

Using Simulation to Estimate VaR and ES (Aula 4)

Ivanildo Batista

1 de marco de 2021

Aula 4 - Semana 2

Usando simulacao para estimar o VaR e o ES

Para estimar o quantil alpha de uma distribuicao podemos simular alguns dados dessa distribuicao e tomar o quantil alpha dos dados simulados.

O VaR e o quantil alpha de uma distribuicao, entao este metodo de simulacao deve funcionar. Vamos supor que queremos calcular o VaR no intervalo de confianca de 95%. Digamos que o retorno vem de uma distribuicao normal (nossa hipotese ate agora). Pode-se simular um amostra de 10.000 resultados da distribuicao normal e quinto percentil dos dados simulados e uma estimativa do VaR.

Uma vez que tenha-se estimado o VaR usamos dados simulados, tambem pode-se encontrar o deficit esperado com bastante facilidade. Pega-se os dados simulados e matenhamos apenas os resultados que sao os piores do que o VaR estimado; e a media deles e uma estimativa do deficit esperado. Se o tamanho amostral da simulacao aumentar, essas duas estimativas devem ser muito proximas dos valores reais. O metodo de simulacao e uma boa maneira de verificar nossas formulas quanto ao valor em risco e ao deficit esperado.

Vamos desenhar uma amostra com 100.000 resultados da distribuicao normal, com media μ e desvio padrao σ . O VaR e o quantil alpha desses 100.000 resultados e o ES e a media dos resultados que sao piores do que o VaR ou pior 100.000 x alpha resultados.

Vamos realizar os mesmos procedimentos das aulas anteriores

```
library(quantmod)
```

```
## Loading required package: xts
```

```
## Loading required package: zoo
```

```
##  
## Attaching package: 'zoo'
```

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

```
## Loading required package: TTR
```

```
## Registered S3 method overwritten by 'quantmod':
##   method           from
##   as.zoo.data.frame zoo
```

```
wilsh = getSymbols("WILL5000IND", src = "FRED", auto.assign = FALSE)
```

```
## 'getSymbols' currently uses auto.assign=TRUE by default, but will
## use auto.assign=FALSE in 0.5-0. You will still be able to use
## 'loadSymbols' to automatically load data. getOption("getSymbols.env")
## and getOption("getSymbols.auto.assign") will still be checked for
## alternate defaults.
##
## This message is shown once per session and may be disabled by setting
## options("getSymbols.warning4.0"=FALSE). See ?getSymbols for details.
```

```
wilsh = na.omit(wilsh)
wilsh = wilsh['1979-12-31/2017-12-31']
names(wilsh) = "TR"
```

Criando os retornos diários

```
logret = apply.daily(wilsh,sum)
```

Media amostral

```
mu = round(mean(logret),8)
mu
```

```
## [1] 32.669
```

Desvio padrao amostral

```
sig = round(sd(logret),8)
sig
```

```
## [1] 28.4018
```

SIMULACAO COM BASE NA DISTRIBUICAO NORMAL

```
alpha = 0.05
set.seed(123789)
rvec1 = rnorm(100000,mu,sig)
VaR1 = quantile(rvec1, alpha)
ES1=mean(rvec1[rvec1<VaR1])

VaR1
```

```
##      5%  
## -14.28874
```

```
ES1
```

```
## [1] -26.47909
```

SIMULACAO COM BASE NA DISTRIBUICAO DOS DADOS (sem supor normalidade nos dados)

```
alpha = 0.05  
set.seed(123789)  
rvec2 = sample(as.vector(logret),100000,replace = TRUE)  
VaR2 = quantile(rvec2, alpha)  
ES2=mean(rvec2[rvec2<VaR2])
```

```
VaR2
```

```
##      5%  
##  2.45
```

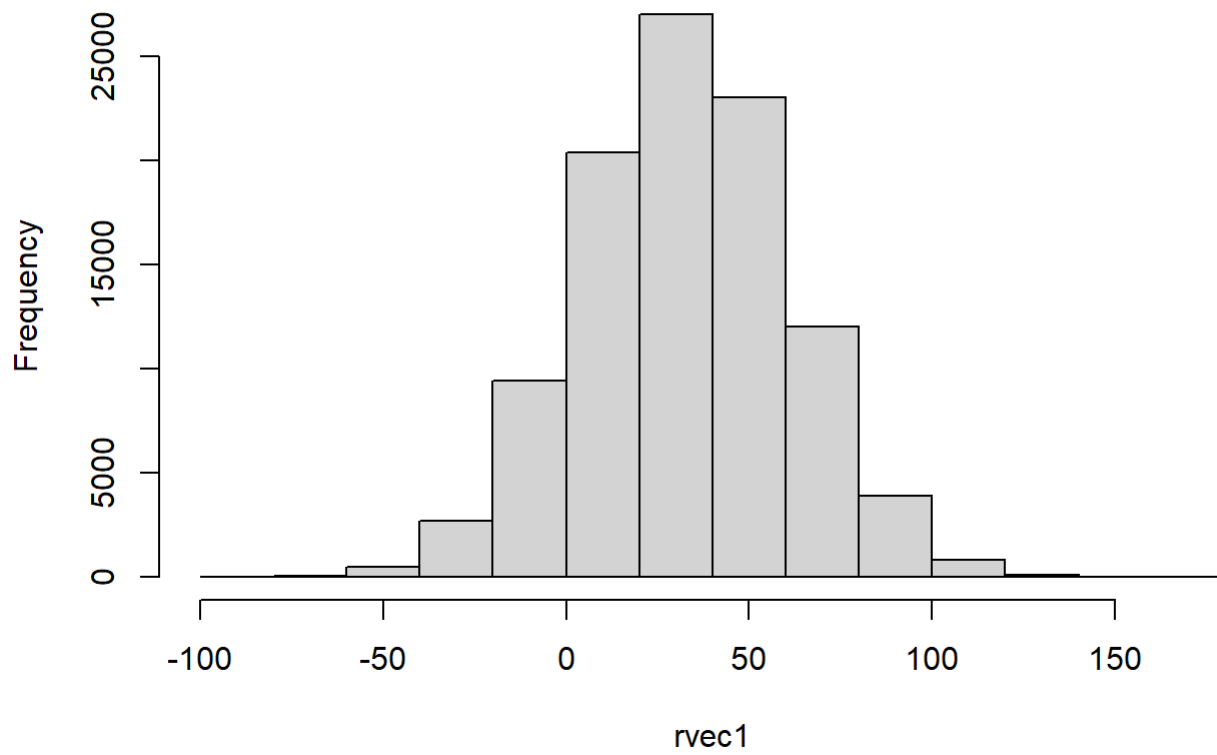
```
ES2
```

```
## [1] 2.200221
```

Distribuicao dos dados

```
hist(rvec1)
```

Histogram of rvec1



```
hist(rvec2)
```

Histogram of rvec2

