

funcoes

April 27, 2018

1 Construção de funções simples

```
In [1]: # Jailson Leocadio
# Biblioteca para uso da funcao multhist
library(plotrix)

graphical.analysis = function(x, y) {

  # Verifica se vetores possuem mesmo tamanho
  if (length(x) != length(y)) {
    stop("Vetores de tamanhos de diferentes!")
  }

  par( mfrow = c(2, 2) )

  # Plota os pontos dos dois vetores num mesmo grafico
  plot(x, pch=20, ylab="", xlab="", col="gray30",
    main="Valores de x e y no eixo das ordenadas")
  par(new=TRUE)
  plot(y, xaxt="n", ylab="", xlab="", yaxt="n", col="darkgoldenrod", pch=20)
  legend("topright", c("x", "y"), fill=c("gray30", "darkgoldenrod"))

  # Boxplot dos dois vetores lado a lado
  boxplot(x, y, xaxt="n", main="Boxplot", col=c("gray30", "darkgoldenrod"))
  axis(1, at=1:2, labels=c("x", "y"))

  # Histograma dos dois vetores num mesmo grafico
  multhist(list(x, y), main="Histograma", ylab="Frequência",
    col=c("gray30", "darkgoldenrod"))
  legend("topright", c("x", "y"), fill=c("gray30", "darkgoldenrod"))

  # Plota o vetor y em funcao de x e apresenta uma linha de tendencia
  plot(x, y, pch=20, main="y ~ x")
  mdl = coef(lm(y ~ x + I(x^2) + I(x^3)))
  curve(mdl[1] +
    mdl[2] * x +
    mdl[3] * (x ^ 2) +
```

```

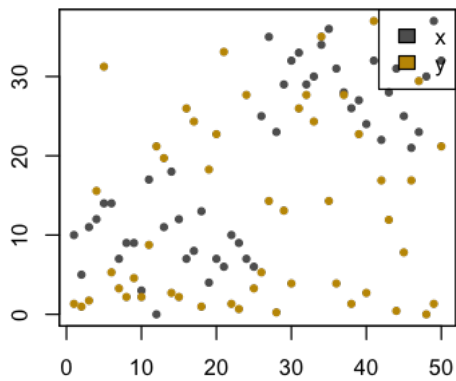
mdl[4] * (x ^ 3),
col = "red", add = T, lty = 2)
}

# Exemplo
x = c(round(rnorm(25, 10, 5), 0), round(rnorm(25, 30, 5), 0))
y = sample(x) ^ 2 / mean(x)

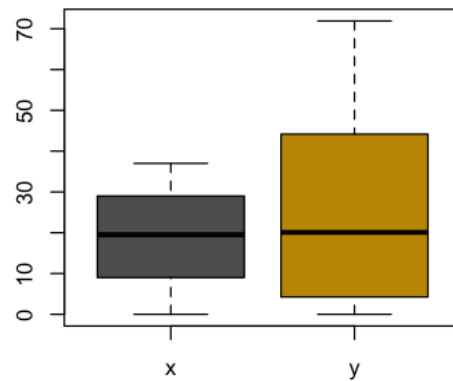
graphical.analysis(x, y)

```

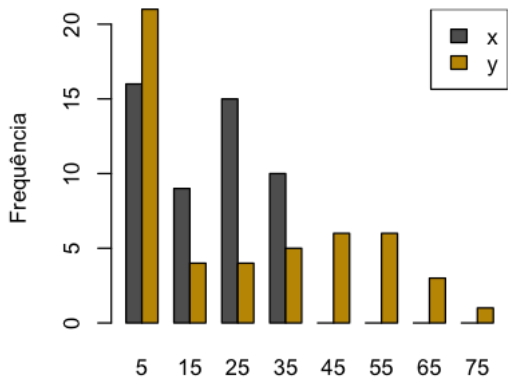
Valores de x e y no eixo das ordenadas



Boxplot



Histograma



y ~ x

