# Resolução Lista de exercicio 1

Aluno: Gerson

pdf/Lista R - PGMAT0061.pdf

## Resolução Lista de exercicio 1

### 1. Questão 1

Escreva uma função no R para calcular o coeficiente de correlação linear de Pearson entre duas variáveis. Compare os resultados de sua função com aqueles obtidos pelo R mediante o uso da função cor (.). Para isso, considere o conjunto de dados a seguir:

#### 1.1 Criar a função

```
correlacao <- function(x, y) {
  # numerador
  numerador = sum(
      (x - mean(x)) *
        (y - mean(y))
    )
  # denominador
  denominador <- sqrt(
    sum((x - mean(x))**2) *
      sum((y - mean(y))**2)
  )
  r_person = numerador / denominador

  return(r_person)
}
```

### 1.2 Teste da função

```
x <- c(0.06,-0.55,-1.41,-1.57,0.07,-0.65,0.73,0.73,-0.22,0.27)
y <- c(-0.46, 0.1, -2.51, -2.31, -1.06, -0.67, 0.72, 0.5, 0.4, 0.77)

a <- correlacao(x, y) |> round(7)
b <- cor(x, y) |> round(7)

glue::glue("A correlação calculada é {a} e a correlação do R é {b}")
```

```
A correlação calculada é 0.8497228 e a correlação do R é 0.8497228
```

## 2. Questão 2

Numa determinada localidade, a distribuição de renda (em unidades monetárias, u.m.) é uma variável aleatória X com função de distribuição de probabilidade:

### f.d.p

Definir a função de probabilidade

```
f_X <- function(x) {
  dplyr::case_when(
   x >= 0 & x <= 2 ~ (1/10*x + 1/10)
  ,x  > 2 & x <= 6 ~ (-3/40*x + 9/20)
  ,TRUE ~ 0
  )
}
```

### 2.a Mostre que f(x) é uma f.d.p

### 1. Para qualquer x_i, f(x) $>=$ 0

```
min_x = 0
f_X(min_x) >= 0 # TRU
```

```
[1] TRUE
```

```r
max_x = 6
f_X(max_x) >= 0 # TRUE
```

```
[1] TRUE
```

Como f(x) é maior que zero para o valor minimo e maximo da função, então f(x) >= 0

### 2. F(x_i) = 1 (Acumulada 0 a 6)

Definir a função de distribuição acumulada

```r
F_X <- function(x) {
  if_else(x >= 0 & x <= 6, integrate(f_X, lower = 0, upper = x)$value, 0)
}

glue::glue("Seja x = 6 o limite superior de X,
           então F({max_x}) = {F_X(max_x)} => f(x) é f.d.p")
```

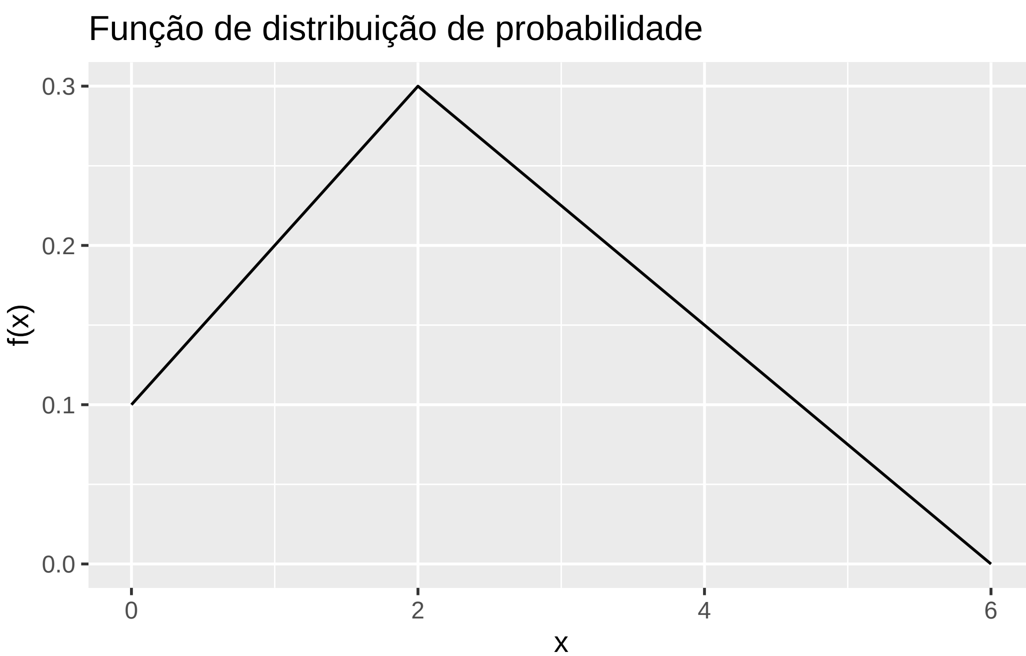
```
Seja x = 6 o limite superior de X,
então F(6) = 1 => f(x) é f.d.p
```

### 2.b Gráfico de f(x).

```r
x <- seq(0, 6, by = 0.1)
y <- purrr::map_dbl(x, f_X)

df <- data.frame(x = x, y = y)

ggplot(df, aes(x = x, y = y)) +
  geom_line() +
  labs(
    title = "Função de distribuição de probabilidade"
    ,x = "x"
    ,y = "f(x)"
    )
```




### 2.c P(x >= 4.5)

```
p_4.5 = (1 - F_X(4.5)) * 100

glue::glue("A probabilidade de encontrar uma pessoa
           com renda superior a 4.5 u.m é {p_4.5 |> round(2)}%")
```

```
A probabilidade de encontrar uma pessoa
com renda superior a 4.5 u.m é 8.44%
```

### 2.d Calcular E(X) e Var(X)

#### E(X)

```
E_X <- integrate(\(x) {x * f_X(x)}, lower = 0, upper = 6)$value
E_X |> round(2)
```

```
[1] 2.47
```

### Var(X)

```
Var_X <- integrate(\(x) {(x - E_X)^2 * f_X(x)}, lower = 0, upper = 6)$value
Var_X |> round(2)
```

```
[1] 1.78
```

## 3. Questão 3

### Seja a amostra abaixo obtida de uma distribuição Poisson de parâmetro lampda:

```
x <- c(5,4, 6,2, 2,4, 5, 3, 3, 0, 1, 7, 6, 5, 3, 6, 5, 3, 7, 2)
```

### 3.1 Obtenha o gráfico da função de log-verossimilhança

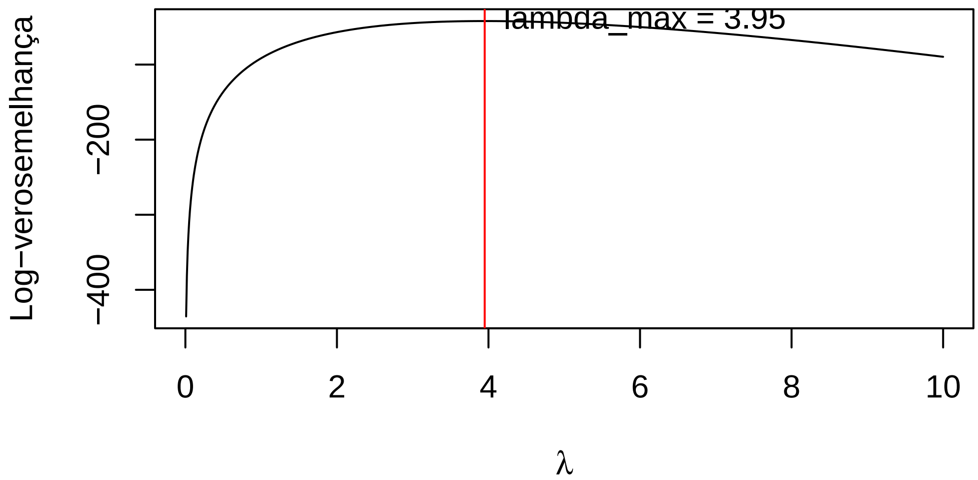
```
# Função de log-verosemelhança
log_verosemelhanca <- function(lambda, x){
  n <- length(x)
    sum(x) *
      log(lambda) -
      n*lambda -
      sum(log(factorial(x)))
}

# Valores de lampda
lambda <- seq(0, 10, 0.01)

# log-verosemelhança
log_vero_poisson  <- purrr::map_dbl(lambda, ~log_verosemelhanca(.x, x))

# Grafico
plot(lambda,
     log_vero_poisson,
     type = "l",
     xlab = expression(lambda),
     ylab = "Log-verosemelhança"
     )
lambda_max <- lambda[which.max(log_vero_poisson)]
abline(v = lambda_max, col = "red")
```

```r
text(lambda_max, max(log_vero_poisson),
     paste("lambda_max =", round(lambda_max, 2)),
     pos = 4)
```




```r
glue::glue("O valor de lambda que maximiza a função de log-verosemelhança é {lambda_max |> ro
```

```
O valor de lambda que maximiza a função de log-verosemelhança é 3.95
```

**Questão 4**

Teste Qui-Quadrado para proporções

- H0: Os fenotipos seguem a relação 9:3:3:1.
- H1: Os fenotipos não seguem a relação 9:3:3:1.

```r
# Dados
obeservado <- c(190, 50, 63, 20)
obeservado <- obeservado/sum(obeservado)

esperado <- c(9, 3, 3, 1)
```

```r
esperado <- esperado/sum(esperado)

# Teste Qui-Quadrado
teste <- chisq.test(obeservado, esperado)
p_value <- teste$p.value
print(teste)
```

```
	Pearson's Chi-squared test

data:  obeservado and esperado
X-squared = 8, df = 6, p-value = 0.2381
```

```r
glue::glue("Como o p-valor({p_value |> round(4)}) é maior que 0.05,
            Não temos evidencia para rejeitamos a hipótese de que
            os fenotiopos seguem a relação 9:3:3:1")
```

```
Como o p-valor(0.2381) é maior que 0.05,
Não temos evidencia para rejeitamos a hipótese de que
os fenotiopos seguem a relação 9:3:3:1
```