Applied Econometrics 4, VaR curve using GARCH

Евгений Орлов

20.11.2014

### Данные

В качестве входных данных взяты цены закрытия по обыкновенным акциям Сбербанка в период с 01.01.2010 по 07.11.2014 включительно.

Источником данных выступила ИТС QUIK.

sber.df <- read.csv("Сбербанк [Price].txt", header=TRUE)  
tail(sber.df)

## X.TICKER. X.PER. X.DATE. X.TIME. X.OPEN. X.HIGH. X.LOW. X.CLOSE.  
## 2997 SBER [TQBR] Daily 20141030 0 73.70 75.13 73.43 74.98  
## 2998 SBER [TQBR] Daily 20141031 0 75.45 76.84 75.23 76.23  
## 2999 SBER [TQBR] Daily 20141103 0 76.07 76.66 75.61 76.29  
## 3000 SBER [TQBR] Daily 20141105 0 75.80 75.94 75.02 75.65  
## 3001 SBER [TQBR] Daily 20141106 0 75.76 76.88 74.75 75.05  
## 3002 SBER [TQBR] Daily 20141107 0 75.16 77.65 74.41 75.53  
## X.VOL.  
## 2997 18099518  
## 2998 15467572  
## 2999 4989221  
## 3000 10108565  
## 3001 10972235  
## 3002 17931891

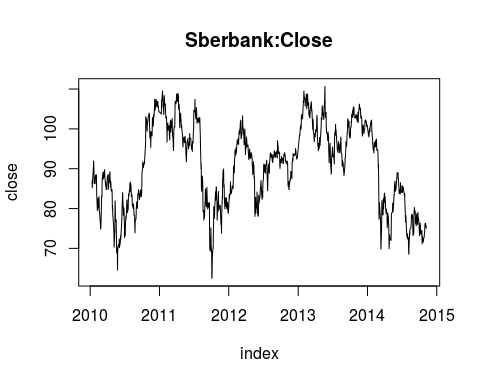
# Цены закрытия начиная с 01.01.2010 по 07.11.2014 включительно  
sber.close <- sber.df[sber.df$X.DATE. >= 20100101, "X.CLOSE."]  
# Длина выборки   
scl <- length(sber.close)  
print(paste0("Размер выборки: ", scl))

## [1] "Размер выборки: 1211"

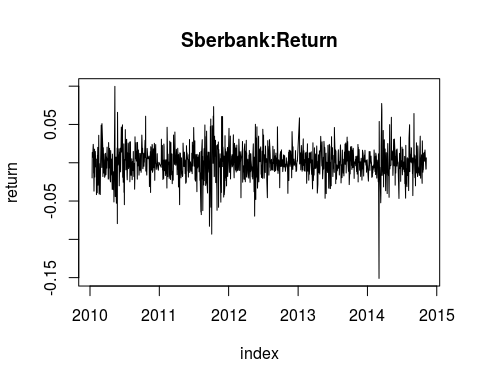
# Вектор доходностей  
sber.ret <- sber.close[2:scl] / sber.close[1:(scl-1)] - 1

Графики цен закрытия и доходностей:

# График цен закрытия  
plot(sber.dates, sber.close,   
 type='l', xlab='index', ylab='close', main='Sberbank:Close')



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



### Тесты на стационарность

Проведем тесты на стационарность доходностей:

library(tseries, quietly=TRUE)  
# ADF-тест  
adf.test(sber.ret)

## Warning in adf.test(sber.ret): p-value smaller than printed p-value

##   
## Augmented Dickey-Fuller Test  
##   
## data: sber.ret  
## Dickey-Fuller = -10.5177, Lag order = 10, p-value = 0.01  
## alternative hypothesis: stationary

# PP-тест  
pp.test(sber.ret)

## Warning in pp.test(sber.ret): p-value smaller than printed p-value

##   
## Phillips-Perron Unit Root Test  
##   
## data: sber.ret  
## Dickey-Fuller Z(alpha) = -1062.767, Truncation lag parameter = 7,  
## p-value = 0.01  
## alternative hypothesis: stationary

# KPSS-тест  
kpss.test(sber.ret, null='Level')

## Warning in kpss.test(sber.ret, null = "Level"): p-value greater than  
## printed p-value

##   
## KPSS Test for Level Stationarity  
##   
## data: sber.ret  
## KPSS Level = 0.0584, Truncation lag parameter = 8, p-value = 0.1

Все три теста говорят в пользу стационарности временного ряда доходностей: для расширенного теста Дики-Фуллера и теста Филлипса-Перрона гипотеза о наличии единичного корня отвергается на уровне значимости 0.01, гипотеза о стационарности временного ряда не отвергается на уровне значимости 0.1 для теста Квятковского-Филлипса-Шмидта-Шина.

### Оптимальный порядок модели ARMA(m, n)

Определим функцию для поиска оптимальных значений коэффициентов модели ARMA(m, n) на основе информационного критерия Акаике:

arma.coef.aic <- function(ret, a, b) {  
 # Выбор оптимальныэх коэффициентов ARMA(a, b) для ret  
 coefs <- expand.grid(a=a, b=b)  
 aic <- apply(coefs, 1,   
 function(coef) AIC(arima(ret, order=c(coef[1], 0, coef[2]))))  
 ind <- which(aic == min(aic))  
 return(c(coefs$a[ind], coefs$b[ind]))  
}

Определим оптимальные коэффициенты модели ARMA(m, n) () для вектора доходностей акций Сбербанка:

## Наилучшие параметры m (1:5) и n (1:5) для модели ARMA(m,n)   
arma.coef.aic(sber.ret, 1:5, 1:5)

## Warning in arima(ret, order = c(coef[1], 0, coef[2])): possible  
## convergence problem: optim gave code = 1

## Warning in arima(ret, order = c(coef[1], 0, coef[2])): possible  
## convergence problem: optim gave code = 1

## Warning in arima(ret, order = c(coef[1], 0, coef[2])): possible  
## convergence problem: optim gave code = 1

## [1] 2 2

Для используемой выборки доходностей наилучшей моделью является ARMA(2, 2).

### Тест на ARCH-эффекты

library(FinTS, quietly=TRUE)

##   
## Attaching package: 'zoo'  
##   
## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

# Тест множителей Лагранжа (LM-тест)  
ArchTest(sber.ret, lags=12)

##   
## ARCH LM-test; Null hypothesis: no ARCH effects  
##   
## data: sber.ret  
## Chi-squared = 93.2742, df = 12, p-value = 1.144e-14

p-value теста очень низкое (<< 0.01), что указывает на наличие ARCH-эффектов для ряда доходностей. Значит, в данном случае для моделирования временного ряда доходностей оправданно использовать ARCH-модели.

### Оценка риска на основе модели ARMA-GARCH

Рассчитаем оценки VaR и ES по всей выборке наблюдений по модели ARMA(m, n)-GARCH(p, q).

Ввиду высокой вычислительной сложности задачи в качестве параметров ARMA(m, n) возьмем полученные ранее значения 2, 2.

Параметры p, q GARCH(p, q) найдем на основании критерия Акаике для модели ARMA(2, 2)-GARCH(p, q), :

library(fGarch, quietly=TRUE)

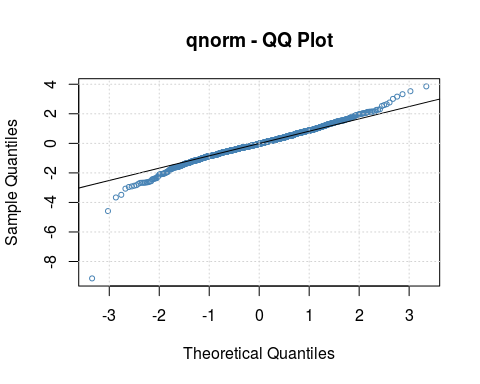
##   
## Attaching package: 'timeSeries'  
##   
## The following object is masked from 'package:zoo':  
##   
## time<-  
##   
## Loading required package: MASS  
##   
## Attaching package: 'fBasics'  
##   
## The following object is masked from 'package:base':  
##   
## norm

# Оптимизация параметров p, q GARCH(p, q)  
arma\_garch.coef.aic <- function(ret, a, b, c, d) {  
 # Выбор оптимальных коэффициентов ARMA(a, b)-GARCH(c, d) для ret  
 coefs <- expand.grid(a=a, b=b, c=c, d=d)  
 aic <- apply(coefs, 1,   
 function(coef) garchFit(substitute(~arma(a, b)+garch(c, d),   
 list(a=coef[1], b=coef[2],   
 c=coef[3], d=coef[4])),   
 data=ret,   
 trace=FALSE)@fit$ics[1])  
 ind <- which(aic == min(aic))  
 return(c(coefs$a[ind], coefs$b[ind], coefs$c[ind], coefs$d[ind]))   
}  
arma\_garch.coef.aic(sber.ret, 2, 2, 1:5, 1:5)

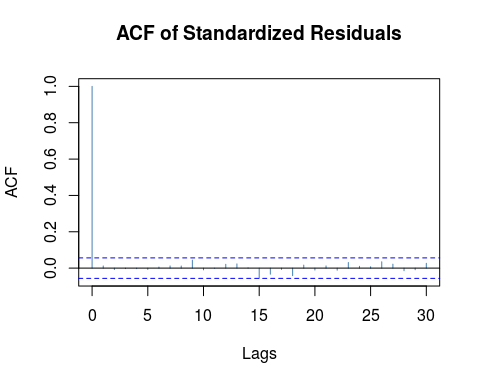
## [1] 2 2 2 1

В качестве наилучшей была отобрана модель ARMA(2, 2)+GARCH(2, 1).

# Наилучшая модель  
sber.gfit <- garchFit(~arma(2, 2)+garch(2, 1),   
 data=sber.ret, trace=FALSE)  
plot(sber.gfit, which=13)



plot(sber.gfit, which=10)



Используя наилучшую модель рассчитаем оценку для 95%-го 1-дневного VaR:

# Расчет VaR по всей выборке наблюдений  
sber.pred <- predict(sber.gfit, n.ahead=1)

## Warning in predict.Arima(ARMA, n.ahead): MA part of model is not  
## invertible

alpha <- 0.05  
VaR <- as.numeric(sber.pred[1]+sber.pred[3]\*qged(alpha, mean=0, sd=1))  
VaR

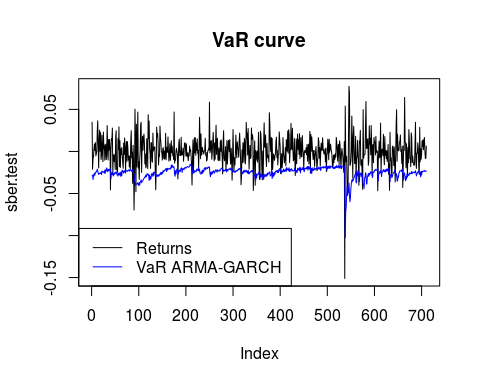
## [1] -0.02432264

### Кривая VaR

Кривая VaR построена с помощью модели ARMA(1, 1)-GARCH(1, 1), используя последние 500 известных значений доходности.

T1 <- 500  
T2 <- length(sber.ret)-T1  
VaR.arma\_garch <- numeric()  
  
h <- T1 # Длина обучающей выборки  
for (i in (T1+1):(T1+T2)) {  
 sber.train <- sber.ret[(i-h):(i-1)]  
 sber.gfit <- garchFit(formula=~arma(1, 1)+garch(1, 1),   
 data=sber.train, delta=2,   
 include.delta=FALSE, include.shape=FALSE,   
 include.skew=FALSE, trace=FALSE)  
 sber.pred <- predict(sber.gfit, n.ahead=1)  
 VaR.arma\_garch[i-T1] <- as.numeric(sber.pred[1]   
 + sber.pred[3]\*qsged(alpha, mean=0, sd=1))  
}

# График кривой VaR  
sber.test <- sber.ret[(T1+1):(T1+T2)]  
plot(sber.test, type="l", main="VaR curve")  
lines(VaR.arma\_garch, col="blue", lwd=1)  
legend('bottomleft', c("Returns", "VaR ARMA-GARCH"), col=c("black", "blue"),   
 lty=c(1, 1))



Мне пришлось использовать в качестве модели ARMA(1, 1)-GARCH(1, 1) вместо ARMA(2, 2)-GARCH(2, 1), которая показала лучший результат по критерию Акаике, так как при использовании последней модели расчет кривой VaR прерывался появлением следующей ошибки:

Error in arima(.series$x, order = c(u, 0, v), include.mean = include.mean) :   
 non-stationary AR part from CSS

К сожалению поиском в интернете найти решение для данной проблемы не удалось. Как мне показалось, лучше всего суть проблемы изложена тут:

<http://r.789695.n4.nabble.com/Problems-with-predict-in-fGarch-td3401938.html>

В связи с этим ожидания относительно качества кривой VaR были снижены.

Для верификации кривой VaR определим функции для теста Купика, функций потерь Лопеса и Бланко-Ила:

# 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))  
}  
  
# Глубина пробоев  
lopez.lf <- function(ret, VaR) {  
 # Функция потерь Лопеса  
 K <- sum(ret < VaR)  
 value <- sum((ret-VaR)^2\*(ret < VaR)) / K  
}  
  
blanco.lf <- function(ret, VaR) {  
 # Функция потерь Бланко-Ила  
 K <- sum(ret < VaR)  
 value <- sum((ret-VaR)/VaR \* (ret < VaR)) / K  
}

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

## [1] "ARMA(1, 1)-GARCH(1, 1): alpha0 = 0.067606, p-value = 0.040585"

## [1] "Lopez loss function: 0.000537"

## [1] "Blanco loss function: 0.548926"

В результате фактическая частота превышений VaR заметно превышает целевую, и модель не проходит тест Купика (p-value < 0.05).