Modelagem em Séries Temporais Financeiras

Anna Beatriz Ayumi Sato, Daniel Vasconcellos Figueiredo Falbel, Júlia Ferreira Aquino Silva

A série de dados foi obtida pelo site da Vale do Rio Doce por meio do seguinte link: http://www.vale.com/brasil/PT/investors/equity-debt/stock-price-history/Paginas/default.aspx. No arquivo temos a informação do preço de fechamento e preço de fechamento ajustado dos valores das ações da Vale desde o dia 3 de Janeiro de 2000 até o dia 30 de Outubro de 2014.

Leitura no `R`

A leitura da série no `R` foi feita utilizando o comando a seguir;

```
serie <- read.table("dados/serie_vale3.txt", header = T, sep = "\t", dec = ",")
```

Para transformar a coluna 'data' em um formato de data, utilizamos o comando a seguir:

```
serie$data <- as.Date(serie$data, format = "%d/%m/%Y")
```

Neste momento também calculamos os log-retornos r_t da série que são dados por:

$$r_t = \log\left(\frac{P_t}{P_{t-1}}\right)$$

No `R` utilizamos o comando abaixo:

```
serie$r <- c(NA, log(serie$fechamento.ajustado[2:3668]/
                 serie$fechamento.ajustado[1:3667]))
```

Análise descritiva

Como análise descritiva da série fizemos os gráficos a seguir. Primeiro observamos os valores verdadeiros da série e em seguida observamos o comportamento do log-retorno.

Na figura 1 vemos que a série teve uma tendência de crescimento acelerado até o ano de 2008. Em 2009 teve uma forte queda, mas logo os valores voltaram a subir. Desde o ano de 2010 a série apresenta uma tendência de queda nos preços dos ativos.

```
library(ggplot2)
qplot(x = data, y = fechamento.ajustado, geom = "line", data = serie) +
  xlab("Data") +
  ylab("Valor de Fechamento Ajustado (R$)") +
  theme(axis.title = element_text(size = rel(0.7)))
```

A figura 2, que mostra o log-retorno, mostra que a série não apresenta tendência e que a média dos log-retornos parece ser próxima de zero. No ano de 2009, a série apresenta um grande distúrbio causado pela grande variação dos preços naquele período.

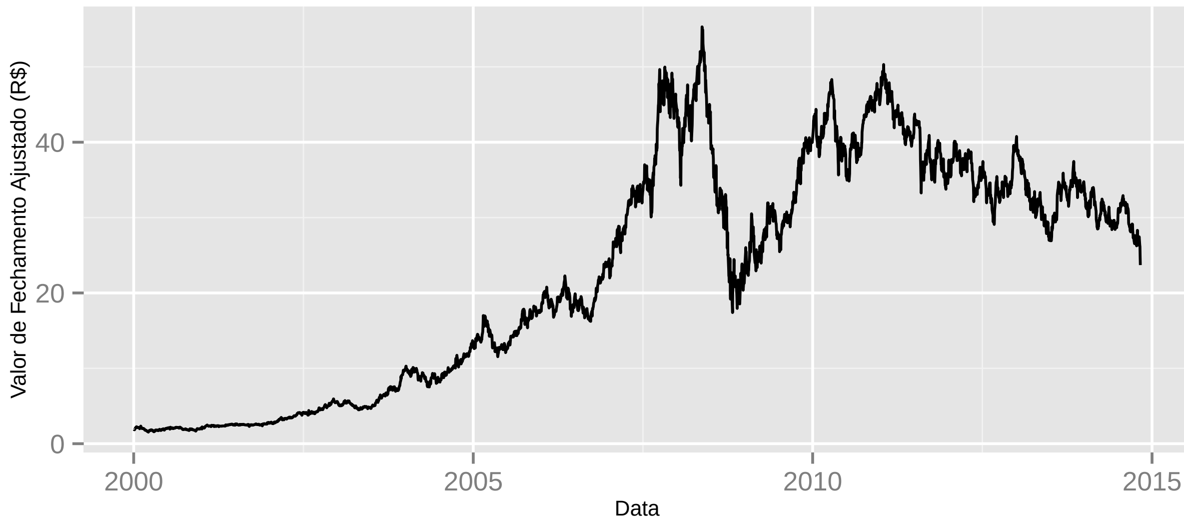

Figure 1: Gráfico dos preços da série VALE3 do ano 2000 até o final de outubro de 2014

```
qplot(x = data, y = r, geom = "line", data = serie) +
  xlab("Data") +
  ylab("Log-Retorno do Valor de Fechamento Ajustado") +
  theme(axis.title = element_text(size = rel(0.7)))
```

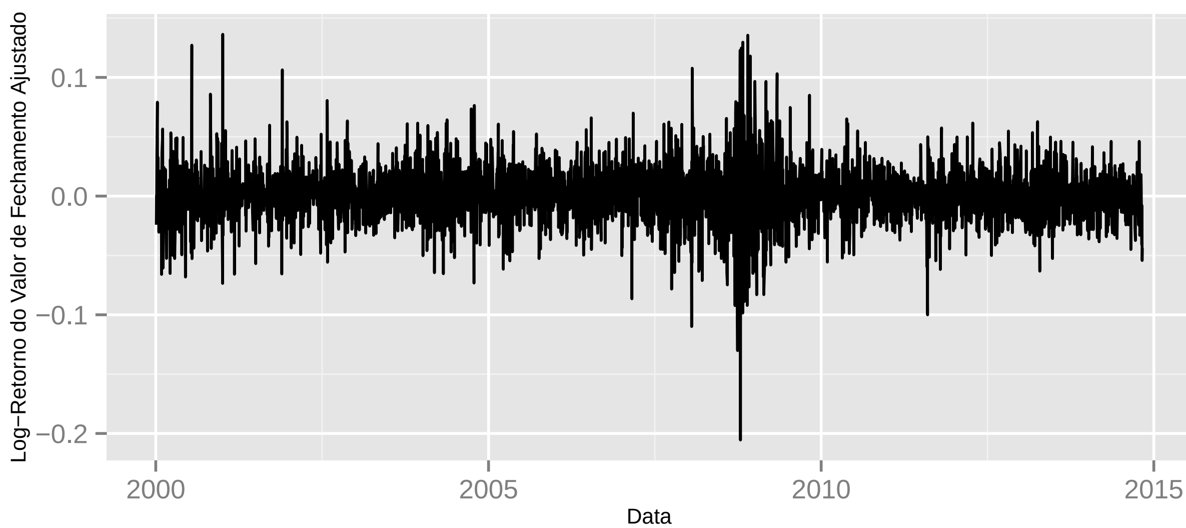

Figure 2: Gráfico dos log-retorno dos preços da série VALE3 do ano 2000 até o final de outubro de 2014

A figura 3 mostra a distribuição dos log-retornos. Ela é aparentemente simétrica em torno do zero e parece ser próxima da distribuição Normal, no entanto ela apresenta uma cauda que é possívelmente mais pesada do que a da Normal.

```
ggplot(serie, aes(r)) + geom_histogram(aes(y = ..density..), colour = "white") +
  geom_density() + xlab("Log-Retorno") + ylab("Densidade")
```

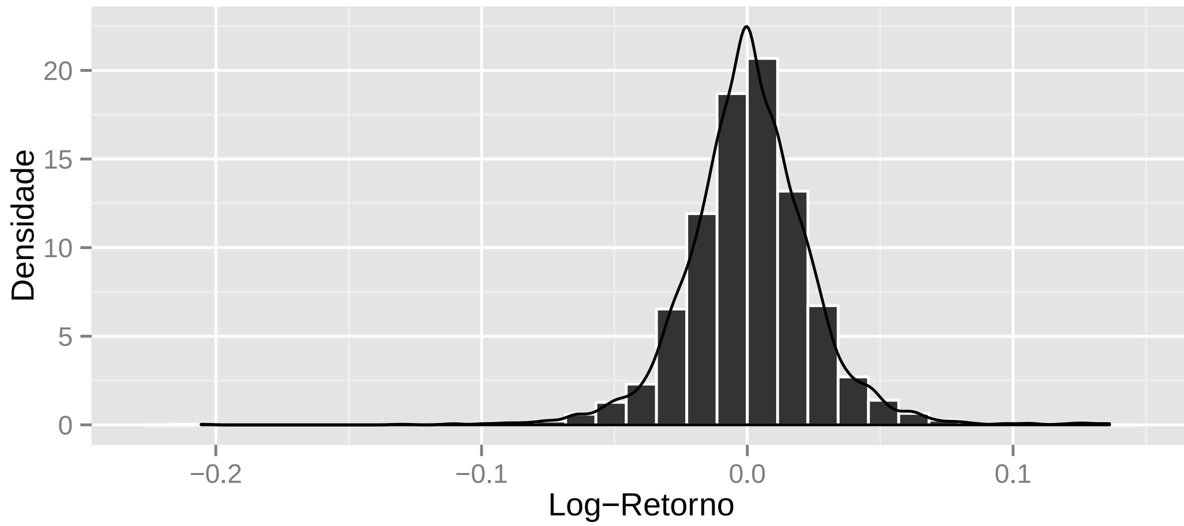

Figure 3: Gráfico da distribuição dos log-retornos

Cálculo do VaR pelo método RiskMetrics

O cálculo do VaR usando o `R` pode ser feito usando uma função feita pelo Ruey S. Tsay para o seu livro "An Introduction to Analysis of Financial Data with R" que pode ser obtida neste link: http://faculty.chicagobooth.edu/ruey.tsay/teaching/introTS/RMfit.R. O código utilizado está abaixo.

```
source("funcoes/RMfit.R")
mm <- RMfit(serie$r[-1])
```

```
## [1] -8755
##   0:    -8754.8059: 0.900000
##   3:    -8790.9781: 0.956917
##   6:    -8791.0290: 0.958553
##
## Coefficient(s):
##         Estimate  Std. Error  t value   Pr(>|t|)
## alpha 0.95855303  0.00505892  189.478 < 2.22e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##  Volatility prediction:
##      Orig   Vpred
## [1,] 3667 0.02311
##
##  Risk measure based on RiskMetrics:
##       prob     VaR      ES
## [1,] 0.950 0.03802 0.04768
## [2,] 0.990 0.05377 0.06160
## [3,] 0.999 0.07142 0.07782
```

Para começar vamos estimar o modelo IGARCH(1,1) para obter uma estimativa do parâmetro α. Nesse caso obtivemos o valor: 0.9586. Também temos que a volatilidade ajustada para a última observação foi 0.0218 e o último log-retorno observado foi -0.0438 Então podemos estimar a volatilidade um passo a frente usando a seguinte fórmula:

$$\sigma^2_{3669} = \hat{\alpha}(\hat{\sigma_{3668}})^2 + (1-\hat{\alpha})(r_{3668})^2$$

Então o valor estimado foi:

$$\sigma^2_{3669} = 0,958553 * 0,000474525 + (1-0,95855)(0,001914447) = 0,0005342054$$

Em seguida utilizando o método RiskMetrics com confiança de 95% temos que o VaR é dado por:

$$VaR[1] = 10.000 * 1,65 * \sigma_{3669} = R\$390,6077$$

O VaR[5] é calculado multiplicando o $VaR[1] * \sqrt{5}$ então temos que $VaR[5] = R\$873.4254$.

Como consideramos a distribuição Normal que é simetrica, não existe diferença nas estimativas do VaR para posições compradas e vendidas.

Cálculo utilizando a teoria dos valores extremos

No caso do TVE ficou:

Posição vendida: Var[1] = R$341,9102 Var[5] = R$421,0283

Posição comprada: Var[1] = R$320,2746 Var[5] = R$270,5744

O código em `R` utilizado para o cálculo está abaixo:

```
r<-t[-seq(1,13,1),1]

mx<-1:174
mn<-1:174

aux=1
for (i in 1:174){

mx[i]<-max(r[aux:(aux+20)])
mn[i]<-min(r[aux:(aux+20)])
aux=aux+21

}

library(ismev)

#posicao vendida
gev.fit(mx)
VaR= 0.03527705-(
  (0.01468186/0.12933211)*
    (1-
            (
                (-21*log(.95))^(-0.12933211)
            )
        )
    )
VaR*10000
```

```r
(5^0.12933211)*VaR*10000


#posicao comprada
gev.fit(-mn)
VaR= -0.03304296+
    (
        (0.01371588/0.10477671)*
        (1-
            (
                (-21*log(.95))^(-0.10477671)
            )
        )
    )
abs(VaR*10000)
abs((5^(-0.10477671))*VaR*10000)
```

Cálculo pelo método dos quantis-empíricos

Os valores do quantis empíricos são:

VaR(1) para posição vendida = q(0,95) = 0.03732015 VaR(1) para posição comprada = q(0,05) = -0.03505672

Como o valor é 10000, fica:

Posição vendida: VaR(1) = R$373,20

Posição comprada: VaR(1) = R$350,57

Para o cálculo utilizamos o seguinte código no `R`:

```r
varqe = 10000*quantile(na.omit(serie$r), probs=c(0.05, 0.95))
```

Para esse método, não é possível calcular o VaR cinco passos a frente.

Cálculo utilizando o ARMA + GARCH(1,1)

Ajuste do ARMA:

```r
acf(serie$r[-1], main="Autocorrelação dos log-retornos da Vale")
```

```r
pacf(serie$r[-1], main="Autocorrelação parcial dos log-retornos da Vale")
```

Vemos que a autocorrelação da série de log retornos da Vale é não significativa a partir do lag 1, assim como a autocorrelação parcial. Assim, não é necessário ajustar um modelo ARMA para a série. Ajustando um modelo GARCH(1,1) para a série:

```r
library(fGarch)
```

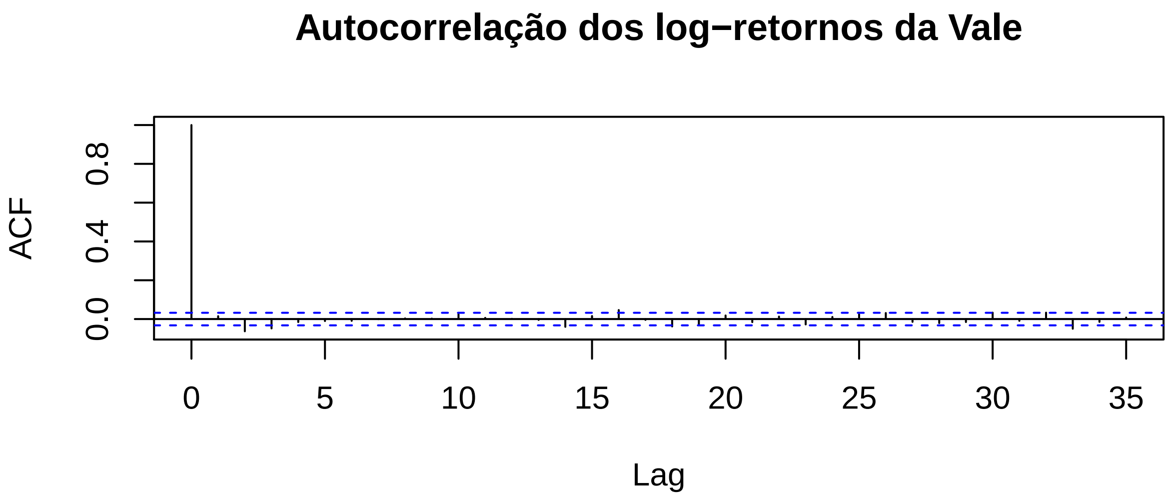

Figure 4: plot of chunk unnamed-chunk-11

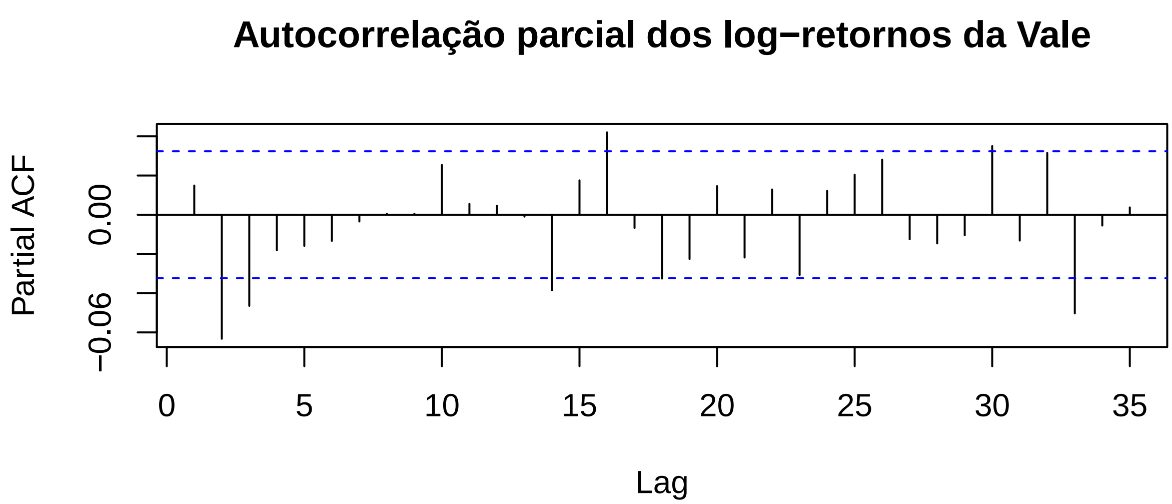

Figure 5: plot of chunk unnamed-chunk-11

```
## Loading required package: timeDate
## Loading required package: timeSeries
## Loading required package: fBasics
##
##
## Rmetrics Package fBasics
## Analysing Markets and calculating Basic Statistics
## Copyright (C) 2005-2014 Rmetrics Association Zurich
## Educational Software for Financial Engineering and Computational Science
## Rmetrics is free software and comes with ABSOLUTELY NO WARRANTY.
## https://www.rmetrics.org --- Mail to: info@rmetrics.org
```

```
mnormal = garchFit(~garch(1,1),data=serie$r[-1],trace=F)
summary(mnormal)
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~garch(1, 1), data = serie$r[-1], trace = F)
##
## Mean and Variance Equation:
##  data ~ garch(1, 1)
## <environment: 0x5d9a130>
##  [data = serie$r[-1]]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##         mu       omega      alpha1       beta1
## 1.0637e-03  1.4945e-05  7.2863e-02  8.9916e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu     1.064e-03   3.358e-04    3.167 0.001539 **
## omega  1.495e-05   4.011e-06    3.726 0.000195 ***
## alpha1 7.286e-02   1.214e-02    6.003 1.93e-09 ***
## beta1  8.992e-01   1.777e-02   50.605  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  8827    normalized:  2.407
##
## Description:
##  Tue Nov  4 16:47:32 2014 by user:
##
##
## Standardised Residuals Tests:
```

```
##                                  Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  275.5     0
##  Shapiro-Wilk Test  R    W      0.9899    1.667e-15
##  Ljung-Box Test     R    Q(10)  16.23     0.09327
##  Ljung-Box Test     R    Q(15)  27.41     0.02556
##  Ljung-Box Test     R    Q(20)  31.2      0.05254
##  Ljung-Box Test     R^2  Q(10)  22.82     0.01143
##  Ljung-Box Test     R^2  Q(15)  25.44     0.04437
##  Ljung-Box Test     R^2  Q(20)  29.95     0.07075
##  LM Arch Test       R    TR^2   22.01     0.03741
##
## Information Criterion Statistics:
##    AIC    BIC    SIC   HQIC
## -4.812 -4.805 -4.812 -4.810
```

```r
mtstud = garchFit(~garch(1,1),data=serie$r[-1],trace=F, cond.dist="std")
summary(mtstud)
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~garch(1, 1), data = serie$r[-1], cond.dist = "std",
##     trace = F)
##
## Mean and Variance Equation:
##  data ~ garch(1, 1)
## <environment: 0x63f11f0>
##  [data = serie$r[-1]]
##
## Conditional Distribution:
##  std
##
## Coefficient(s):
##         mu       omega      alpha1       beta1       shape
## 7.5673e-04  9.3896e-06  5.4127e-02  9.2839e-01  7.1913e+00
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu     7.567e-04   3.239e-04    2.336  0.01948 *
## omega  9.390e-06   3.046e-06    3.083  0.00205 **
## alpha1 5.413e-02   1.008e-02    5.371 7.81e-08 ***
## beta1  9.284e-01   1.385e-02   67.054  < 2e-16 ***
## shape  7.191e+00   8.189e-01    8.782  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  8889    normalized:  2.424
##
```

```
## Description:
##  Tue Nov  4 16:47:33 2014 by user:
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  320.8     0
##  Shapiro-Wilk Test  R    W      0.9891    3.357e-16
##  Ljung-Box Test     R    Q(10)  15.87     0.1035
##  Ljung-Box Test     R    Q(15)  26.7      0.03128
##  Ljung-Box Test     R    Q(20)  30.71     0.05917
##  Ljung-Box Test     R^2  Q(10)  33.48     0.0002258
##  Ljung-Box Test     R^2  Q(15)  36.26     0.001623
##  Ljung-Box Test     R^2  Q(20)  41.23     0.003477
##  LM Arch Test       R    TR^2   31.72     0.001526
##
## Information Criterion Statistics:
##    AIC    BIC    SIC   HQIC
## -4.845 -4.837 -4.845 -4.842
```

Escolheremos o modelo que apresenta o menor AIC. O AIC do modelo com distribuição normal é -4.811989. Já o AIC do modelo com distribução t-student é -4.845354. Escolhemos, portanto, o modelo com distriuição condicional t-student com 7 graus de liberdade.

O modelo fica:

$$r_t = 7,567 \times 10^{-4} + a_t, \quad a_t = \sigma_t \varepsilon_t, \quad \varepsilon_t \sim t_7$$
$$\sigma_t^2 = 9,39 \times 10^{-6} + 0,05413 a_{t-1}^2 + 0,9284 \sigma_{t-1}^1$$

Escolhido o modelo, obtemos o VaR:

Inicialmente, calculamos a previsão da série de log-retornos 1 e 5 passos a frente.

```
predict(mtstud, n.ahead=5)
```

```
##   meanForecast meanError standardDeviation
## 1    0.0007567   0.02592           0.02592
## 2    0.0007567   0.02587           0.02587
## 3    0.0007567   0.02583           0.02583
## 4    0.0007567   0.02578           0.02578
## 5    0.0007567   0.02574           0.02574
```

Agora, calcularemos o Var utilizando a função abaixo, desenvolvida por Ruey S. Tsay:

```
"RMeasure" <- function(mu,sigma,cond.dist="norm",df=0){
  # calculate VaR and ES for a specified conditional distribution
  # p = 0.05, 0.01, 0.001
  #
  # cond.dist = "norm", "t", "std"
  prob=c(0.95,0.05)
  if(cond.dist=="norm"){
    q1=qnorm(prob)
    d1=dnorm(q1)
```

```r
    VaR=mu+q1*sigma
    ES=mu+d1/(1-prob)*sigma
    tt=cbind(prob,VaR,ES)
  }
  #
  if(cond.dist=="std"){
    library(fGarch)
    if(df < 2.001)df=2.01
    q1=qstd(prob,nu=df)
    d1=dstd(q1,nu=df)
    VaR=mu+q1*sigma
    ES=mu+sigma*(d1/(1-prob))*(((df-2)+q1^2)/(df-1))
    tt=cbind(prob,VaR,ES)
  }
  #
  if(cond.dist=="t"){
    if(df < 2.01)df=2.01
    q1=qt(prob,df)
    d1=dt(q1,df)
    #VaR=mu+q1*sigma
    VaR=mu+q1*sigma/sqrt(df/(df-2))
    ES=mu+sigma/sqrt(df/(df-2))*(d1/(1-prob))*((df+q1^2)/(df-1))
    tt=cbind(prob,VaR,ES)
  }
  cat("\n Risk Measures for selected probabilities: \n")
  print(tt)

  RMeasure <- list(results=tt)
}
```

Calculando o VaR(1):

```r
RMeasure(0.0007567318,0.02591705)
```

```
##
##  Risk Measures for selected probabilities:
##      prob      VaR      ES
## [1,] 0.95  0.04339 0.05422
## [2,] 0.05 -0.04187 0.00357
```

Assim, o VaR(1) para posição vendida é 0.04338649, e para posição comprada é -0.04187302.

Portanto:

$$VaR(1)_{vendida} = 10000 * 0.04338649 = R\$433,86$$
$$VaR(1)_{comprada} = 10000 * 0.04187302 = R\$418,73$$

Calculando o VaR[5]:

```r
RMeasure(0.0007567318,0.02573931)
```

```
##
##  Risk Measures for selected probabilities:
##      prob      VaR       ES
## [1,] 0.95  0.04309 0.053850
## [2,] 0.05 -0.04158 0.003551
```

Assim, o VaR[5] para posição vendida é 0.04309413, e para posição comprada é -0.04158067.

Portanto:

$$VaR[5]_{vendida} = 10000 * 0.04309413 = R\$430,94$$
$$VaR[5]_{comprada} = 10000 * 0.04158067 = R\$415,81$$