# Equações em diferenças no R

*Ivette Luna*

*30 de abril de 2019*

## Contents

# Equação de ordem dois com raízes reais e diferentes

Seja a equação em diferenças

$$y_t - 3y_{t-1} + 2y_{t-2} = 0$$

com $y_1 = 1$ e $y_2 = -2$.

1. Determine o polinômio característico e as suas raízes;
2. Determine as constantes arbitrárias;
3. Determine a trajetória temporal.

## Solução

Para o primeiro item, precisamos usar a função `polyroot`:

```
library(ggplot2)
library(limSolve)
```

```
##
## Attaching package: 'limSolve'

## The following object is masked from 'package:ggplot2':
##
##     resolution
```

```
library(matlib) # para o ginv
```

```
##
## Attaching package: 'matlib'

## The following object is masked from 'package:limSolve':
##
##     Solve
```

```
library(MASS) # para o ginv


# Exercicio 1
# p(x) = x_t -3*x_t-1 + 2*Xt-2


n = 10 # numero de iteracoes - periodos de tempo
y = rep(0, n) # serie a gerar
y[1] = 1 # condicoes iniciais
y[2] = -2

a2 = 1
a1 = -3
a0 = 2
coefs = c(a0, a1, a2)


roots <- polyroot( coefs )
roots
```

```
## [1] 1+0i 2-0i
```

```r
#zapsmall(roots)

# calculando o discriminante
delta = a1^2 -4*a2*a0
delta
```

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

Logo, a equação possui uma raíz maior que um em módulo e ambas as raízes são positivas: a série temporal resultante srá explosiva e com oscilações.

Veja também o sinal dos coeficientes. Por descartes, podemos deduzir, antes de qualquer cálculo, a existência de duas raízes positivas, descartando assim a existência de oscilações na série.

Para o segundo item, podemos usar sistemas lineares com duas incógnitas:

```r
lambda = Re(roots)
lambda
```

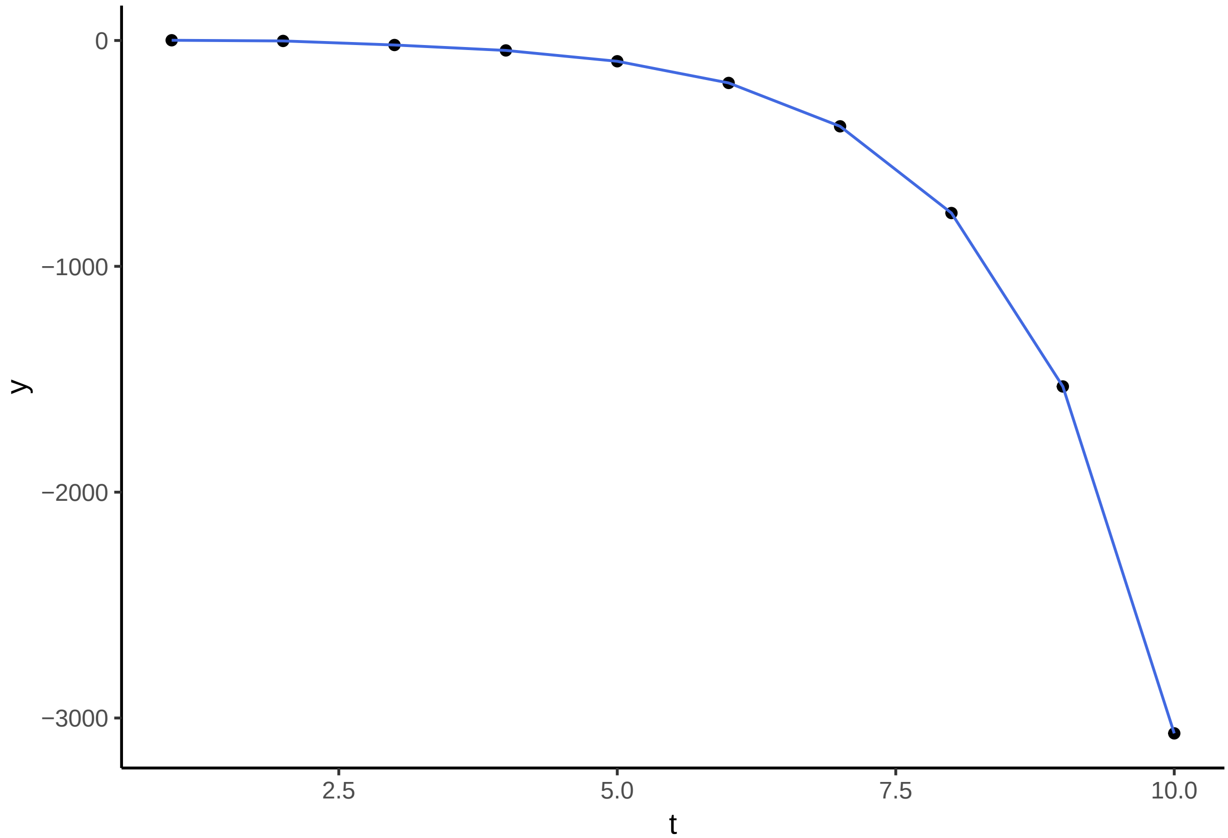
```
## [1] 1 2
```

```r
A = matrix( c(1, lambda[1], 1, lambda[2]), 2  )
B = c(  y[1], y[2]  )

X = ginv(A)%*%B

X # [A1, A2]
```

```
##      [,1]
## [1,]    4
## [2,]   -3
```

Para determinar a trajetória temporal, precisamos da forma funcional ou da equação em diferenças:

```r
A1 = X[1]

A2 = X[2]

n = 10

for (t in 3:n){

  y[t] = A1*lambda[1]^t + A2*lambda[2]^t

}

series = data.frame(t=1:n, y)

ggplot(series, aes(x=t,  y=y)) +

  geom_point(color="black") +

  geom_line(color = "Royalblue") +

  theme_classic()
```




### Analisando o exercício proposto

Para o caso da equação em diferenças

$$y_t - 0.25y_{t-2} = 0$$

com $y_1 = 1$ e $y_2 = 3$, temos que:

```
# Exercicio 2
a2 = 1
a1 = 0
a0 = -.25

y = rep(0, n) # serie a gerar
y[1] = 1 # condicoes iniciais
y[2] = 3


coefs = c(a0, a1, a2)
delta = a1^2 -4*a2*a0
delta
```

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

```
roots <- polyroot( coefs )
roots
```

```
## [1]  0.5+0i -0.5+0i
```

## Equação em diferenças com raízes reais e iguais

Seja a equação em diferenças

$$y_t - 6y_{t-1} + 9y_{t-2} = 0$$

com $y_1 = 1$ e $y_2 = 3$.

```
a2 = 1
a1 = -6
a0 = 9

y = rep(0, n) # serie a gerar
y[1] = 1 # condicoes iniciais
y[2] = 3


coefs = c(a0, a1, a2)
delta = a1^2 -4*a2*a0
delta
```

```
## [1] 0
```

```
roots <- polyroot( coefs )
roots
```

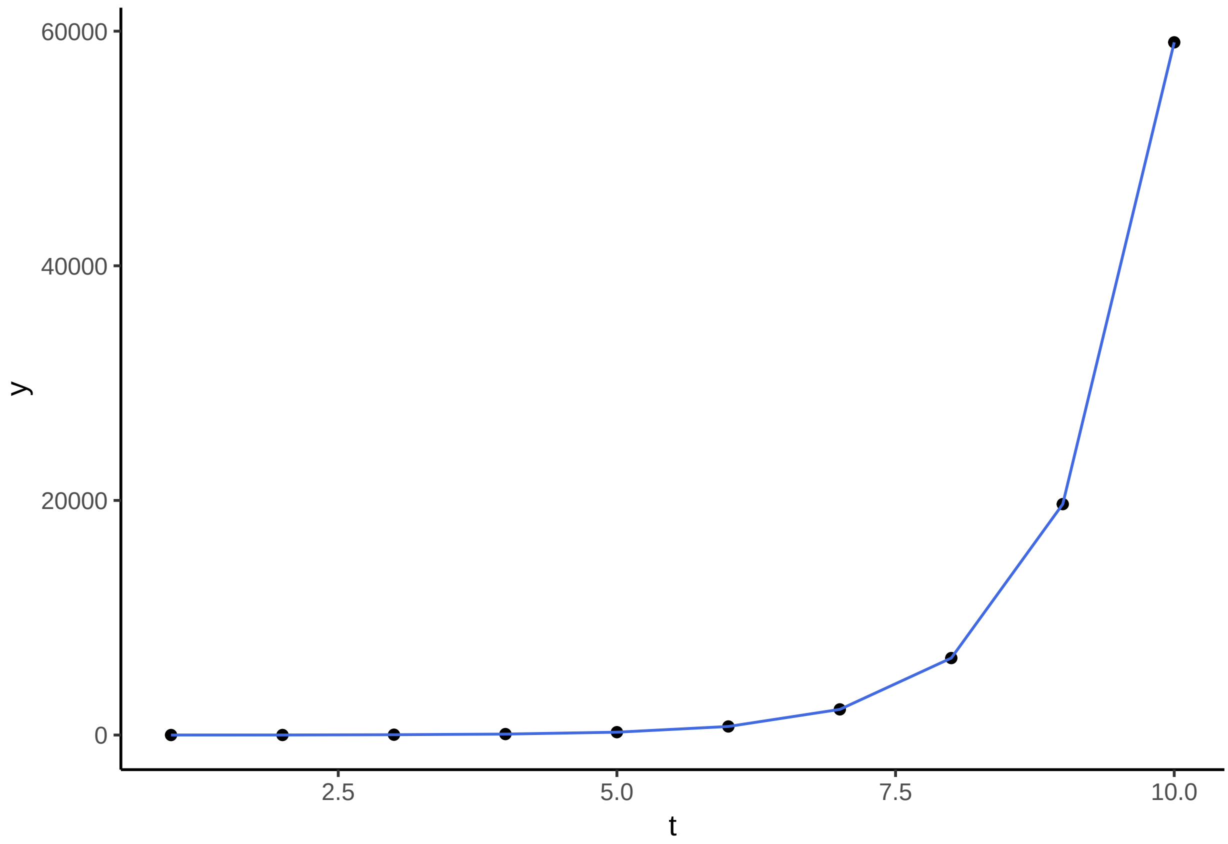
```
## [1] 3-0i 3+0i
```

```
if (delta==0) {# raizes reais e iguais
        print('Caso 2: Raízes reais e iguais')

        lambda=Re(roots[1])

        # identificando as constantes arbitrarias

        # yt = A1*lambda1^t+a2*t*lambda2^t

        # identificando as constantes arbitrarias
        A1 = y[1]
        A2 = y[2]/lambda-A1
        for (t in 3:n){
                y[t] = A1*lambda^t + A2*t*lambda^t # trajetoria temporal
        } # end for

} # end if
```

```
## [1] "Caso 2: Raízes reais e iguais"
```

```
series = data.frame(t=1:n, y)

ggplot(series, aes(x=t,  y=y)) +

  geom_point(color="black") +
```

```
  geom_line(color = "Royalblue") +

  theme_classic()
```




## O caso de raízes complexas

Precisamos de utilizar Euler e Moivre para reescrever a solução homogênea.

Se, por exemplo,

$$y_t - y_{t-1} + \frac{5}{4}y_{t-2} = 0$$

com $y_1 = 1$ e $y_2 = 3$.

```
#n = 50

a2 = 1
a1 = -1
a0 = 5/4

y = rep(0, n) # serie a gerar
y[1] = 1 # condicoes iniciais
y[2] = 3
```

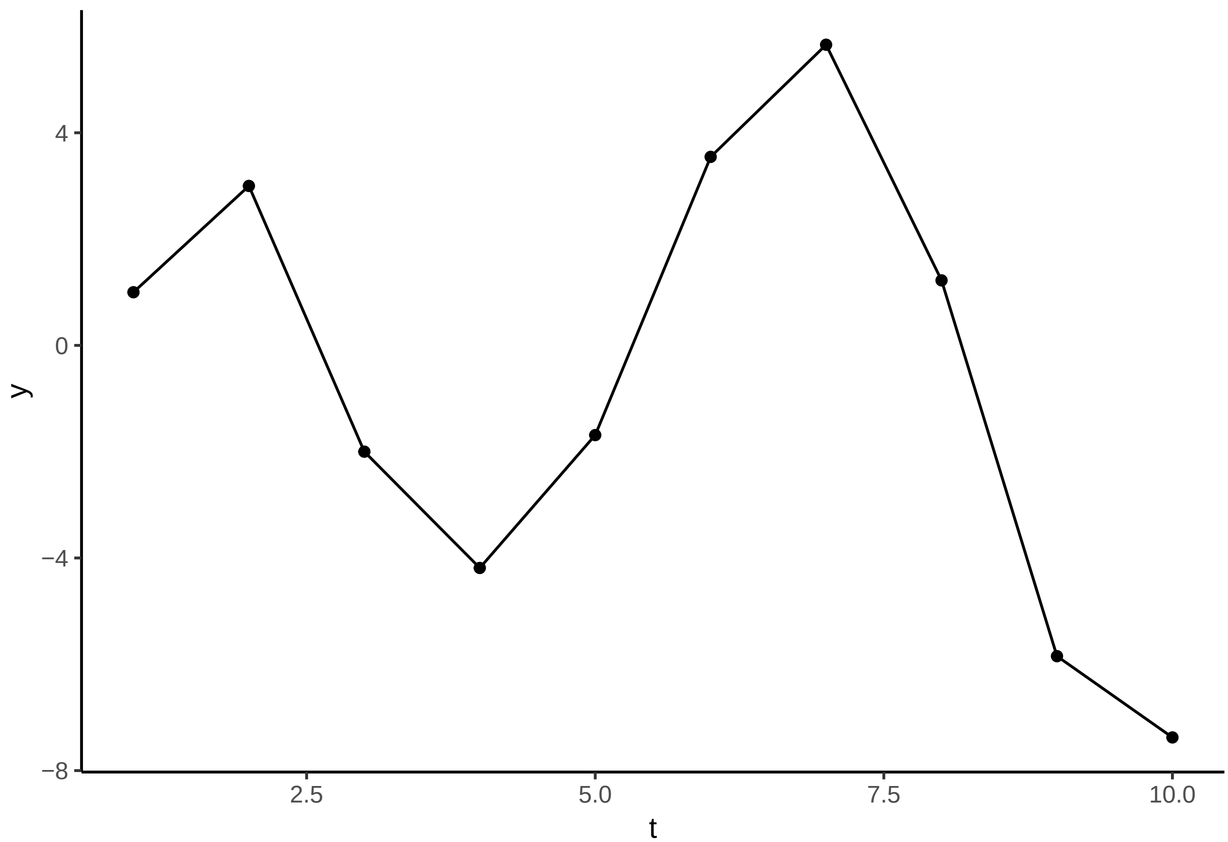
```r
coefs = c(a0, a1, a2)
delta = a1^2 -4*a2*a0
delta
```

```
## [1] -4
```

```r
roots <- polyroot( coefs )
roots
```

```
## [1] 0.5+1i 0.5-1i
```

```r
R = Mod(roots[1])
theta = Arg(roots[1])


# identificando constantes
# y0 = A3
# y1 = R*(A3*cos(theta) + A4*sin(theta))
# y1 = A3*R*cos(theta) + A4*R*sin(theta)
A = matrix(c(1, R*cos(theta), 0, R*sin(theta)), 2 )
B = c(y[1], y[2])
X = ginv(A)%*%B
A3 = X[1]
A4 = X[2]

for (t in 3:n){

  y[t] = R^t *(A3*cos(theta*t) + A4*sin(theta*t))

}

series = data.frame(t = 1:n, y = y)


ggplot(series, aes(x=t, y=y)) +

  geom_point() +

  geom_line() +

  theme_classic()
```


y
4
0
−4
−8
2.5
5.0
7.5
10.0
t