VaR using volatility models

PADS - Financial Analytics

Paloma Vaissman Uribe

Nov, 2022

Introdução

O objetivo desta análise é estimar o VaR (Value at Risk) de uma carteira utilizando modelos para a variância condicional (volatilidade). O VaR é amplamente utilizado para medir o risco de mercado para instituições financeiras.

Informalmente, do ponto de vista de risco de mercado, o VaR é uma medida da variação potencial máxima do valor de um ativo (ou carteira de ativos) sobre um período pré-fixado, com dada probabilidade. Ou seja, quanto se pode perder, com probabilidade p, sobre um horizonte h fixado. É uma medida de perda financeira associada a um evento extremo, por exemplo, a pandemia de COVID repentina ou a queda da barreira de Brumadinho, que gerou uma oscilação negativa nos preços da Vale.

Nassim Taleb (Black Swam, 2007):

"Banks are now more vulnerable to the Black Swan than ever before with"scientists" among their staff taking care of exposures. The giant firm J. P. Morgan put the entire world at risk by introducing in the nineties RiskMetrics, a phony method aiming at managing people's risks. A related method called "Value-at-Risk," which relies on the quantitative measurement of risk, has been spreading."

Dados e bibliotecas utilizadas

Para esse caso, pegaremos os dados dos últimos 5 anos das séries temporais de ações do índice IBOVESPA utilizando o pacote `quantmod`.

```
# Load libraries
library(tidyverse)
library(ggthemes)
library(forecast)
library(tseries)
library(gridExtra)
library(rugarch)
library(quantmod)
```

O Valor em Risco: teoria

Exemplo: suponha que exista uma chance de 95% de que a taxa de câmbio Real/USD não caia mais do que 1% em um dia. Suponha, ainda, que uma empresa tenha 100 milhões de reais aplicados num fundo cambial. Calculemos a perda potencial sobre este valor aplicado. Assumindo normalidade para os erros na especificação

$$r_t = \sqrt{h_t}\varepsilon_t, \quad \varepsilon_t \sim N(0,1),$$

suponha que a estimativa do desvio padrão da volatilidade seja hoje $\sqrt{h_t} = 0.46\%$. Então o VaR é calculado usando o percentil 95% da distribuição Normal:

$$VaR = 1.65\sqrt{h_t} = 1.65 \times 0.45 = 0.759\%.$$

Ou seja, não se espera que a taxa de câmbio caia mais do que 0.759% com 95% de probabilidade. A empresa, portanto, não deverá perder mais do que R$ 759 mil reais em um dia, em 95% das vezes.

O VaR irá depender se a posição é

- **Comprada (detém o ativo)** ou
- **Vendida (venda de um ativo que não possui)**.

Queremos calcular o risco da posição dentre de um horizonte $h > 0$ e a perda ou lucro da posição (P&L), definida por $\Delta P(h)$:

$$\Delta P(h) = P(t + h) - P(t),$$

onde $P(t)$ é o preço do ativo no instante t, em que queremos calcular o VaR. Ou seja, o lucro ou perda $\Delta P(h)$ é obtido marcando-se a posição a mercado hoje e deixando-a sem mudanças até uma nova marcação h dias depois.

A definição formal do VaR de uma **posição comprada** é definida por meio de

$$p = P(\Delta P(h) \leq VaR) = F_h(VaR),$$

ou seja o VaR é o p-quantil da distribuição acumulada $F_h(.)$. Na prática temos que estimar esse valor utilizando a distribuição empírica dos retornos. No caso de uma posição vendida, há perda se $\Delta P(h) > 0$, uma vez que não tenho o ativo e irei comprá-lo no dia de liquidação, ao final do horizonte h. Logo, um aumento do preço me gera perdas. Nesse caso, o VaR é definido com base em

$$p = P(\Delta P(h) \geq VaR) = 1 - F_h(VaR),$$

o que equivale à **cauda direita** da distribuição $F_h(.)$.

Resumindo: O VaR de uma posição comprada tem valor negativo, pois quem tem uma posição comprada sofre uma perda se $\Delta P(h) < 0$. Com isso para $p = 0.05$, considerando a distribuição 95% das chances esse o valor de perda será no máximo o módulo de 1.65 vezes o desvio padrão estimado.

Abordagens para cálculo do VaR

Existem algumas abordagens:

- RiskMetrics
- Modelos GARCH (abordagem econométrica)
- Quantis empíricos
- Teoria dos valores extremos

Nesse material, abordaremos somente as abordagens RiskMetrics e econométrica.

1. Abordagem RiskMetrics

RiskMetrics é um modelo e empresa antigamente pertencente ao JP Morgan. Para mais informaçãoes, ver: https://en.wikipedia.org/wiki/RiskMetrics (https://en.wikipedia.org/wiki/RiskMetrics)

Aqui, a suposição é que a distribuição condicional dos retornos dada a informação passada $\mathcal{F}_{t-1}$ é normal com média zero e variância h_t:

$$r_t | \mathcal{F}_{t-1} \sim N(0, h_t),$$

sendo a volatilidade h_t estimada por um modelo EWMA ("exponentially weighted moving average"):

$$h_t = \lambda h_{t-1} + (1 - \lambda) r_{t-1}^2, \quad 0 < \lambda < 1.$$

O parâmetro λ pode ser obtido por mínimos quadrados ou ainda, pode-se pensado como um IGARCH(1,1), como visto na última aula.

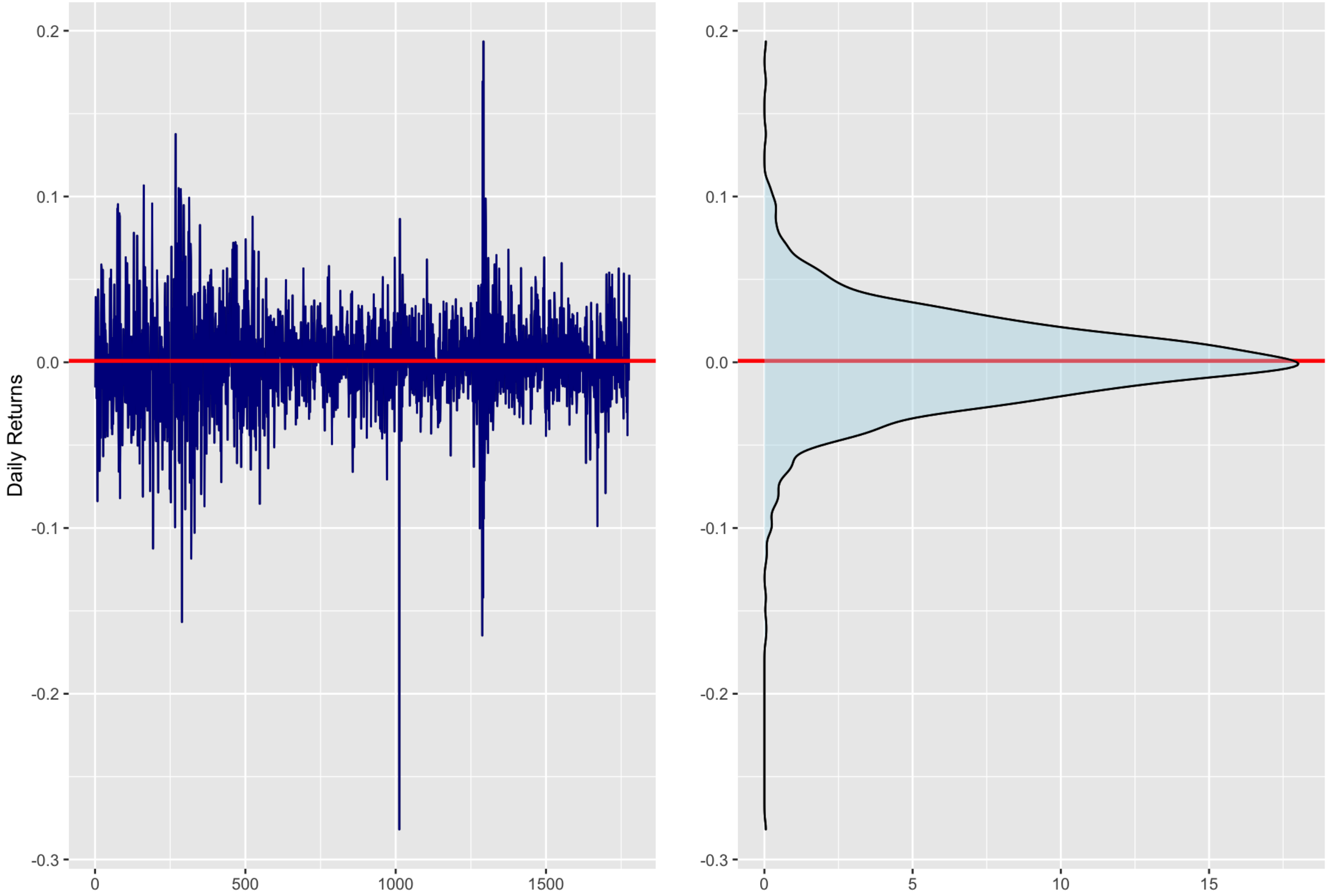
Exemplo: Vale do Rio Doce

Vamos baixar os dados utilizando o pacote `quantmod` e definir se ajustamos um modelo para a média dos retornos.

```
ticker_symbols <- c('VALE3.SA')
start_date <- '2015-01-01'
end_date <- '2022-02-28'

# Getting data from APIs
getSymbols(ticker_symbols,
           from = start_date,
           to = end_date)
```

```
## [1] "VALE3.SA"
```

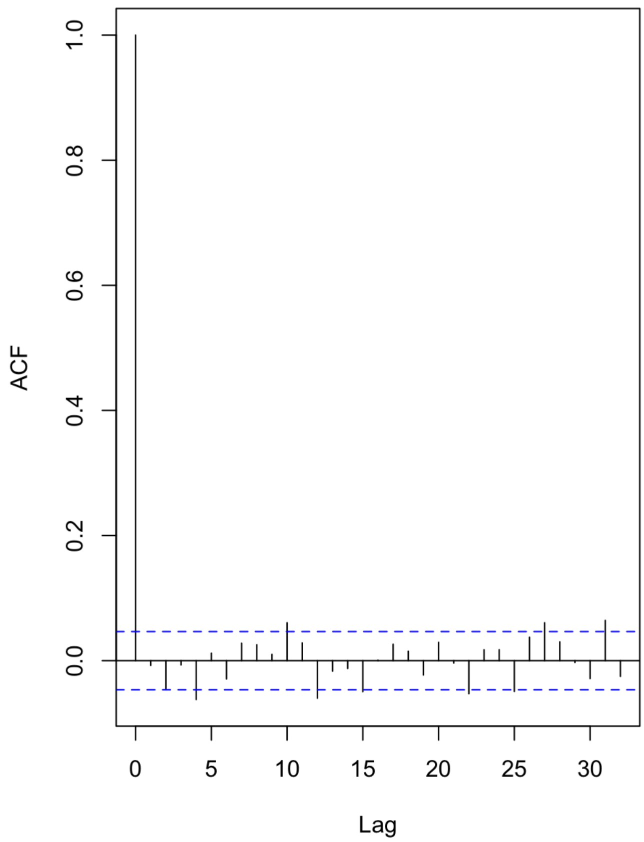
```
rets <- diff(log(VALE3.SA$VALE3.SA.Close))[-1]

p1 = qplot(x = 1:length(rets) , y = rets , geom = 'line') + geom_line(color = 'darkblue') +
    geom_hline(yintercept = mean(rets) , color = 'red' , size = 1) +
    labs(x = '' , y = 'Daily Returns')

p2 = qplot(rets , geom = 'density') + coord_flip() + geom_vline(xintercept = mean(rets) , color = 'red' ,
        size = 1) +
    geom_density(fill = 'lightblue' , alpha = 0.4) + labs(x = '')

grid.arrange(p1 , p2 , ncol = 2)
```

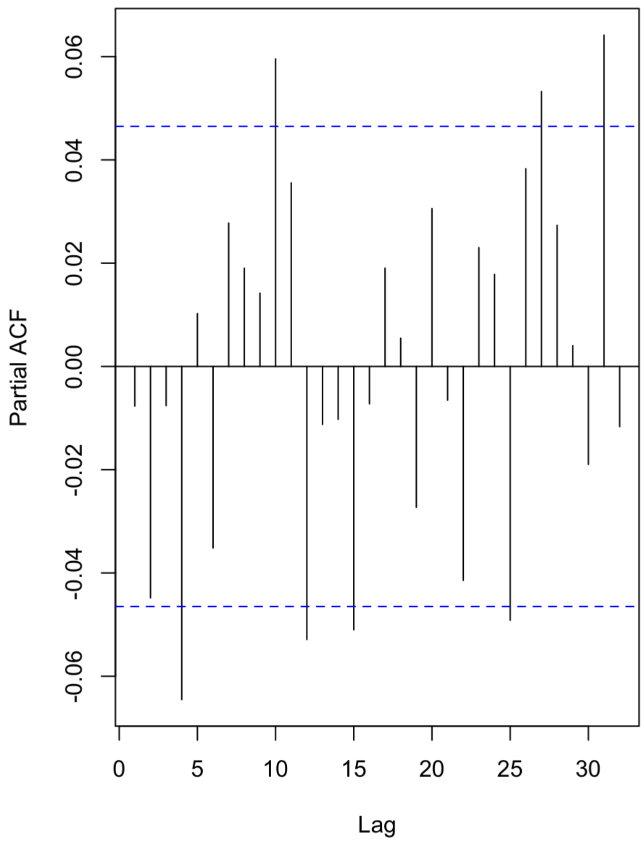


Plotando as funções de autocorrelação e autocorrelação parcial amostral, vemos que há alguns lags fora do intervalo de confiança. Portanto, podemos ajustar um modelo ARMA para a média dos retornos. Lembrando da aula sobre volatilidade, um dos **fatos estilizados em Finanças** é que os retornos em geral são não autocorrelacionados e estacionários. Entretanto, nem sempre são um ruído branco e assim temos que ajustar um ARMA (nunca um ARIMA, uma vez que não possuem em geral raíz unitária).

Utilizamos a função `forecast::auto.arima` para facilitar a primeira etapa da Metodologia de Box Jenkins: **identificação**.

```
# acf graphs to identify lags
par(mfrow=c(1,2))
acf(rets,main = "")
acf(rets, main = "",type = "partial")
```






```
# checking non stationarity and auto arima suggestions
adf.test(rets)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  rets
## Dickey-Fuller = -11.435, Lag order = 12, p-value = 0.01
## alternative hypothesis: stationary
```

```
arma_model = auto.arima(rets,
                        trace = F,
                        ic = 'aicc',
                        stepwise = FALSE)
```

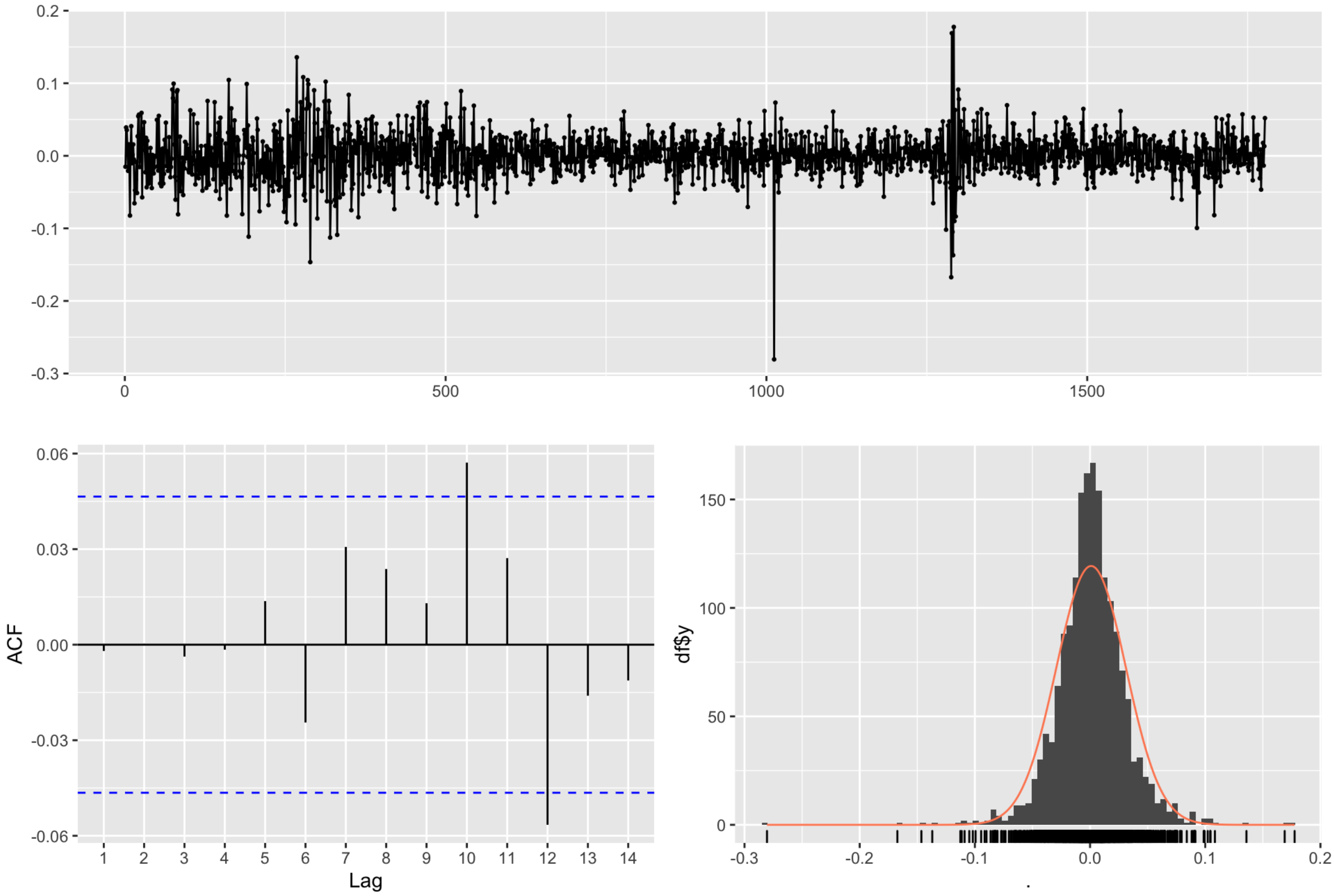
Aparentemente, o modelo gerado pelo `auto.arima` faz sentido também olhando os gráficos das ACF e PACF (com exceção do lag 4, quase todos os lags estão dentro do intervalo). Vamos então utilizar um ARMA(0,4) ou MA(4) como modelo para a média, cuja estimativa (etapa 2 de Box-Jenkins, **estimação**) é a seguinte:

```
arma_model
```

```
## Series: rets
## ARIMA(0,0,4) with zero mean
##
## Coefficients:
##           ma1      ma2      ma3      ma4
##       -0.0059  -0.0499  -0.0013  -0.0608
## s.e.   0.0237   0.0237   0.0241   0.0232
##
## sigma^2 = 0.0009167:  log likelihood = 3695.33
## AIC=-7380.67   AICc=-7380.64   BIC=-7353.26
```

Vamos passar agora para a etapa 3: **diagnóstico**. Veremos se os resíduos estão bem ajustados (ou seja, se são de fato um ruído branco).

```
arma_model$residuals %>% ggtsdisplay(plot.type = 'hist' , lag.max = 14)
```




```
ar.res = arma_model$residuals
Box.test(ar.res, lag = 10 , fitdf = 2 , type = 'Ljung-Box')
```

```
##
##  Box-Ljung test
##
## data:  ar.res
## X-squared = 10.279, df = 8, p-value = 0.246
```

De acordo com os testes de Ljung-Box, os resíduos são não autocorrelacionados, logo podemos assumir esse modelo como bem ajustado.

Embora vimos que seria razoável ajustar um modelo ARMA para os retornos (equação da média), o RiskMetrics não trabalha com essa hipótese, assumindo que os retornos são um ruído branco estacionário. Portanto, vamos ajustar um modelo IGARCH(1,1) ou EWMA para a volatilidade, considerando média zero para os retornos e sem ajuste prévio de um ARMA.

```
# Model specification
model.spec = ugarchspec(
  variance.model = list(model = 'iGARCH', garchOrder = c(1 , 1)),
  mean.model = list(armaOrder = c(0, 0),
                  include.mean=F)
  )
model.fit = ugarchfit(spec = model.spec , data = rets)
model.fit@fit$matcoef
```

```
##            Estimate   Std. Error  t value     Pr(>|t|)
## omega  8.918355e-06 2.147813e-06 4.152296 3.291557e-05
## alpha1 6.889953e-02 7.993741e-03 8.619184 0.000000e+00
## beta1  9.311005e-01           NA       NA           NA
```

A previsão 1 passo à frente da volatilidade e o $VaR_{95\%} = 1.65 \times \hat{h}_t(1)$ é igual a

```
# Model specification
fore = ugarchforecast(model.fit,n.ahead=1,data=rets)
VaR = fore@forecast$sigmaFor*1.65*1000000
VaR
```

```
##     2022-02-25
## T+1   43734.41
```

Concluímos que a volatilidade prevista (desvio padrão) para o dia seguinte é 0.02651, aproximadamente. Assim, se um fundo assumir uma posição comprada em Vale de R$ 1 MM, espera-se uma perda potencial máxima (VaR) de R$ 43,734.41, aproximadamente para o dia seguinte.

Para calcular o VaR para k períodos à frente, usamos a seguinte propriedade: log-retorno de k períodos, $r_t[k]$, do instante $t+1$ ao instante $t+k$, é dado por

$$r_t[k] = r_{t+1} + r_{t+2} + \cdots + r_{t+k},$$

de modo que

$$r_t[k]|\mathcal{F}_t \sim N(0, h_t[k]),$$

e $h_t[k] = k\hat{h}_t(1)$, ou seja, k vezes a previsão da volatilidade um passo à frente. Portanto, considerando o exemplo estimado, teríamos um VaR 95% para 10 dias de $1.65 \times 0.02651 \times \sqrt{10} = 0.1383228$ e uma perda de R$ 138,322.8.

Exemplo 2: Vale do Rio Doce e Itaú

Queremos usar abordagem RiskMetrics para calcular o VaR de uma carteira com R$ 500 mil em Vale e R$ 500 mil em Itaú.

Agora, no lugar do modelo de volatilidade somente para a Vale, consideramos

$$\gamma_{ij,t} = \lambda\gamma_{ij,t} + (1-\lambda)\gamma_{ij,t}r_{i,t-1}r_{j,t-1},$$

sendo i equivalente à Vale e j ao Itaú. O $\gamma_{ij,t}$ tem uma relação com correlação linear entre os ativos:

$$\rho_{ij} = \gamma_{ij,t}/\sigma^2_{ii,t}\sigma^2_{jj,t},$$

sendo $\sigma^2_{ii,t}$ a variância do ativo i, $\sigma^2_{jj,t}$ a variância do ativo j e $\gamma_{ij,t}$ a covariância entre eles.

O VaR da carteira então é dado por:

$$VaR_{carteira} = \sqrt{VaR_i^2 + VaR_j^2 + 2\rho_{ij}VaR_iVaR_j},$$

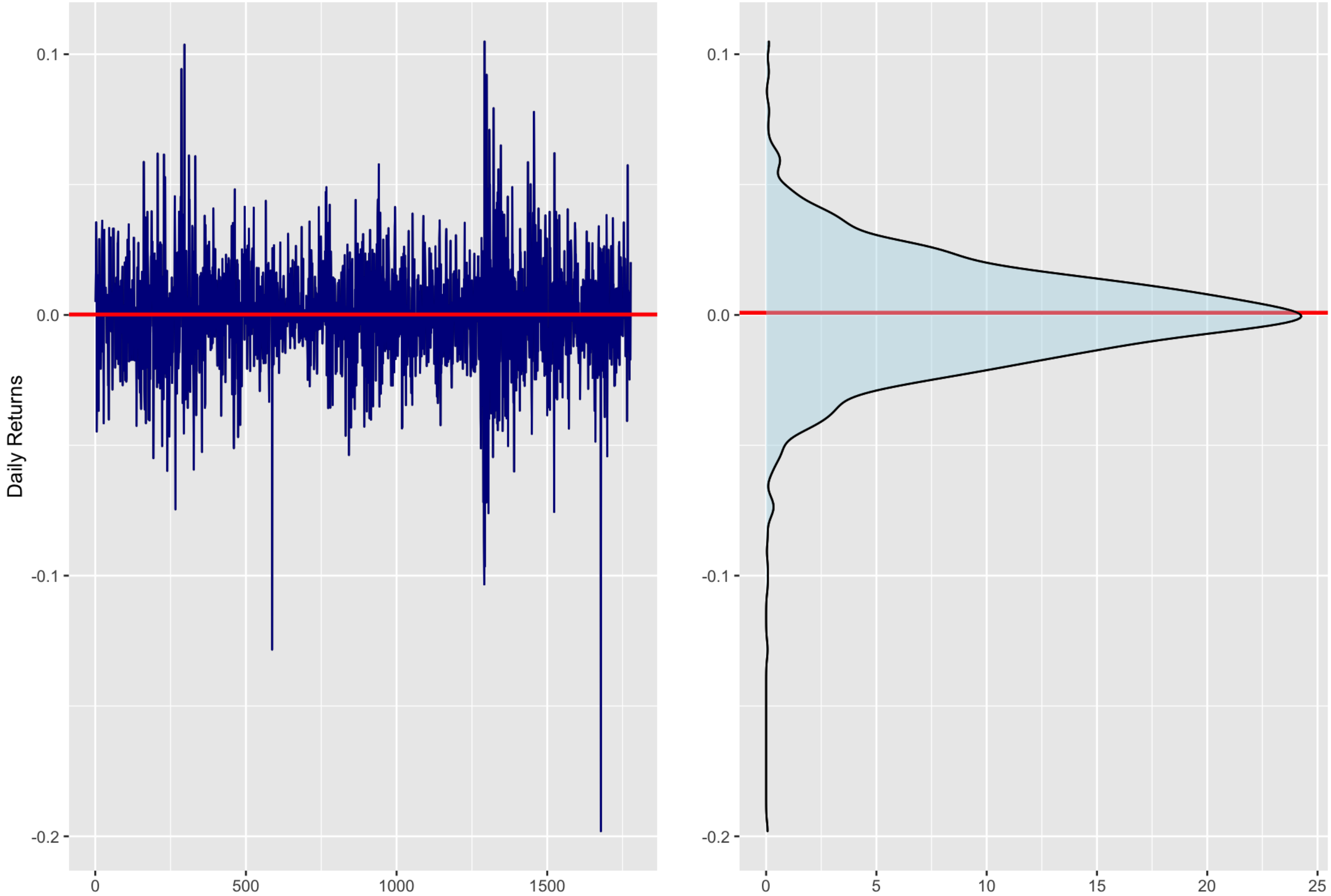
onde VaR_i e VaR_j são os valores em risco individuais.

Vamos então calcular a volatilidade condicional do Itaú, considerando as mesmas etapas anteriores:

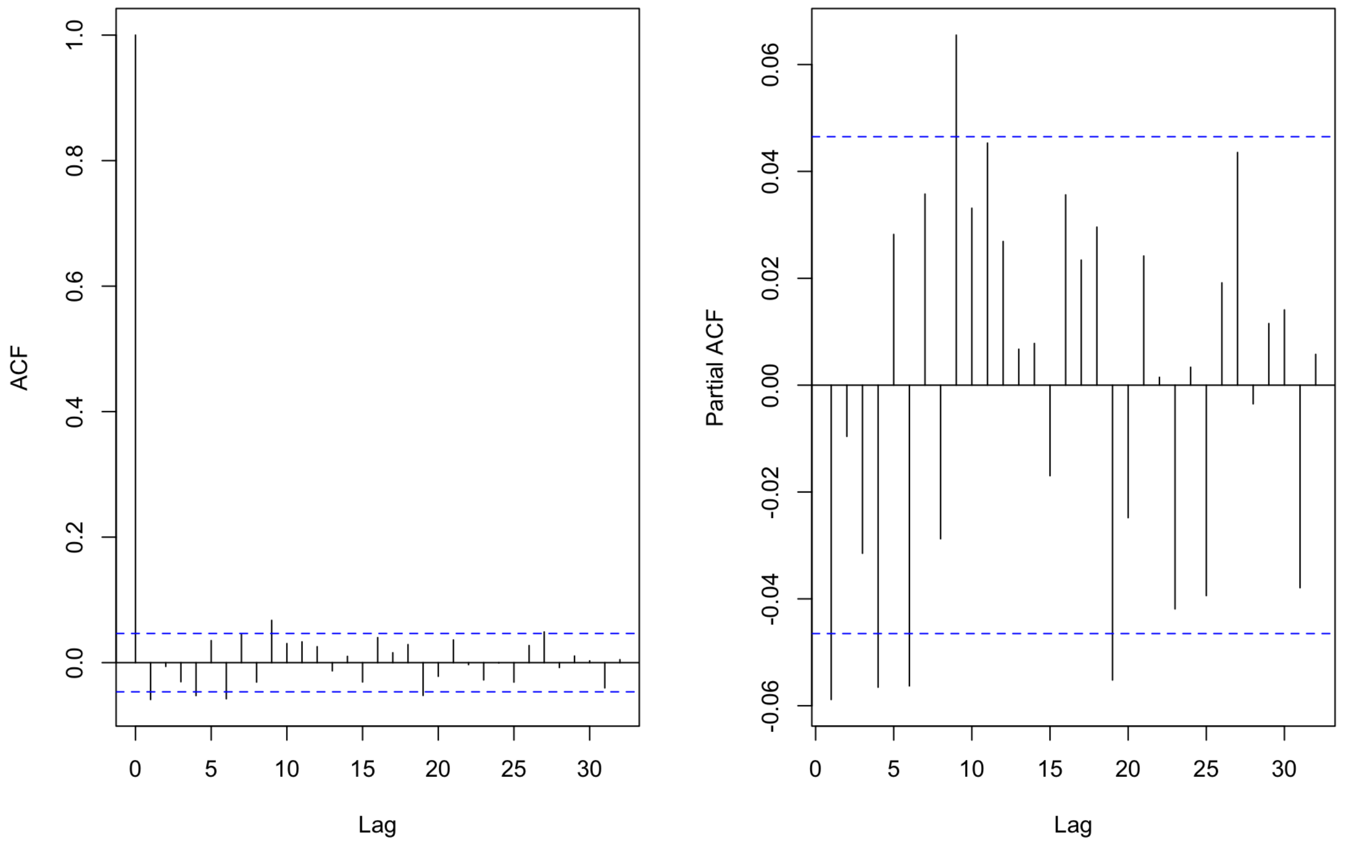
```
ticker_symbols <- c('ITUB4.SA')
start_date <- '2015-01-01'
end_date <- '2022-02-28'

# Getting data from APIs
getSymbols(ticker_symbols,
          from = start_date,
         to = end_date)
```

```
## [1] "ITUB4.SA"
```

```
rets2 <- diff(log(ITUB4.SA$ITUB4.SA.Close))[-1]

p1 = qplot(x = 1:length(rets2) , y = rets2 , geom = 'line') + geom_line(color = 'darkblue') +
    geom_hline(yintercept = mean(rets2) , color = 'red' , size = 1) +
    labs(x = '' , y = 'Daily Returns')

p2 = qplot(rets2 , geom = 'density') + coord_flip() + geom_vline(xintercept = mean(rets) , color = 'red'
         , size = 1) +
    geom_density(fill = 'lightblue' , alpha = 0.4) + labs(x = '')

grid.arrange(p1 , p2 , ncol = 2)
```




```
# acf graphs to identify lags
par(mfrow=c(1,2))
acf(rets2,main = "")
acf(rets2, main = "",type = "partial")
```




```
# checking non stationarity and auto arima suggestions
adf.test(rets2)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  rets2
## Dickey-Fuller = -10.495, Lag order = 12, p-value = 0.01
## alternative hypothesis: stationary
```

```
arma_model = auto.arima(rets2,
                        trace = F,
                        stepwise = F,
                        ic = 'aicc')
arma_model
```

```
## Series: rets2
## ARIMA(4,0,1) with zero mean
##
## Coefficients:
##           ar1      ar2      ar3      ar4    ma1
##       -0.8196  -0.0573  -0.0439  -0.0857  0.765
## s.e.   0.0741   0.0309   0.0306   0.0244  0.071
##
## sigma^2 = 0.000451:  log likelihood = 4326.13
## AIC=-8640.27   AICc=-8640.22   BIC=-8607.37
```

Nesse caso, o modelo para a média seria um ARMA(4,1) com média zero. Por ora, não usaremos esse resultado, uma vez que o RiskMetrics assume que o retorno é um ruído branco.

```
# Model specification
model.spec = ugarchspec(variance.model = list(model = 'iGARCH' , garchOrder = c(1 , 1)),mean.model = list
        (armaOrder = c(0, 0),include.mean=F))
model.fit3 = ugarchfit(spec = model.spec , data = rets2)
model.fit3@fit$matcoef
```

```
##            Estimate   Std. Error  t value     Pr(>|t|)
## omega  7.129818e-06 2.125919e-06 3.353758 7.972202e-04
## alpha1 9.769857e-02 1.317758e-02 7.413998 1.225686e-13
## beta1  9.023014e-01           NA       NA           NA
```

```
ugarchforecast(model.fit3,n.ahead=1,data=rets2)
```

```
##
## *------------------------------------*
## *       GARCH Model Forecast         *
## *------------------------------------*
## Model: iGARCH
## Horizon: 1
## Roll Steps: 0
## Out of Sample: 0
##
## 0-roll forecast [T0=2022-02-25]:
##     Series   Sigma
## T+1      0 0.02097
```

Na prática, usamos a correlação histórica como proxy para ρ_{ij}. Vamos então calcular o VaR da carteira, usando o fato de que o desvio padrão estimado para Vale foi de 0.02651 e para Itaú 0.02097 e de que a correlação amostral entre os retornos é de 0.3435571. Usando 95% de confiança, temos que para o dia seguinte:

$$VaR_{carteira} = \sqrt{(1.65 \times 0.02651)^2 + (1.65 \times 0.02097)^2 + 2 * 0.3435571 * (1.65 \times 0.02651) * (1.65 \times 0.02097)} = 0.06442395,$$

o que equivale a uma perda de R$ 64,423.95, considerando posição comprada. Note que se a correlação entre os ativos fosse negativa, teríamos um VaR da carteira menor que o calculado, gerando diversificação e um menor risco. Com isso, os gestores de carteira conseguem fazer *hedge*.

2. Abordagem econométrica

A abordagem baseia-se em modelos do tipo GARCH. Aqui não necessariamente se assume normalidade e retornos “white noise”, e além disso, não necessariamente se assume o modelo EWMA para a volatilidade ou um IGARCH(1,1).

Vimos que para fazer a previsão da volatilidade no instante t para os próximos k períodos era $h_t[k] = k\hat{h}_t(1)$. Ou seja, bastava saber a previsão um passo à frente.

No caso dos modelos GARCH genéricos, temos que:

$$r_t[k] = \hat{r}_t(1) + \ldots + \hat{r}_t(k),$$

e

$$h_t[k] = \hat{h}_t(k) + (1 + \psi_1)^2 \hat{h}(k-1) + \ldots + (\sum_{i=0}^{k-1} \psi_i)^2 \hat{h}_t(1),$$

onde ψ_i são parâmetros de uma representação MA(∞) para um processo ARMA.

Portanto, precisamos saber as previsões um passo à frente, dois passos à frente e assim por diante dos retornos e da volatilidade.

Além disso, assumindo distribuição Normal para os erros do modelo ARMA-GARCH, a distribuição condicional dos retornos no instante $t+1$, assumindo toda a informação disponível até t é

$$r_{t+1} | \mathcal{F} \sim N(\hat{r}_t(1), \hat{h}_t(1)),$$

onde $\hat{r}_t(1)$ é a previsão dos retornos um passo à frente e $\hat{h}_t(1)$ é a previsão da volatilidade. Os quantis 5% da distribuição (considerando posição comprada) são, portanto, $\hat{r}_t(1) - 1.65 \times \sqrt{\hat{h}_t(1)}$.

Assumindo distribuição t-Student para os erros, temos o quantil equivalente $\hat{r}_t(1) + t_v^*(p) \times \sqrt{\hat{h}_t(1)}$, onde $t_v^*(p)$ é o p-quantil da distribuição t-Student padrão com v graus de liberdade. No modelo ARMA-GARCH considerado, temos que usar a seguinte correção para o cálculo do VaR 1 passo à frente, de forma a “padronizar” o dado para uma t-Student padrão:

$$\hat{r}_t(1) + \frac{t_v^*(p) \times \sqrt{\hat{h}_t(1)}}{\sqrt{v/(v-2)}},$$

sendo que a estimativa do v é calculada na estimação via R.

Exemplo Vamos estimar um ARMA(0,4)+GARCH(1,1) com erros t-Student para o retorno da Vale, conforme exposto acima na aplicação da metodologia de Box-Jenkins:

```
# Model specification
model.spec = ugarchspec(
  mean.model = list(armaOrder = c(0,4),include.mean=F),
  variance.model = list(garchOrder=c(1,1)),
  distribution.model = "std"
  )
model.fit4 = ugarchfit(spec = model.spec , data = rets)
model.fit4@fit$matcoef
```

```
##             Estimate   Std. Error    t value     Pr(>|t|)
## ma1    -3.044499e-02 2.428690e-02 -1.2535562 2.100033e-01
## ma2    -2.772992e-02 2.420899e-02 -1.1454389 2.520274e-01
## ma3     2.751224e-03 2.409240e-02  0.1141947 9.090835e-01
## ma4    -2.290141e-02 2.315636e-02 -0.9889901 3.226680e-01
## omega   1.991598e-05 7.601758e-06  2.6199170 8.795118e-03
## alpha1  8.234284e-02 1.773110e-02  4.6439786 3.417629e-06
## beta1   8.947666e-01 2.271741e-02 39.3868256 0.000000e+00
## shape   6.237178e+00 8.606988e-01  7.2466441 4.272138e-13
```

```
ugarchforecast(model.fit4,n.ahead=1,data=rets)
```

```
##
## *------------------------------------*
## *       GARCH Model Forecast         *
## *------------------------------------*
## Model: sGARCH
## Horizon: 1
## Roll Steps: 0
## Out of Sample: 0
##
## 0-roll forecast [T0=2022-02-25]:
##        Series   Sigma
## T+1 -0.002382 0.02634
```

Por este modelo, teríamos um VaR 5% para o próximo dia de

```
# Model specification
fore = ugarchforecast(model.fit4,n.ahead=1,data=rets)
v = model.fit4@fit$matcoef["shape",][1]
r = fore@forecast$seriesFor
sigma = fore@forecast$sigmaFor# sqrt(h)
tc = qt(0.05,v)
tc
```

```
## [1] -1.930027
```

```
VaR = r+tc*sigma/sqrt((v/(v-2)))
-VaR*1000000
```

```
##      2022-02-25
## T+1   44286.92
```

perda de R$ 44,286.92 para o dia seguinte (contra R$ 43,734.41 do RiskMetrics).

Ou seja, cada abordagem tem seus prós e contras e gera um valor diferente. É preciso que o gestor de risco saiba diferenciar e escolher o método que considera mais adequado.

No caso de carteiras (múltiplas posições em diferentes ativos), há também a possibilidade de modelos GARCH multivariados, que veremos brevemente mais à frente.

Por fim, o VaR da abordagem ARMA-GARCH para k períodos à frente pode ser calculado usando a função `ARMAtoMA` do R (que estima os parâmetros ψ da representação MA infinita). Contudo, essa especificidade foge do programa do curso.

Notas

Os pacotes `rugarch` , `fGarch` e o `arch` (do Python) tem em suas funcionalidades o cálculo do VaR histórico, utilizando os percentis dos valores previstos. Contudo, para a predição do VaR, é necessário utilizar as "contas" manuais descritas nesse tutorial e/ou simulações de Monte Carlo. O pacote `rugarch` , por exemplo, possui as funções *ugarchpath* e *ugarchsim* que podem ser utilizadas para este objetivo.

Referências

Para mais informações, ver:

- Tsay, Ruey S. (2002). *Analysis of Financial Time Series*, cap. 7.
- Morettin, Pedro A. (2008). *Econometria Financeira*, cap. 6.
- https://www.investopedia.com/terms/v/var.asp (https://www.investopedia.com/terms/v/var.asp)