

Lista 1 - Introdução a Análise de Dados

Funções

Gabarito

Guilherme Masuko

March 2023

Questão 1

Crie uma função que recebe três parâmetros a , b , c , os coeficientes de uma função do segundo grau e retorne as raízes dessa função.

Uma função do segundo grau tem a seguinte forma

$$f(x) = ax^2 + bx + c$$

As raízes de uma função do segundo grau são os x 's cujo a função $f(x)$ cruza o eixo das abscissas (horizontal), isso é, $f(x) = 0$.

Para criar a função precisamos relembrar da fórmula de Bhaskara.¹

$$\Delta = b^2 - 4 \cdot a \cdot c$$
$$x_{1,2} = \frac{-b \pm \sqrt{\Delta}}{2 \cdot a}$$

Lembre-se dos casos em que Δ é negativo, positivo e igual a zero.

Teste sua função para as seguintes funções:

- $f(x) = x^2$
- $f(x) = 2x^2 - 18$
- $f(x) = x^2 - 4x + 10$

¹<https://pt.wikipedia.org/wiki/F%C3%B3rmula_quadr%C3%A1tica>

- $f(x) = -2x^2 + 20x - 50$

Solução

```
raizes_funcao_2grau <- function(a, b, c){
  # primeiro definimos o delta
  delta = b^2 - 4*a*c
  # apos isso, vamos quebrar nos três casos
  if(delta < 0){
    # se o delta é negativo, a parabola não tem raiz
    return(cat("Essa função não tem raízes. A parábola não corta o
              eixo das abscissas!"))
  } else if(delta == 0){
    # se o delta é igual a zero, a raiz é unitária
    x = (-b)/(2*a)
    return(cat("Essa função tem apenas uma raiz. Seu valor é",x,"."))
  } else{
    # se o delta é positivo, então temos duas raízes
    x_1 = (-b + sqrt(delta))/(2*a)
    x_2 = (-b - sqrt(delta))/(2*a)
    return(cat("Essa função tem duas raízes. As duas raízes dessa
              função são respectivamente",x_1,"e",x_2,"."))
  }
}

# Testando nas funcoes

# f(x) = x^2
raizes_funcao_2grau(1, 0, 0)

# f(x) = 2x^2 - 18
raizes_funcao_2grau(a=2, b=0, c=-18)

# f(x) = x^2 - 4x + 10
raizes_funcao_2grau(a=1, b=-4, c=10)
```

```
# f(x) = -2x^2 + 20x - 50
raizes_funcao_2grau(a=-2, b=20, c=-50)
```

Questão 2

Com base na questão anterior, crie uma nova função que recebe os mesmos parâmetros a , b e c , coeficientes de uma função do segundo grau, mas que agora retorne um gráfico dessa função.

Solução

```
grafico_funcao_2grau <- function(a, b, c) {
  # primeiro geramos uma sequência de valores para x, de -10 a 10,
  # de meio em meio
  x <- seq(-10, 10, 0.5)
  # depois geramos uma sequência de valores nulos do mesmo tamanho
  # que o vetor x
  y <- rep(0, length(x))
  # precisamos de uma variável de índice
  index <- 0
  # agora faremos um loop, para cada valor de x, temos um valor de y
  # respectivo
  for (valor_x in x) {
    index <- index + 1
    y[index] <- a * valor_x^2 + b * valor_x + c
  }
  plot(x, y)
}

# f(x) = x^2
grafico_funcao_2grau(1, 0, 0)

# f(x) = 2x^2 - 18
grafico_funcao_2grau(a=2, b=0, c=-18)

# f(x) = x^2 - 4x + 10
grafico_funcao_2grau(a=1, b=-4, c=10)

# f(x) = -2x^2 + 20x - 50
```

```
grafico_funcao_2grau(a=-2, b=20, c=-50)
```

Plots resultantes.

Figure 1: $f(x) = x^2$

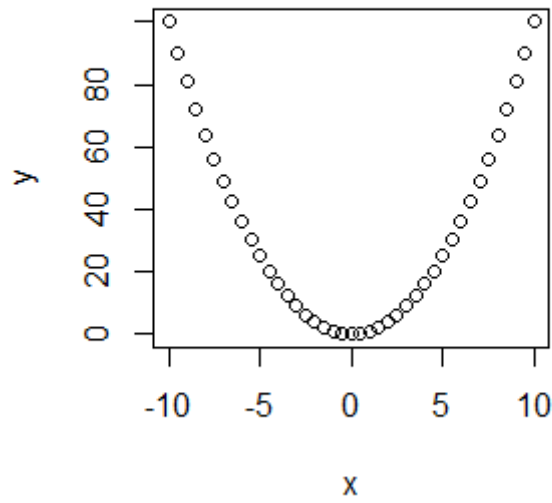


Figure 2: $f(x) = 2x^2 - 18$

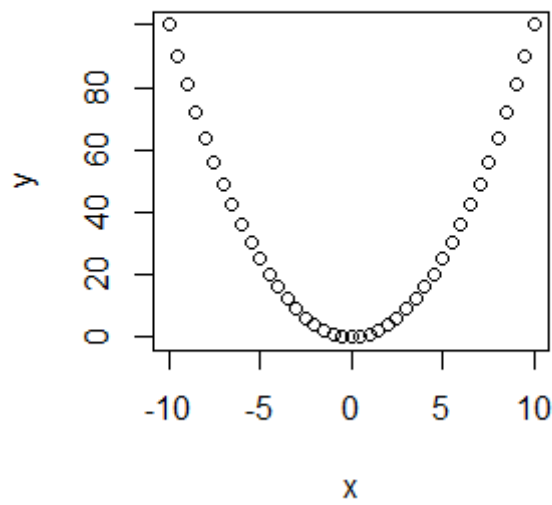


Figure 3: $f(x) = x^2 - 4x + 10$

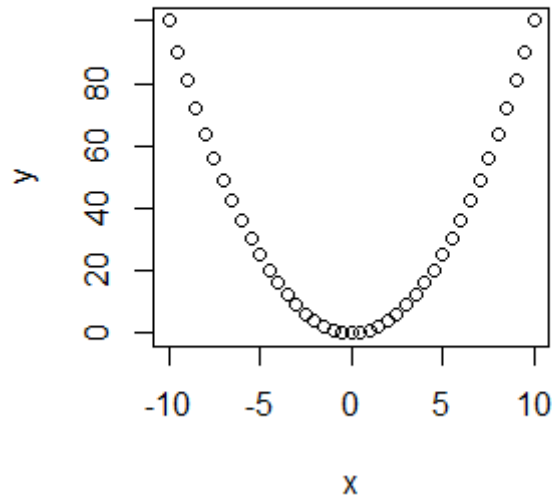
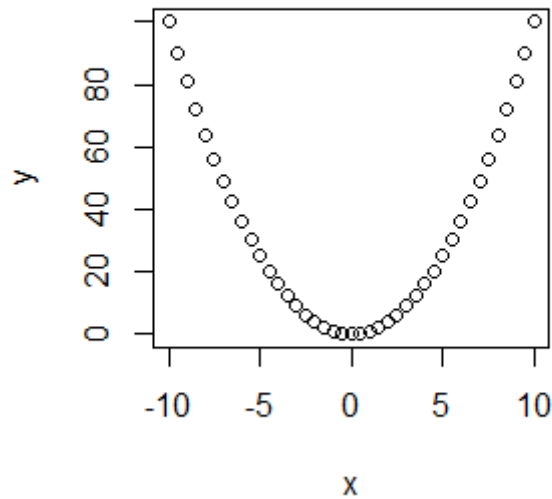


Figure 4: $f(x) = -2x^2 + 20x - 50$



Questão 3

Crie uma função que recebe um número como parâmetro e devolve se o número é primo ou não.

Definição de número primo: Um número primo é um número natural maior que um, tal que não é resultado do produto da multiplicação de dois números naturais, isso é, são números que são divisíveis apenas por ele mesmo dentro da classe dos naturais sem o 1.²

Dica: A função que retorna o resto da divisão inteira é '%%%'.

Teste sua função para os seguintes valores

- valores de 2 à 20
- 577
- 753
- 997

Solução

```
grafico_funcao_2grau <- function(a, b, c) {  
  # primeiro geramos uma sequência de valores para x, de -10 a 10,  
  # de meio em meio  
  x <- seq(-10, 10, 0.5)  
  # depois geramos uma sequência de valores nulos do mesmo tamanho  
  # que o vetor x  
  y <- rep(0, length(x))  
  # precisamos de uma variável de índice  
  index <- 0  
  # agora faremos um loop, para cada valor de x, temos um valor de y  
  # respectivo  
  for (valor_x in x) {  
    index <- index + 1  
    y[index] <- a * valor_x^2 + b * valor_x + c  
  }  
  plot(x, y)  
}
```

```
# f(x) = x^2
```

```
grafico_funcao_2grau(1, 0, 0)
```

```
# f(x) = 2x^2 - 18
```

²<https://pt.wikipedia.org/wiki/Numero_primo>

```
grafico_funcao_2grau(a=2, b=0, c=-18)
```

```
#  $f(x) = x^2 - 4x + 10$ 
```

```
grafico_funcao_2grau(a=1, b=-4, c=10)
```

```
#  $f(x) = -2x^2 + 20x - 50$ 
```

```
grafico_funcao_2grau(a=-2, b=20, c=-50)
```
