Aula 3.4 - criando gráficos no R

Elder Sodre

Wednesday, February 25, 2015

#### Preparando os dados

Para a aula de hoje, usaremos como exemplo alguns dados com os quais vocês já devem estar acostumados das últimas aulas.

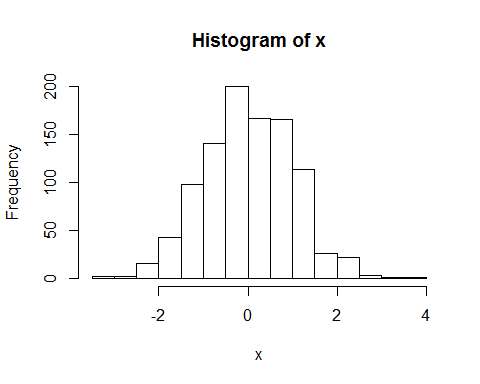
setwd("C:\\Users\\Elder\\Desktop\\Minicurso Elementos de Linguagem R")  
  
#Dados do ecolagoas de 2008:  
ecolagoas <- read.csv2("ecolagoas2008.csv",na.strings="-",row.names=1)

#### histogramas - Função hist()

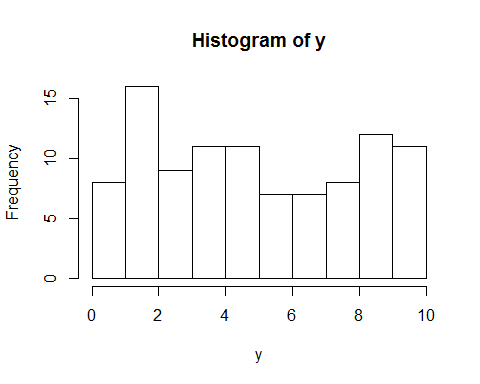
O histograma é um tipo de gráfico que serve para observar a distribuição dos dados.Pode ser considerado um tipo especial de gráfico de barras, em que o valor das barras é a quantidade de observações dentro do intervalo especificado.

Histogramas são criados com a função hist(). Vejamos alguns exemplos:

set.seed(10)  
x <- rnorm(1000)  
  
hist(x)

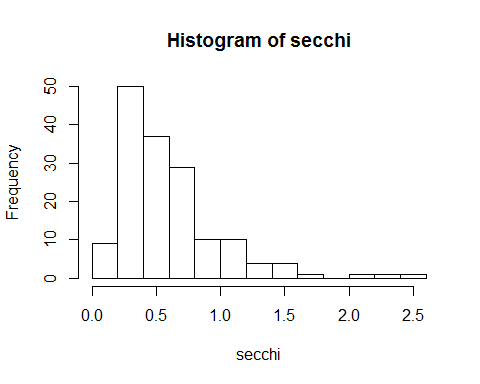


set.seed(2)  
y <- runif(100,0,10)  
  
hist(y)



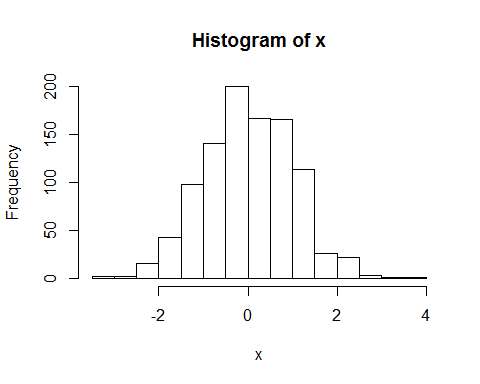
Mais um exemplo:

secchi <- ecolagoas$secchi  
  
hist(secchi)

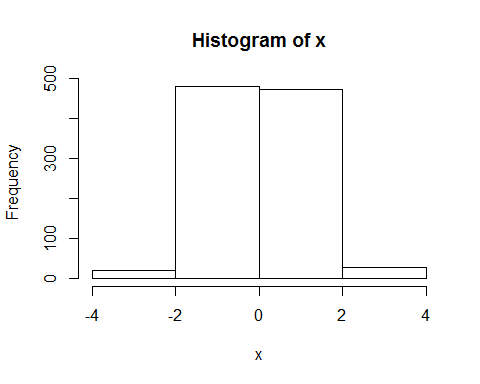


Um argumento da função hist é breaks. Este argumento permite que você determine o número de barras do histograma.

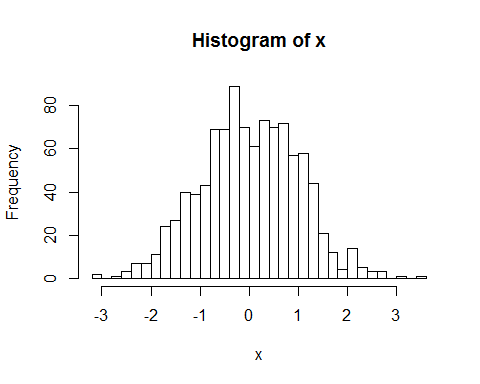
hist(x)



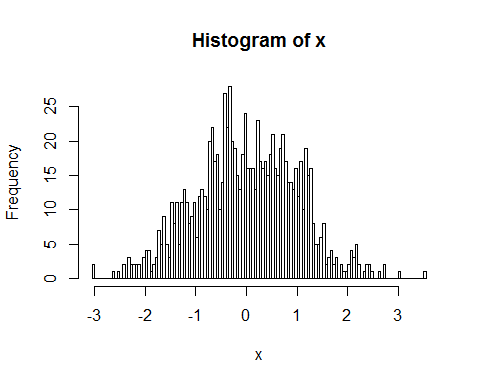
hist(x,breaks=4)



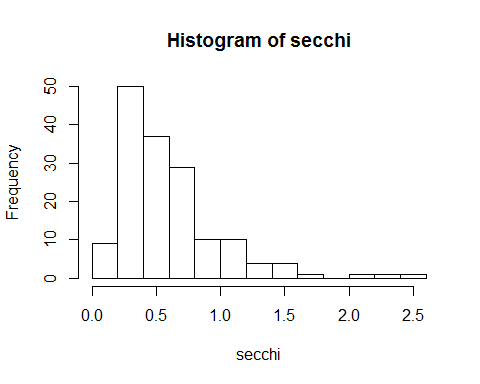
hist(x,breaks=25)



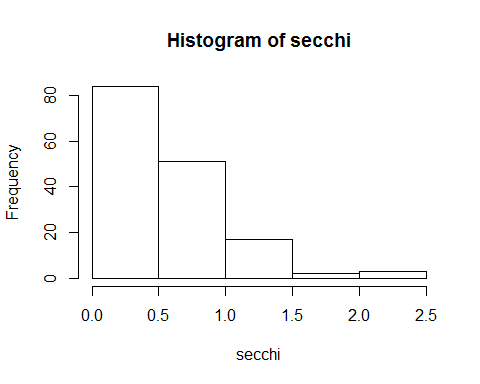
hist(x,breaks=200)



hist(secchi)

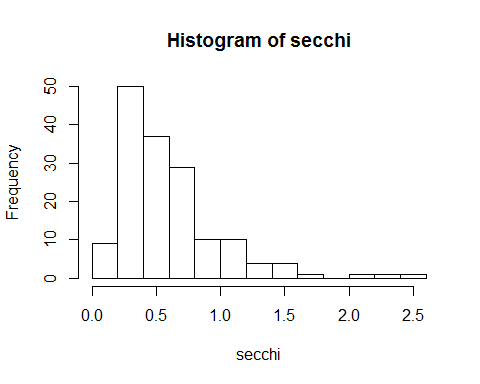


hist(secchi,breaks=5)

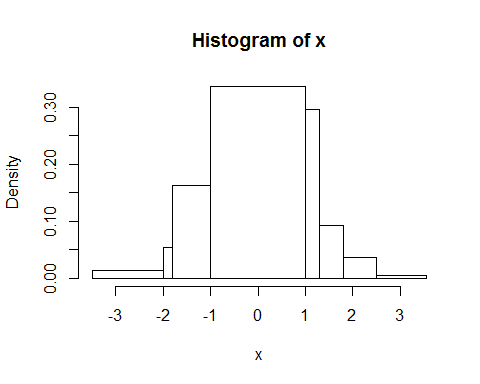


Breaks também pode ser um vetor especificando as posições dos intervalos:

hist(secchi,breaks=seq(0,2.6,by=0.2))



hist(x,breaks=c(-3.5,-2,-1.8,-1,1,1.3,1.8,2.5,3.55))

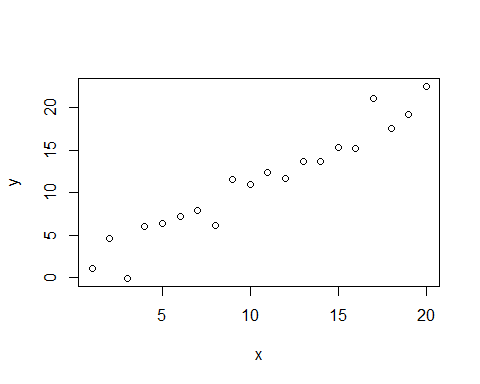


#### Gráficos de dispersão: função plot()

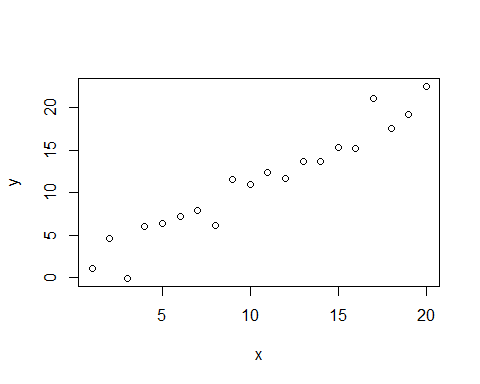
A função plot serve para gráficos de dispersão. Sua fórmula básica é:

plot(x,y,...)

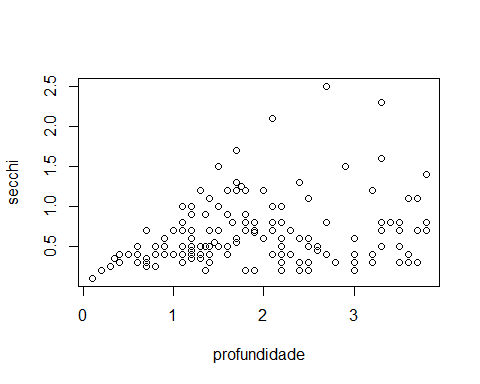
x <- 1:20  
  
set.seed(32)  
y <- 1:20 + rnorm(20,sd=3)  
  
plot(x,y)



#Uma outra alternativa é escrever   
plot(y~x) #,que significa y em função de x.

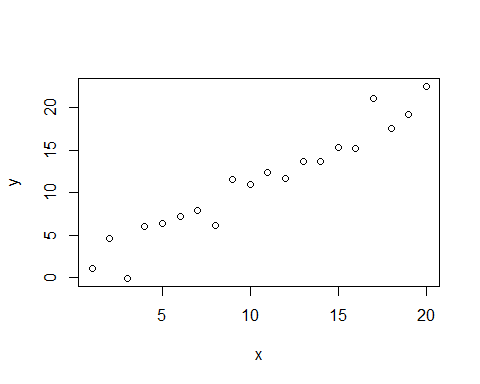


profundidade <- ecolagoas$profundidade  
plot(profundidade,secchi)

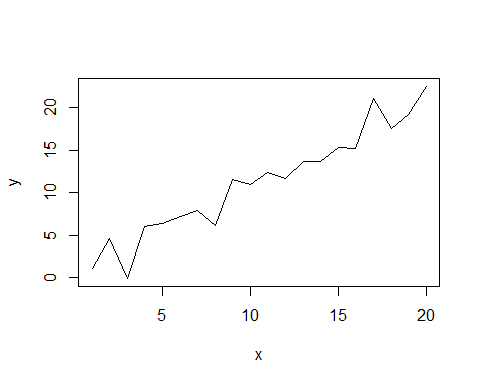


O argumento type permite que você escolha o tipo de gráfico que será feito.

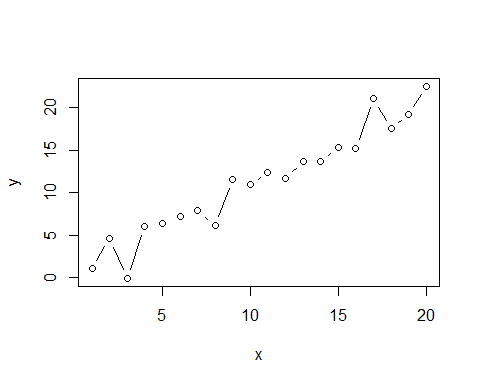
plot(x,y,type="p") #p--> Pontos. Este é o default.



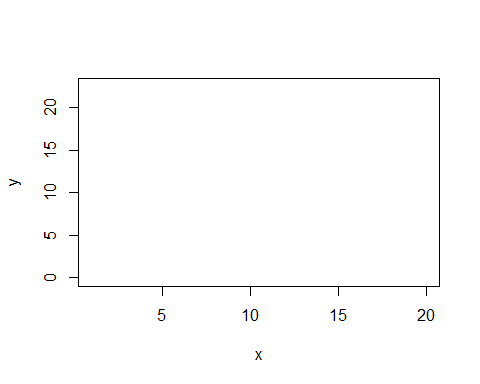
plot(y~x,type="l") #l--> Linhas



plot(x,y,type="b") #b--> Linhas e pontos (both;ambos)



plot(x,y,type="n") #n--> Nada. Apenas os eixos, sem os dados.

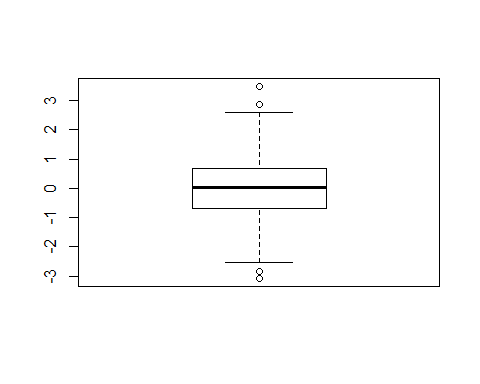


Há ainda outros tipos possíveis. Não se esqueça de conferir em ?plot.

#### Gráficos de boxplot

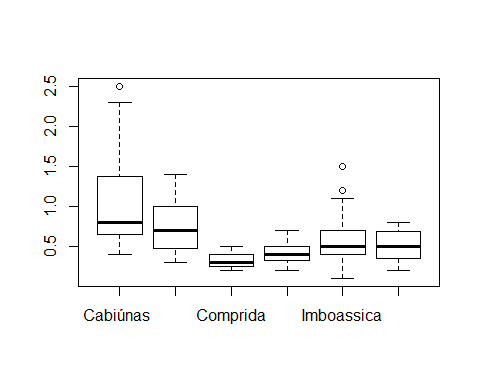
A função boxplot() permite criar gráficos de boxplot.

set.seed(55)  
boxplot(rnorm(1000)) #Boxplot único



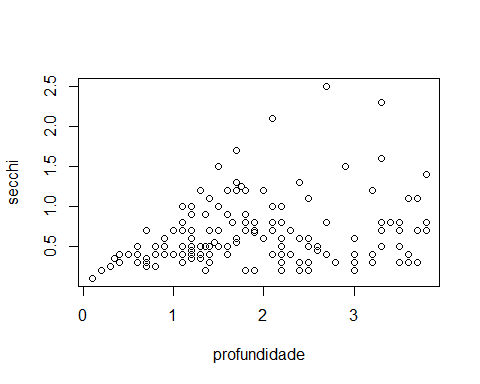
Para fazer mais de um boxplot, a fórmula é boxplot(y~grp), onde y é um vetor numérico de dados, a serem divididos de acordo com o grp (normalmente, um fator)

boxplot(secchi~ecolagoas$lagoa)

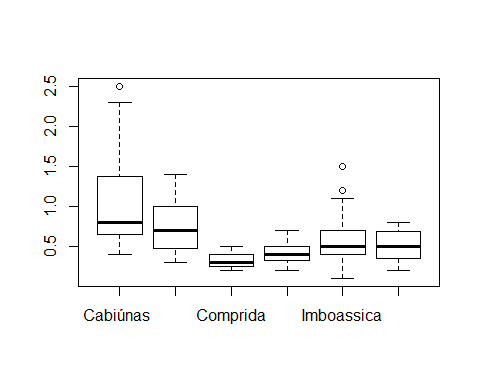


Boxplots são criados automaticamente com a função plot() quando o valor de x é categórico.

plot(profundidade,secchi) #X contínuo; gráfico de dispersão é criado.



plot(ecolagoas$lagoa,secchi) #X categórico; boxplot é criado.



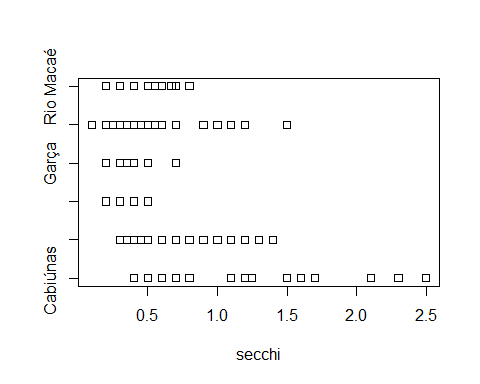
#Observe que criando boxplots com a função plot(), a fórmula plot(grp,y) também funciona.

#### stripchart()

Gráficos de boxplots são úteis quando há muitos dados. Quando há poucos dados, é mais recomendável plotar todos os pontos. Isto é feito com a função stripchart().

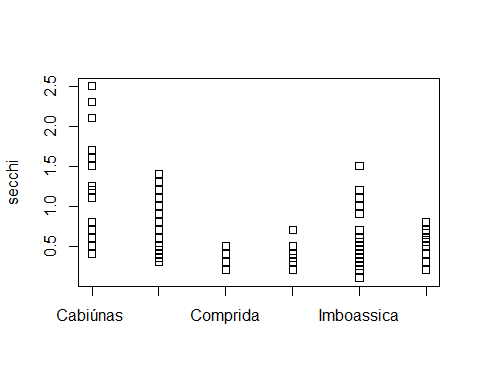
Assim como na função boxplot, sua fórmula é stripchart(y~grp)

stripchart(secchi~ecolagoas$lagoa)



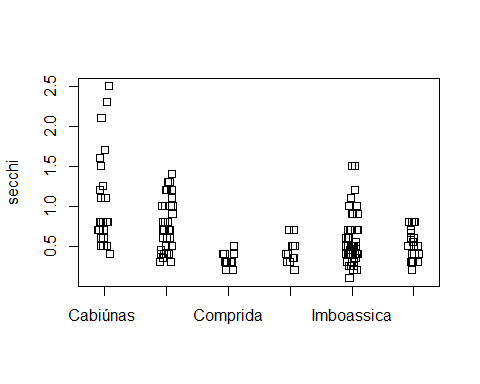
Por definição, o gráfico tem essa aparência deitada. Isto pode ser alterado com o argumento vertical=TRUE

stripchart(secchi~ecolagoas$lagoa,vertical=T)



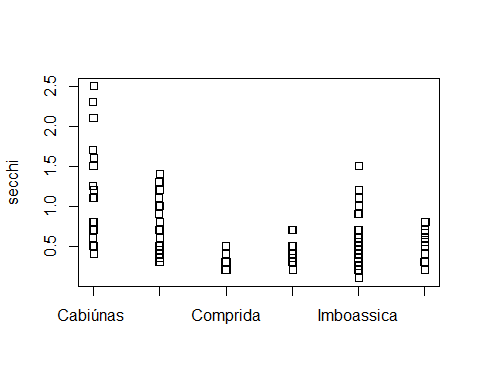
Muitos pontos estão sobrepostos, o que dificulta a sua visualização. Use method="jitter" para contornar este problema.

stripchart(secchi~ecolagoas$lagoa,vertical=T,method="jitter")

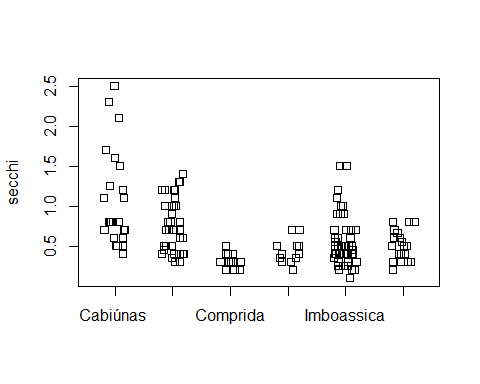


Quando escolhemos method="jitter", aos valores de x é acrescentada uma variação aleatória, que diminui a sobreposição. Use o argumento jitter para controlar a sobreposição.

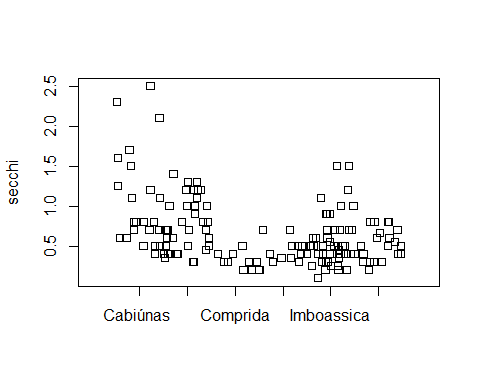
stripchart(secchi~ecolagoas$lagoa,vertical=T,method="jitter",jitter=0.01)



stripchart(secchi~ecolagoas$lagoa,vertical=T,method="jitter",jitter=0.2)



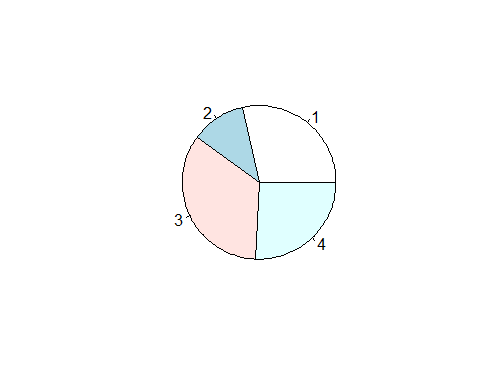
stripchart(secchi~ecolagoas$lagoa,vertical=T,method="jitter",jitter=0.5)



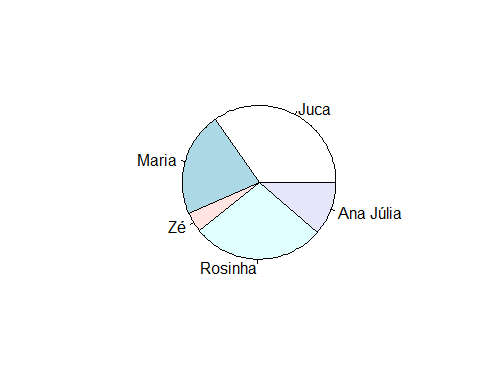
#### Gráficos de pizza - função pie

Gráficos de pizza são criados com a função pie(). O principal argumento desta função é um vetor numérico. Vejamos alguns exemplos:

pie(c(10,4,12,9))



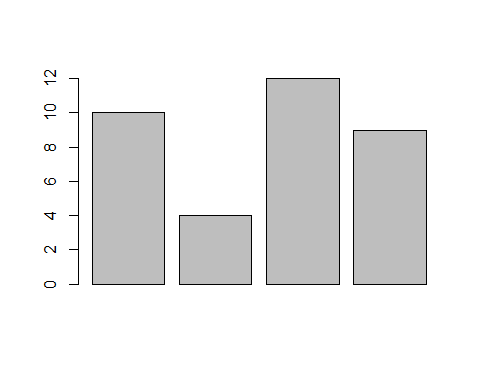
eleições <- c(Juca=789,Maria=502,Zé=95,Rosinha=636,"Ana Júlia"=261)  
  
pie(eleições)



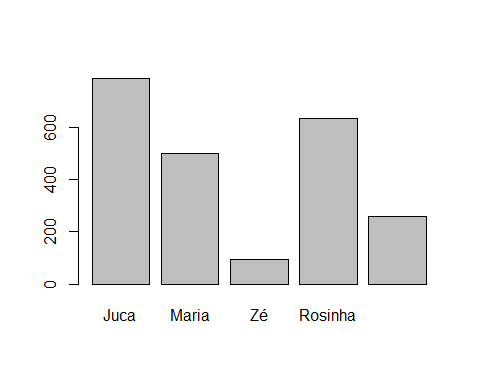
#### Gráficos de barra - função barplot()

Gráficos de barra podem ser feitos de forma similar aos gráficos de pizza.

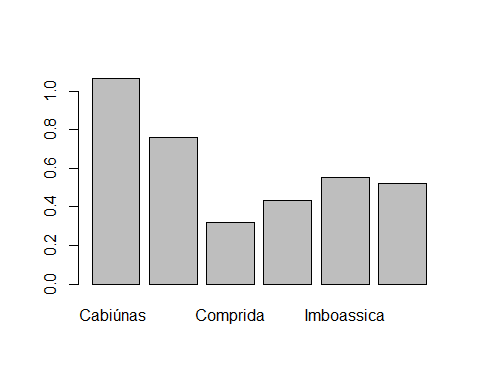
barplot(c(10,4,12,9))



barplot(eleições)

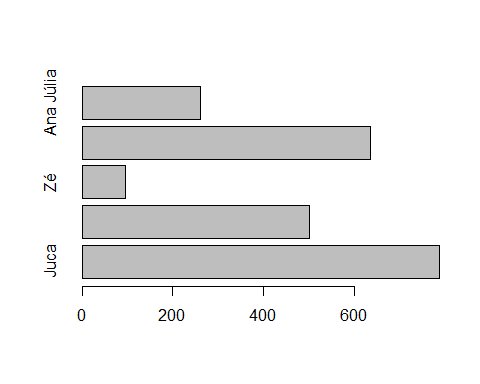


sec.lagoa <- tapply(secchi,ecolagoas$lagoa,mean)  
barplot(sec.lagoa)

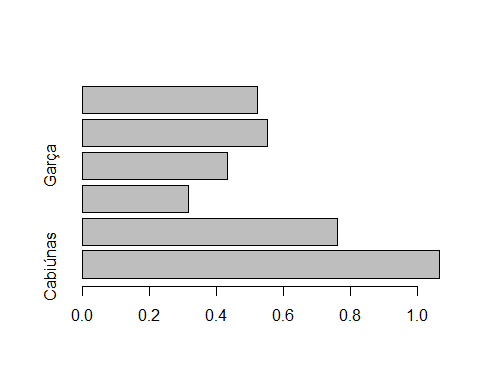


O argumento horiz=TRUE faz com que as barras fiquem na horizontal.

barplot(eleições,horiz=T)

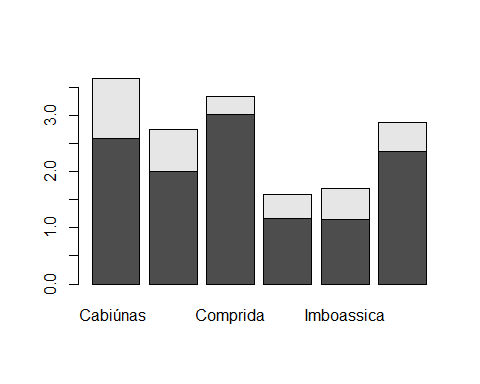


barplot(sec.lagoa,horiz=T)

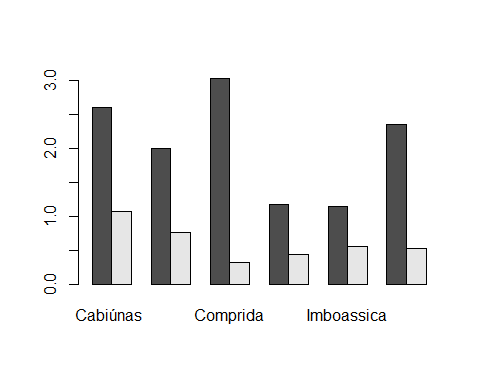


Os dados de um boxplot também podem ser uma matriz.

prof.sec <- cbind(  
 profundidade=tapply(ecolagoas$profundidade,ecolagoas$lagoa,mean),  
 secchi=tapply(ecolagoas$secchi,ecolagoas$lagoa,mean)  
 )  
  
prof.sec <- t(prof.sec)  
  
barplot(prof.sec) #Observe que os dados estão empilhados nas barras. Se você quiser os dados lado a lado, use o argumento beside=TRUE.



barplot(prof.sec,beside=T)



#### Mais detalhes

Para maiores detalhes sobre as possibilidades destes gráficos, uma dica é usar a função example(). Digite as funções abaixo no seu R e veja o que acontece.

example(hist)

example(plot)

example(boxplot)

example(stripchart)

example(pie)

example(barplot)

Não deixe de visitar também o website abaixo:

<http://rgraphgallery.blogspot.com.br/> Blog com diversos exemplos de gráficos no R e os scripts usados para criá-los.