на моделирование и объем симуляции.