

Попов Максим

Applied financial econometrics, task 4

Для получения данных о том или ином тикере использовалась библиотека quantmod, позволяющая загружать данные о тикере при помощи функции getSymbols(). Для расчетов использовалась колонка Adjusted данных, которая учитывает в себе различные эффекты (выплаты дивидендов, разбиение акций).

При расчете доходности использовалась формула:

$$\text{return} = (\text{today_price} - \text{yesterday_price}) / \text{yesterday_price};$$

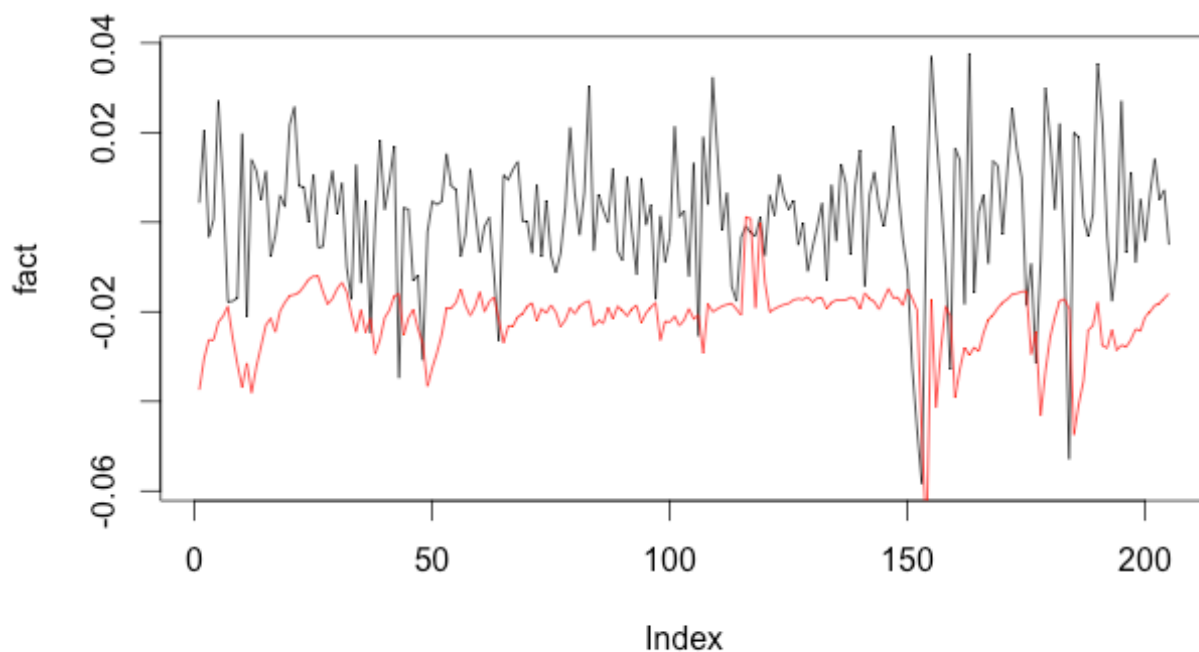
То есть для упрощения не учитывались выходные и нерабочие дни.

В примере был использован тикер ADBE.

Первоначально были рассчитаны значения VaR и ES:

VaR	-0.01933481
ES	0.001081552

После была построена кривая VaR и проведен тест Купика. Результаты построения кривой:



При этом тест Купика показал, что:

$$\alpha = 0.05 ; \alpha_0 = 0.06829268 ; p\text{-value} = 0.2537561;$$

Таким образом отбросить нулевую гипотезу, которая говорит, что $\alpha = \alpha_0$ мы не можем.

Итоговая версия скрипта представлена ниже:

```
##initialization
library("zoo")
library("xts")
library("quantmod")
library("ghyp")
library("fGarch")

getReturns <- function(prices)
{
  difference = prices[1:length(prices)-1] - prices[2:length(prices)];
  return (-difference/prices[1:length(prices)-1]);
}

##loading data
getSymbols("ADBE",from="2014-01-01",src="yahoo", return.class="data.frame");

##calculating returns
returnsADBE <- getReturns(ADBE[, 'ADBE.Adjusted']);

ADBE.gfit <- garchFit(formula=~aparch(1,1),data=returnsADBE,leverage=TRUE,cond.dist="ged",
  trace=FALSE)

ADBE.forecast <- predict(ADBE.gfit,n.ahead=10^4)

alpha <- 0.05
VaR <- ADBE.forecast[1,1]+ADBE.forecast[1,3]*qged(alpha,mean=0,sd=1, nu=ADBE.gfit@fit$par["shape"])
ABDE.forecast <- sort(ADBE.forecast[,1])
ES <- mean(ABDE.forecast[1:(alpha*10^4-1)])

T1 <- 260; T2 <- length(returnsADBE) - T1

VaR <- numeric()
h <- 0.5*260
for (i in (T1+1):(T1+T2)) {
  h.ADBE <- returnsADBE[(i-h):(i-1)]

  ADBE.gfit <- garchFit(formula=~aparch(1,1),data=h.ADBE,leverage=TRUE,cond.dist="ged",
    trace=FALSE)

  ADBE.forecast <- predict(ADBE.gfit,n.ahead=1)

  VaR[i-T1] <- ADBE.forecast[1,1]+ADBE.forecast[1,3]*qged(alpha,mean=0,sd=1, nu=ADBE.gfit@fit$par["shape"])
}

fact <- returnsADBE[(T1+1):(T1+T2)]
plot(fact,type="l")
lines(VaR,col="red")

K <- sum(fact<VaR);
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)
```