

Graphics: BoxPlot, ScatterPlot, BarPlot, Histograms, Pie

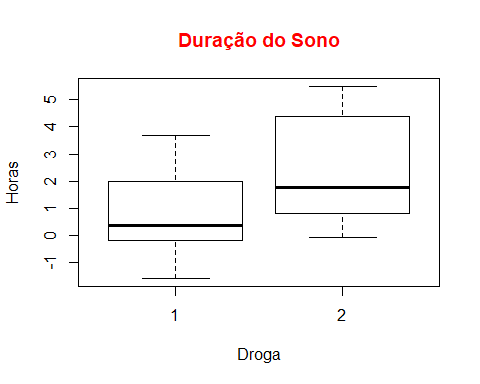
Janeci Leoni Dewes

Basic Graphics options available in the R language.

## 1º Boxplot

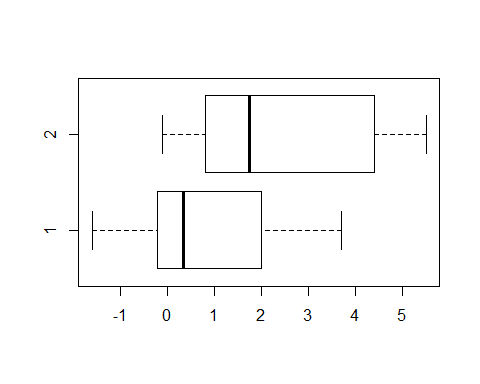
# def Data  
attach(sleep)

sleepboxplot = boxplot(data = sleep, extra ~ group,  
 main = "Duração do Sono",  
 col.main = "red", ylab = "Horas", xlab = "Droga")

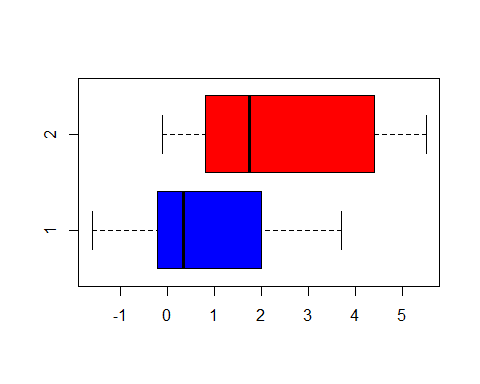


## Horizontal boxplot

# Horizontal boxplot  
horizontalboxplot = boxplot(data = sleep, extra ~ group,  
 ylab = "", xlab = "", horizontal = T)



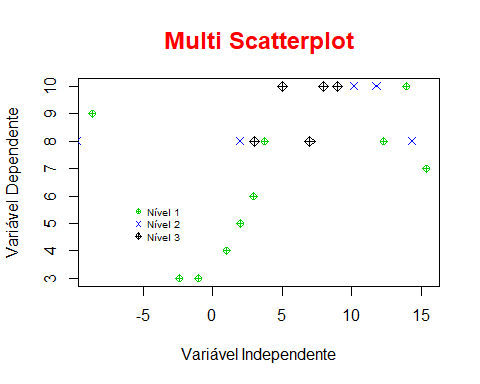
# Add color  
horizontalboxplot = boxplot(data = sleep, extra ~ group,  
 ylab = "", xlab = "", horizontal = T,  
 col = c("blue", "red") )



**2º Scatterplot**

# Def data  
x = rnorm(10,5,7)  
y = rpois(10,7)  
z = rnorm(10,6,7)  
t = rpois(10,9)

# Plot  
plot(x, y, col = 123, pch = 10, main = "Multi Scatterplot",  
 col.main = "red", cex.main = 1.5, xlab = "Variável Independente",   
 ylab = "Variável Dependente")  
  
# Add points  
points(z, t, col = "blue", pch = 4)  
# Add points  
points(y, t, col = 777, pch = 9)  
# Add legend  
legend(-6,5.9, legend = c("Nível 1", "Nível 2", "Nível 3"),  
 col = c(123, "blue", 777), pch = c(10,4,9),  
 cex = 0.65, bty = "n")



## 3º Barplot

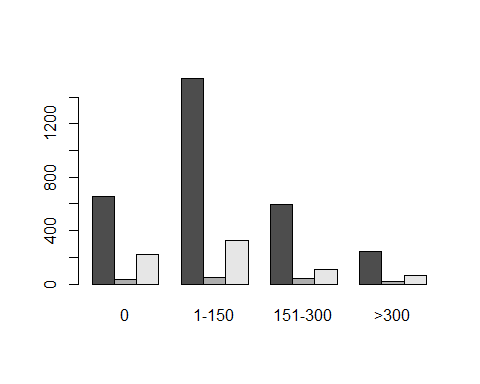
# Def data  
dados <- matrix(c(652,1537,598,242,36,46,38,21,218,327,106,67), nrow = 3, byrow = T)  
dados

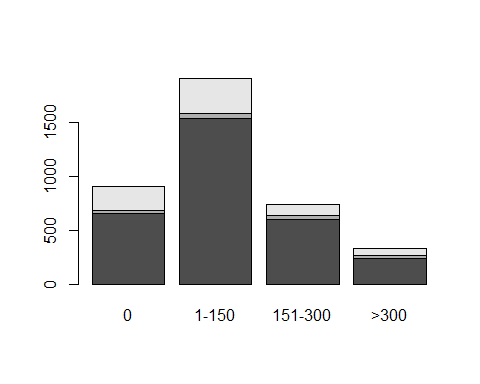
## [,1] [,2] [,3] [,4]  
## [1,] 652 1537 598 242  
## [2,] 36 46 38 21  
## [3,] 218 327 106 67

# Nomeando linhas e colunas na matriz  
colnames(dados) <- c("0","1-150","151-300",">300")   
rownames(dados) <- c("Jovem","Adulto","Idoso")   
dados

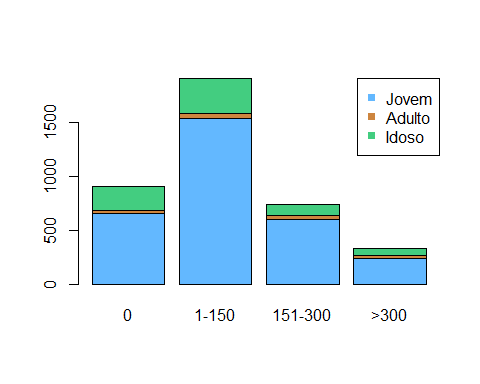
## 0 1-150 151-300 >300  
## Jovem 652 1537 598 242  
## Adulto 36 46 38 21  
## Idoso 218 327 106 67

# Construindo o Barplot  
barplot(dados, beside = T)

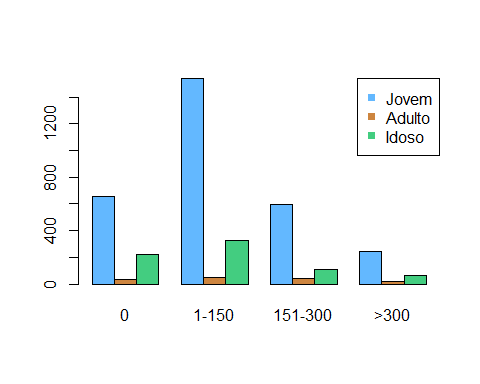
barplot(dados)



# Construindo o plot - Stacked Bar Plot  
# As 3 faixas de idade são representadas na mesma coluna para as diferentes quantidades  
barplot(dados, col = c("steelblue1", "tan3", "seagreen3"))  
legend("topright", pch = 15, col = c("steelblue1", "tan3", "seagreen3"), legend = c("Jovem","Adulto","Idoso"))



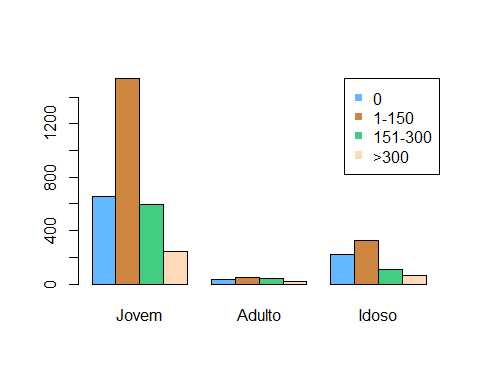
# Color and legend  
barplot(dados, beside = T, col = c("steelblue1", "tan3", "seagreen3"))  
legend("topright", pch = 15, col = c("steelblue1", "tan3", "seagreen3"), legend = c("Jovem","Adulto","Idoso"))



# Com a Transposta da matriz, inverter as posições entre faixa etária e faixa de quantidade  
t(dados)

## Jovem Adulto Idoso  
## 0 652 36 218  
## 1-150 1537 46 327  
## 151-300 598 38 106  
## >300 242 21 67

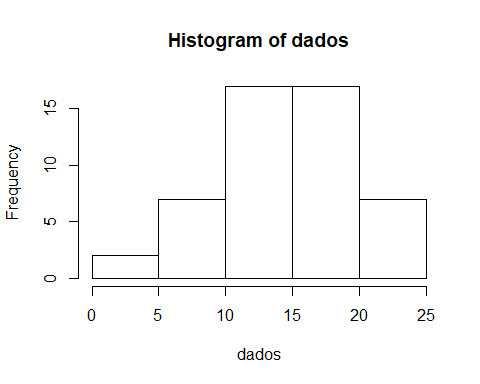
# Grafico de transp dos dados  
barplot(t(dados), beside = T, col = c("steelblue1", "tan3", "seagreen3", "peachpuff1"))  
legend("topright", pch = 15, col = c("steelblue1", "tan3", "seagreen3", "peachpuff1"), legend = c("0","1-150","151-300",">300"))



## 4º Histograms

# Def data  
dados = cars$speed

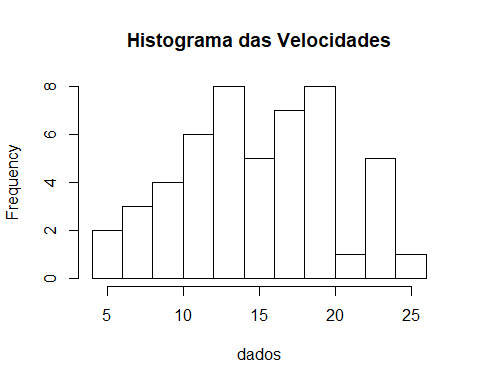
# Construindo um histograma - é usado para visualizar a distribuição de freq de UMA variável.  
# divide em classes (grupos) de dados que aparecem com maior/menor frequencia.  
hist(dados)

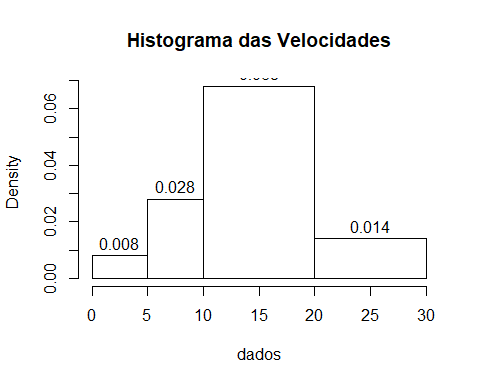


# ?hist

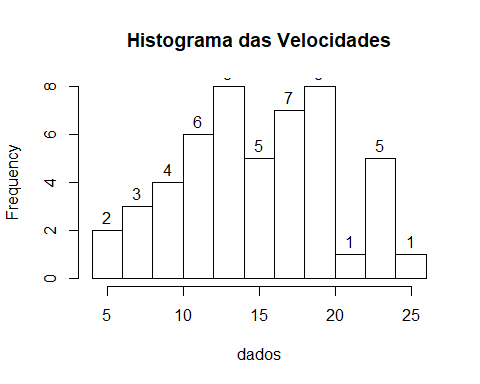
Conforme consta no help, o parâmetro breaks pode ser um dos itens abaixo: Um vetor para os pontos de quebra entre as células do histograma Uma função para calcular o vetor de breakpoints Um único número que representa o número de células para o histograma Uma cadeia de caracteres que nomeia um algoritmo para calcular o número de células Uma função para calcular o número de células.

hist(dados, breaks = 10, main = "Histograma das Velocidades") # breaks: caixinhas

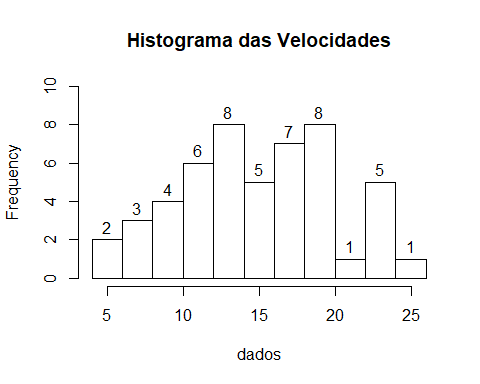


hist(dados, labels = T, breaks = c(0,5,10,20,30), main = "Histograma das Velocidades") # caixinhas específicas

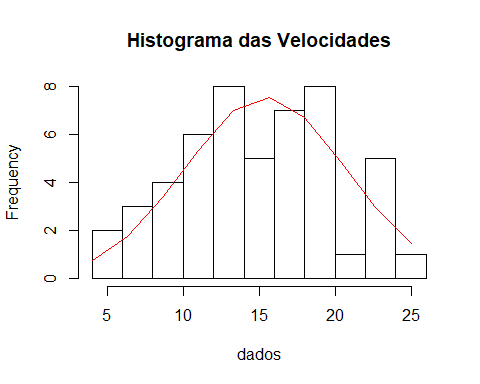
hist(dados, labels = T, breaks = 10, main = "Histograma das Velocidades") # mostra o total de freq. das caixinhas



hist(dados, labels = T, ylim = c(0,10), breaks = 10, main = "Histograma das Velocidades")



# Adicionando linhas ao histograma  
grafico <- hist(dados, breaks = 10, main = "Histograma das Velocidades")  
# Criando uma distribuição normal, media, desvio padrão   
xaxis = seq(min(dados), max(dados), length = 10)  
yaxis = dnorm(xaxis, mean = mean(dados), sd = sd(dados))  
#calcular a diferença:  
yaxis = yaxis\*diff(grafico$mids)\*length(dados)  
  
#acrescentar mais uma camada = distribuição normal   
lines(xaxis, yaxis, col = "red")



distribuição normal mostra se os dados estão com: media ~ = 0 desvio padrão ~ = 1

## 

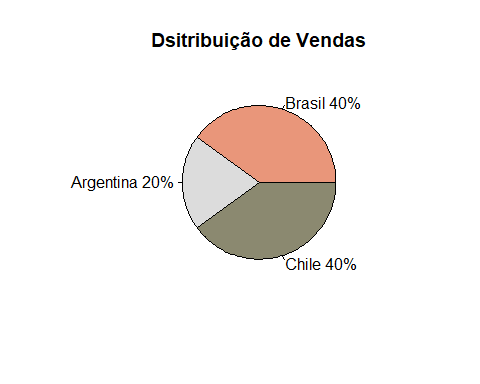
## 5º Pie

# Criando as fatias  
fatias = c(40, 20, 40)  
# Nomeando os labels  
paises = c("Brasil", "Argentina", "Chile")

# Unindo paises e fatias - com a fç paste  
paises = paste(paises, fatias)

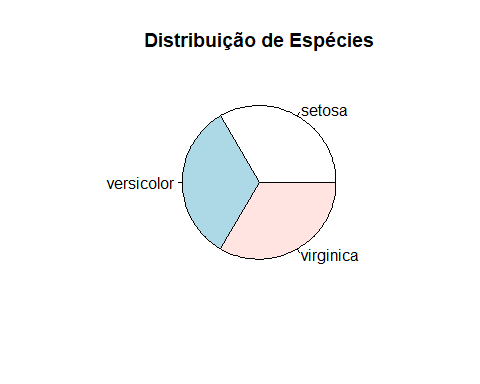
# Incluindo mais detalhes no label  
paises = paste(paises, "%", sep = "")

# Construindo um gráfico  
pie(fatias, labels = paises,  
 col = c("darksalmon", "gainsboro", "lemonchiffon4"),   
 main ="Dsitribuição de Vendas")



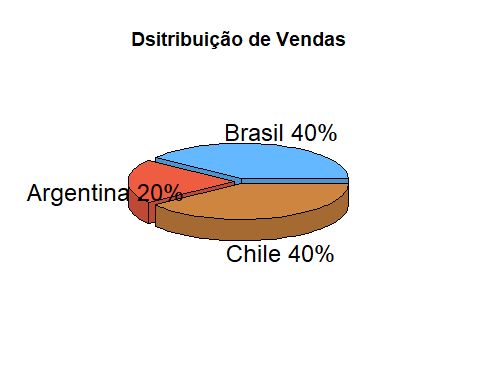
# Def dataframe (iris)  
attach(iris)

Values = table(Species)   
labels = paste(names(Values))  
pie(Values, labels = labels, main = "Distribuição de Espécies")



# 3D - Pcte Plotrix  
library(plotrix)

pie3D(fatias, labels = paises, explode = 0.05,  
 col = c("steelblue1", "tomato2", "tan3"),   
 main = "Dsitribuição de Vendas")

# explode = divisão entre as fatias.