

Gráficos (Introdução)

Ivanildo Batista

10 de março de 2021

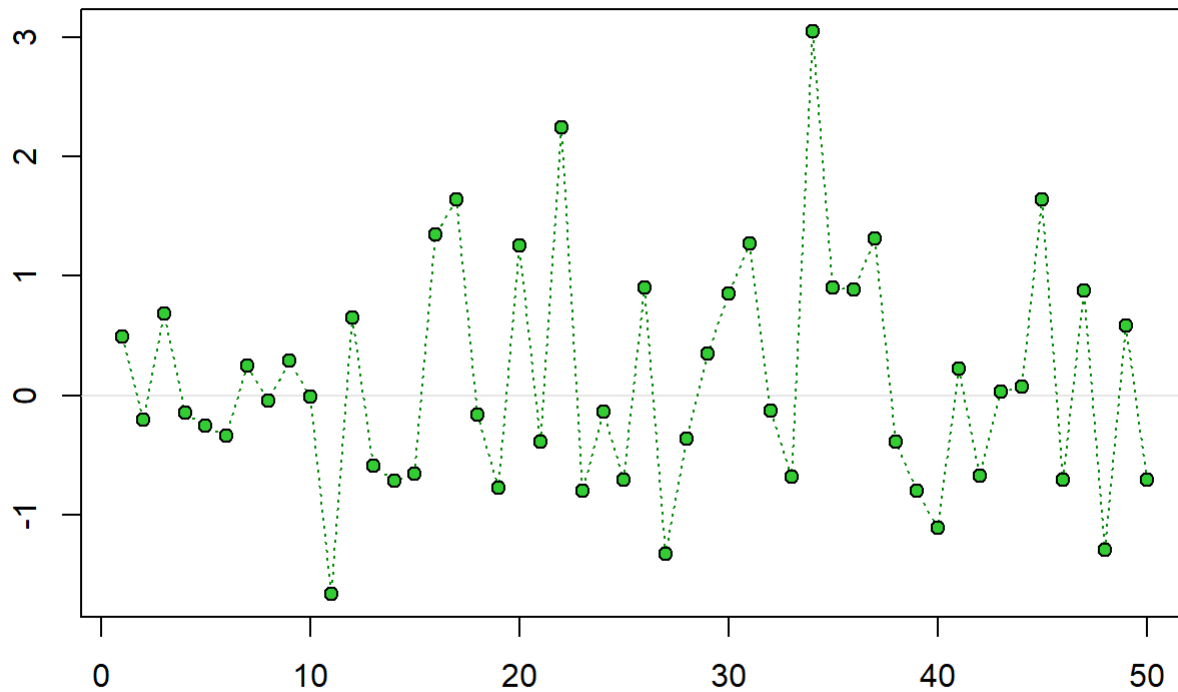
Introdução a gráficos no R

Demonstração de gráficos

```
demo('graphics')
```

```
##  
##  
## demo(graphics)  
## ---- ~~~~~  
##  
## > # Copyright (C) 1997-2009 The R Core Team  
## >  
## > require(datasets)  
##  
## > require(grDevices); require(graphics)  
##  
## > ## Here is some code which illustrates some of the differences between  
## > ## R and S graphics capabilities. Note that colors are generally specified  
## > ## by a character string name (taken from the X11 rgb.txt file) and that line  
## > ## textures are given similarly. The parameter "bg" sets the background  
## > ## parameter for the plot and there is also an "fg" parameter which sets  
## > ## the foreground color.  
## >  
## >  
## > x <- stats::rnorm(50)  
##  
## > opar <- par(bg = "white")  
##  
## > plot(x, ann = FALSE, type = "n")
```

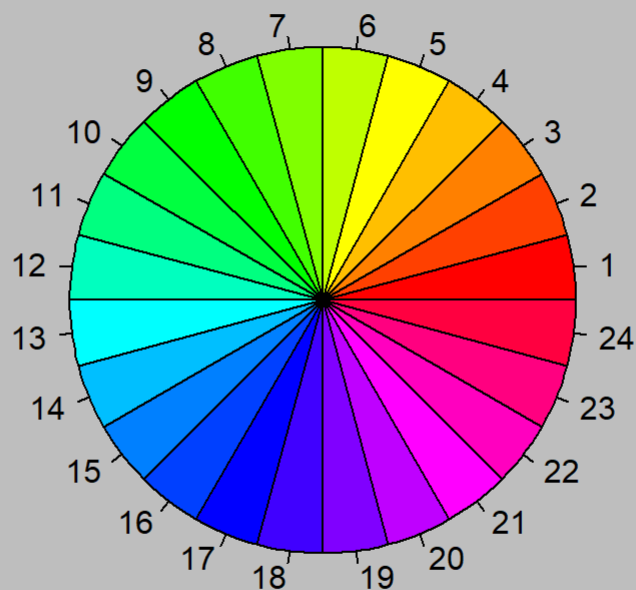
Simple Use of Color In a Plot



Just a Whisper of a Label

```
##
## > abline(h = 0, col = gray(.90))
##
## > lines(x, col = "green4", lty = "dotted")
##
## > points(x, bg = "limegreen", pch = 21)
##
## > title(main = "Simple Use of Color In a Plot",
## +       xlab = "Just a Whisper of a Label",
## +       col.main = "blue", col.lab = gray(.8),
## +       cex.main = 1.2, cex.lab = 1.0, font.main = 4, font.lab = 3)
##
## > ## A little color wheel.    This code just plots equally spaced hues in
## > ## a pie chart.    If you have a cheap SVGA monitor (like me) you will
## > ## probably find that numerically equispaced does not mean visually
## > ## equispaced. On my display at home, these colors tend to cluster at
## > ## the RGB primaries. On the other hand on the SGI Indy at work the
## > ## effect is near perfect.
## >
## > par(bg = "gray")
##
## > pie(rep(1,24), col = rainbow(24), radius = 0.9)
```

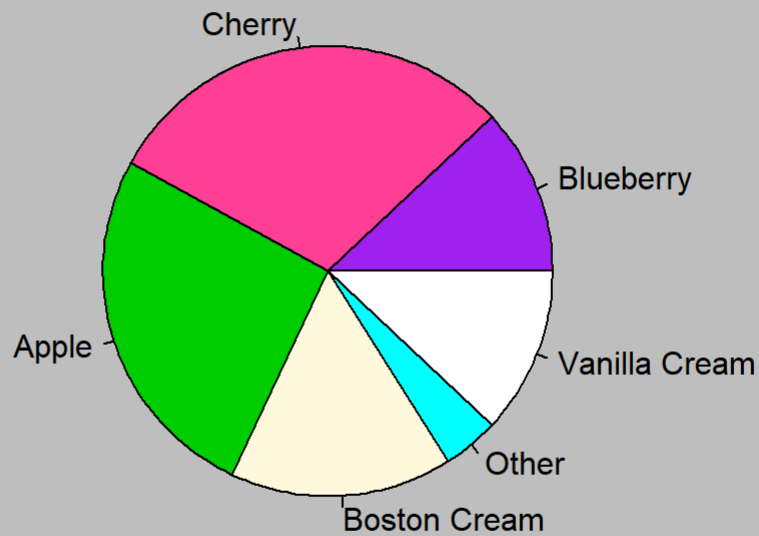
A Sample Color Wheel



(Use this as a test of monitor linearity)

```
##
## > title(main = "A Sample Color Wheel", cex.main = 1.4, font.main = 3)
##
## > title(xlab = "(Use this as a test of monitor linearity)",
## +       cex.lab = 0.8, font.lab = 3)
##
## > ## We have already confessed to having these. This is just showing off X11
## > ## color names (and the example (from the postscript manual) is pretty "cute".
## >
## > pie.sales <- c(0.12, 0.3, 0.26, 0.16, 0.04, 0.12)
##
## > names(pie.sales) <- c("Blueberry", "Cherry",
## +                       "Apple", "Boston Cream", "Other", "Vanilla Cream")
##
## > pie(pie.sales,
## +     col = c("purple", "violetred1", "green3", "cornsilk", "cyan", "white"))
```

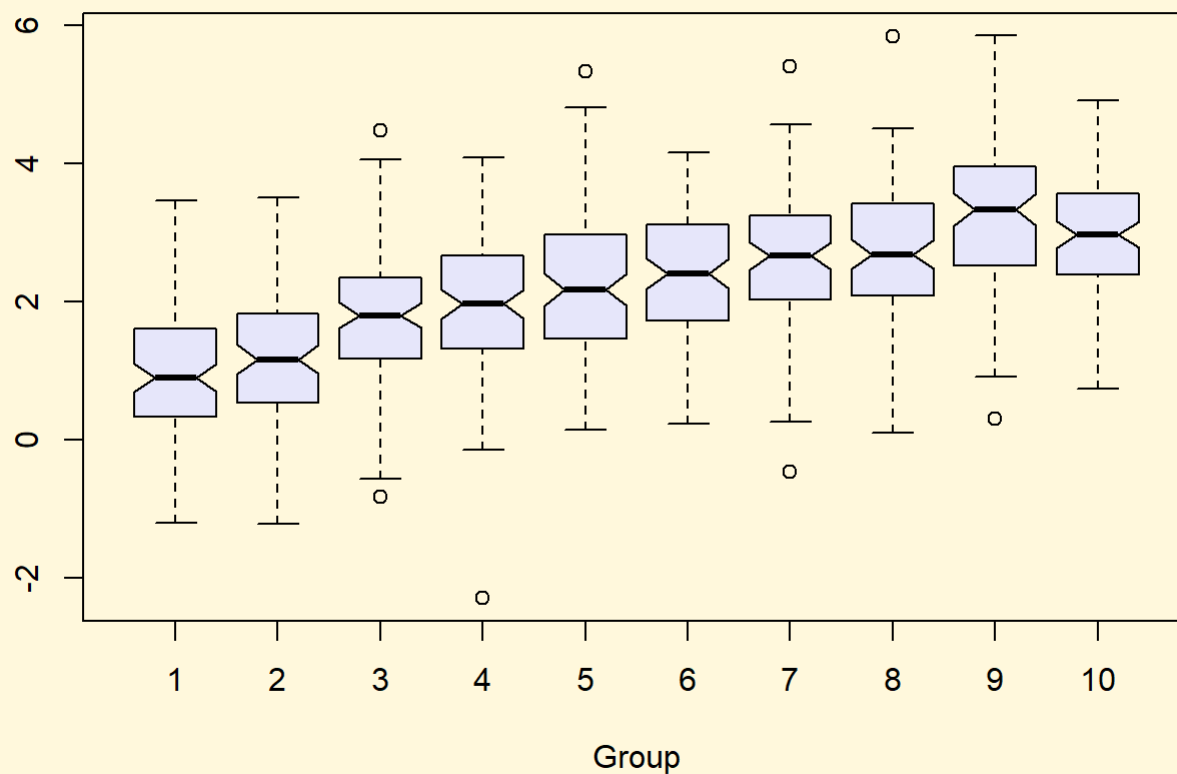
January Pie Sales



(Don't try this at home kids)

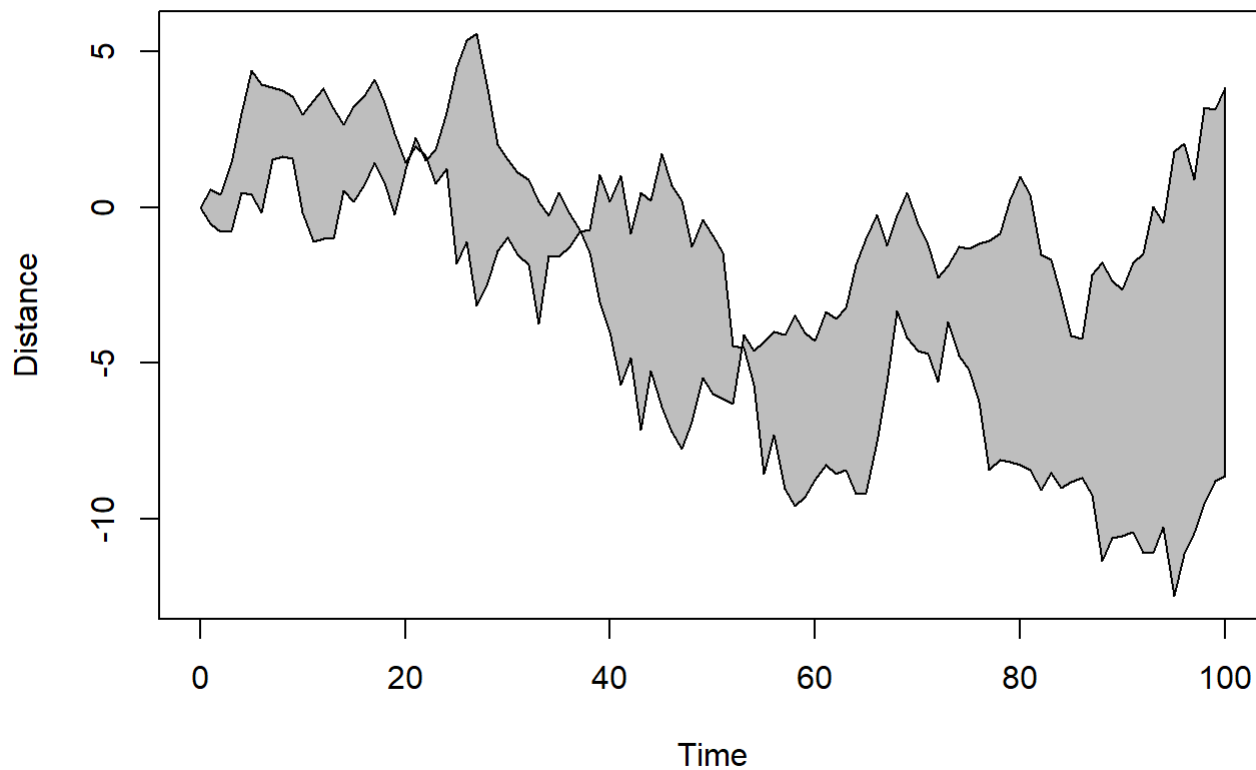
```
##
## > title(main = "January Pie Sales", cex.main = 1.8, font.main = 1)
##
## > title(xlab = "(Don't try this at home kids)", cex.lab = 0.8, font.lab = 3)
##
## > ## Boxplots: I couldn't resist the capability for filling the "box".
## > ## The use of color seems like a useful addition, it focuses attention
## > ## on the central bulk of the data.
## >
## > par(bg="cornsilk")
##
## > n <- 10
##
## > g <- gl(n, 100, n*100)
##
## > x <- rnorm(n*100) + sqrt(as.numeric(g))
##
## > boxplot(split(x,g), col="lavender", notch=TRUE)
```

Notched Boxplots



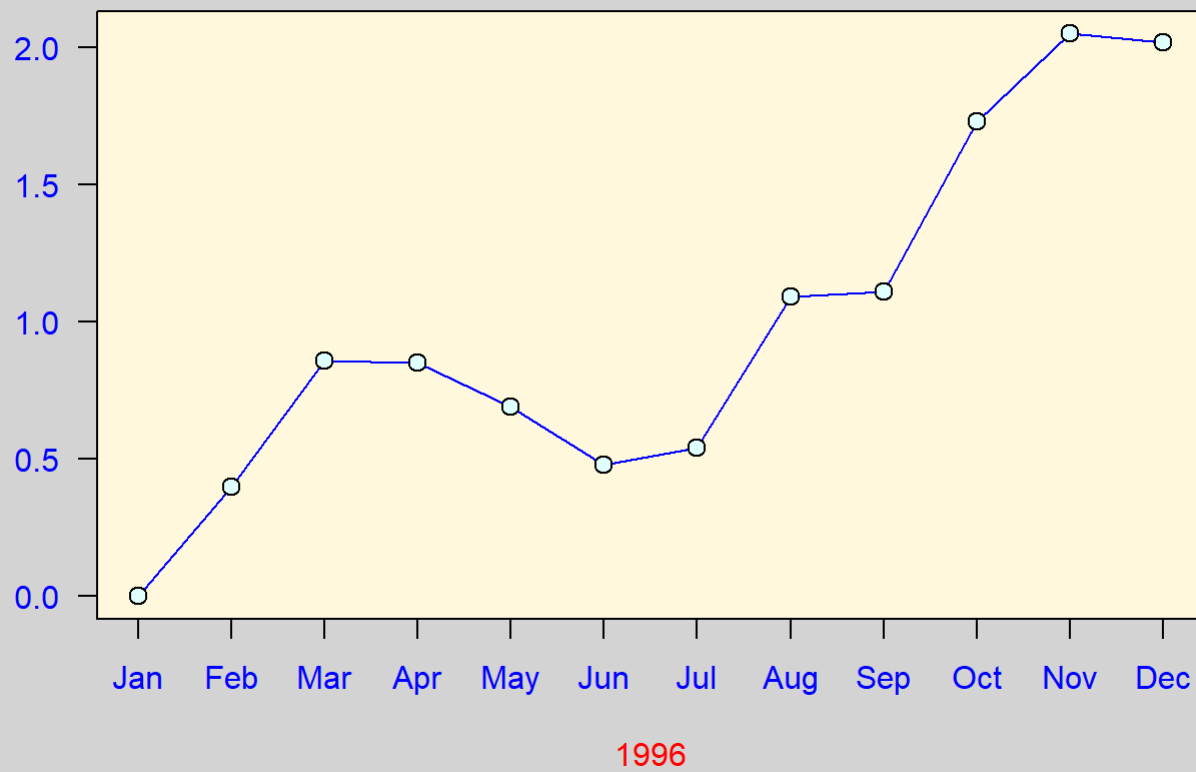
```
##
## > title(main="Notched Boxplots", xlab="Group", font.main=4, font.lab=1)
##
## > ## An example showing how to fill between curves.
## >
## > par(bg="white")
##
## > n <- 100
##
## > x <- c(0,cumsum(rnorm(n)))
##
## > y <- c(0,cumsum(rnorm(n)))
##
## > xx <- c(0:n, n:0)
##
## > yy <- c(x, rev(y))
##
## > plot(xx, yy, type="n", xlab="Time", ylab="Distance")
```

Distance Between Brownian Motions

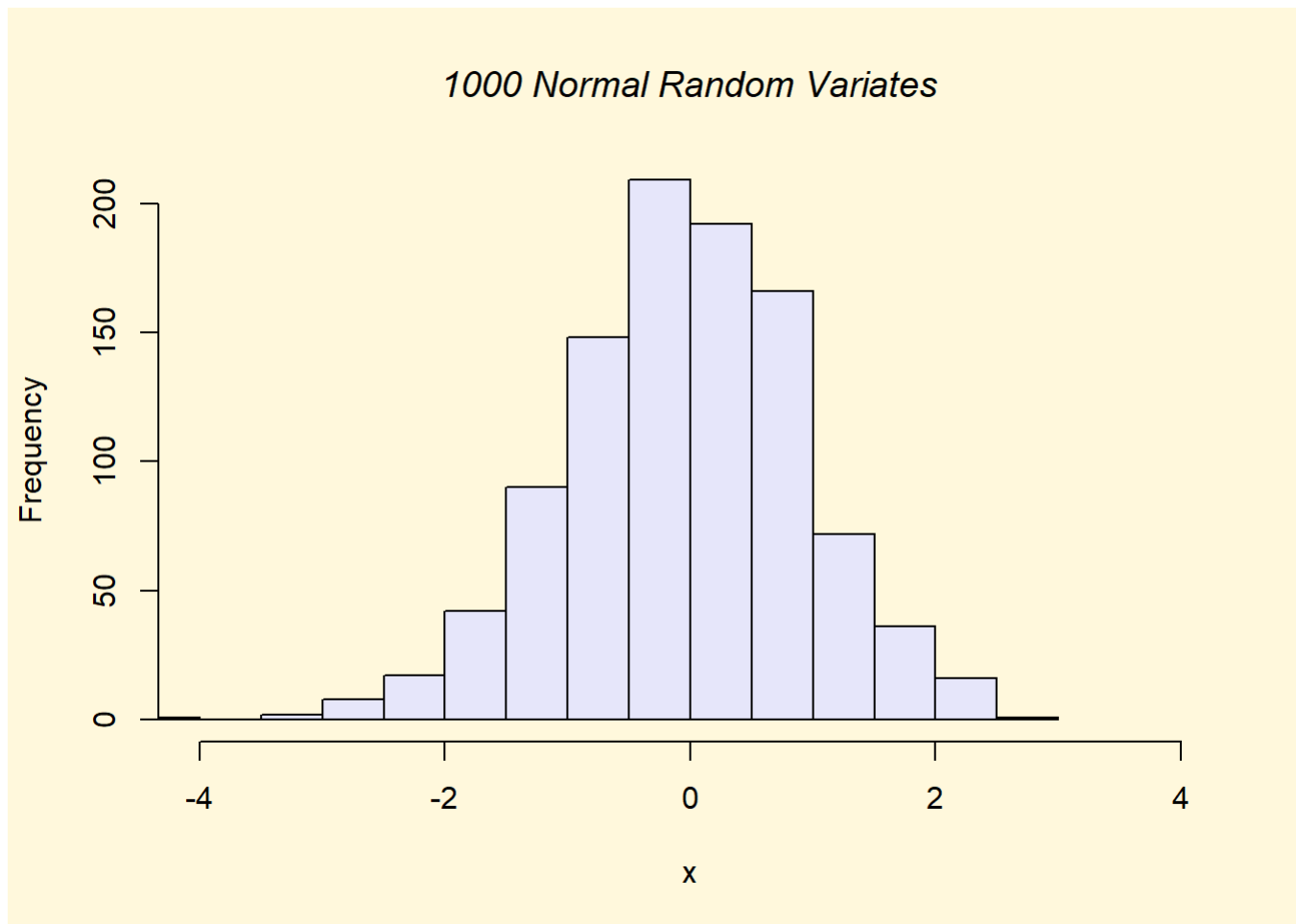


```
##
## > polygon(xx, yy, col="gray")
##
## > title("Distance Between Brownian Motions")
##
## > ## Colored plot margins, axis labels and titles.    You do need to be
## > ## careful with these kinds of effects.    It's easy to go completely
## > ## over the top and you can end up with your lunch all over the keyboard.
## > ## On the other hand, my market research clients love it.
## >
## > x <- c(0.00, 0.40, 0.86, 0.85, 0.69, 0.48, 0.54, 1.09, 1.11, 1.73, 2.05, 2.02)
##
## > par(bg="lightgray")
##
## > plot(x, type="n", axes=FALSE, ann=FALSE)
```

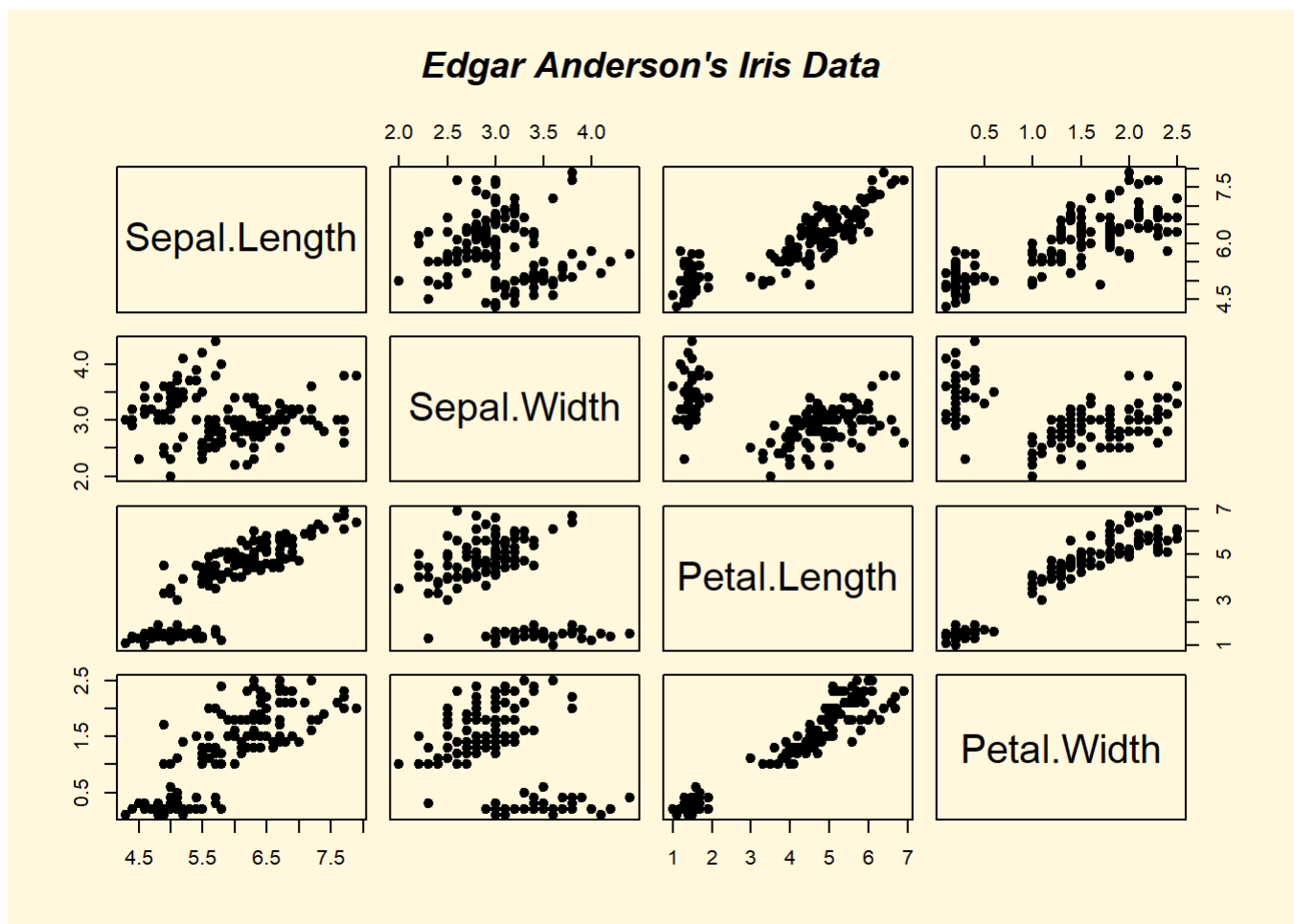
The Level of Interest in R



```
##  
## > usr <- par("usr")  
##  
## > rect(usr[1], usr[3], usr[2], usr[4], col="cornsilk", border="black")  
##  
## > lines(x, col="blue")  
##  
## > points(x, pch=21, bg="lightcyan", cex=1.25)  
##  
## > axis(2, col.axis="blue", las=1)  
##  
## > axis(1, at=1:12, lab=month.abb, col.axis="blue")  
##  
## > box()  
##  
## > title(main= "The Level of Interest in R", font.main=4, col.main="red")  
##  
## > title(xlab= "1996", col.lab="red")  
##  
## > ## A filled histogram, showing how to change the font used for the  
## > ## main title without changing the other annotation.  
## >  
## > par(bg="cornsilk")  
##  
## > x <- rnorm(1000)  
##  
## > hist(x, xlim=range(-4, 4, x), col="lavender", main="")
```

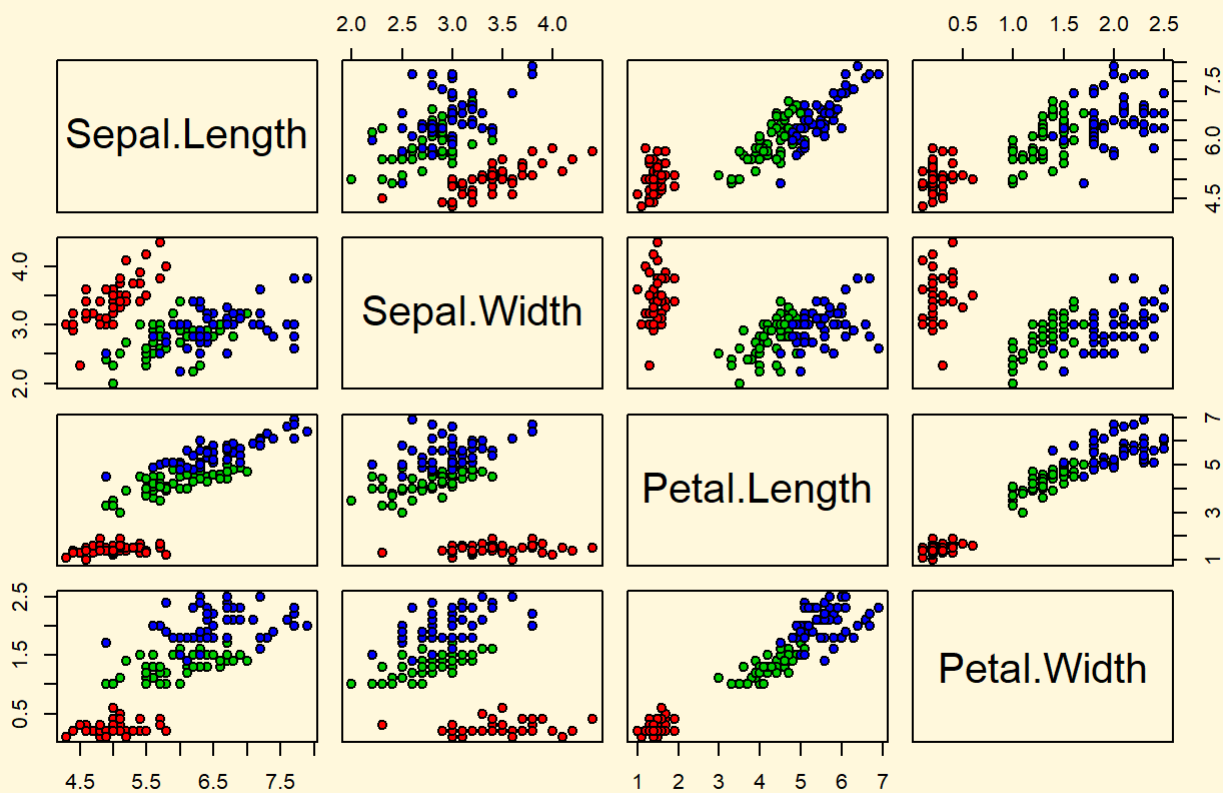



```
##  
## > title(main="1000 Normal Random Variates", font.main=3)  
##  
## > ## A scatterplot matrix  
## > ## The good old Iris data (yet again)  
## >  
## > pairs(iris[1:4], main="Edgar Anderson's Iris Data", font.main=4, pch=19)
```



```
##
## > pairs(iris[1:4], main="Edgar Anderson's Iris Data", pch=21,
## +      bg = c("red", "green3", "blue")[unclass(iris$Species)])
```

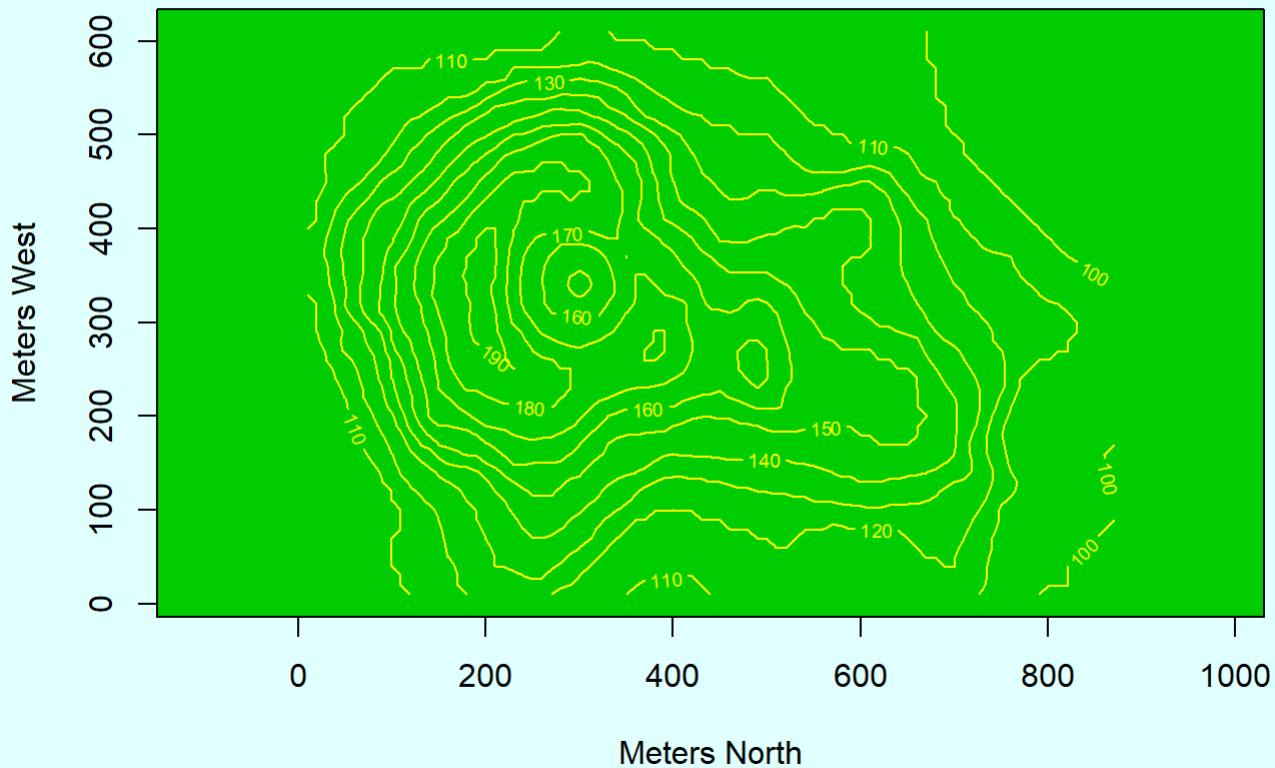
Edgar Anderson's Iris Data



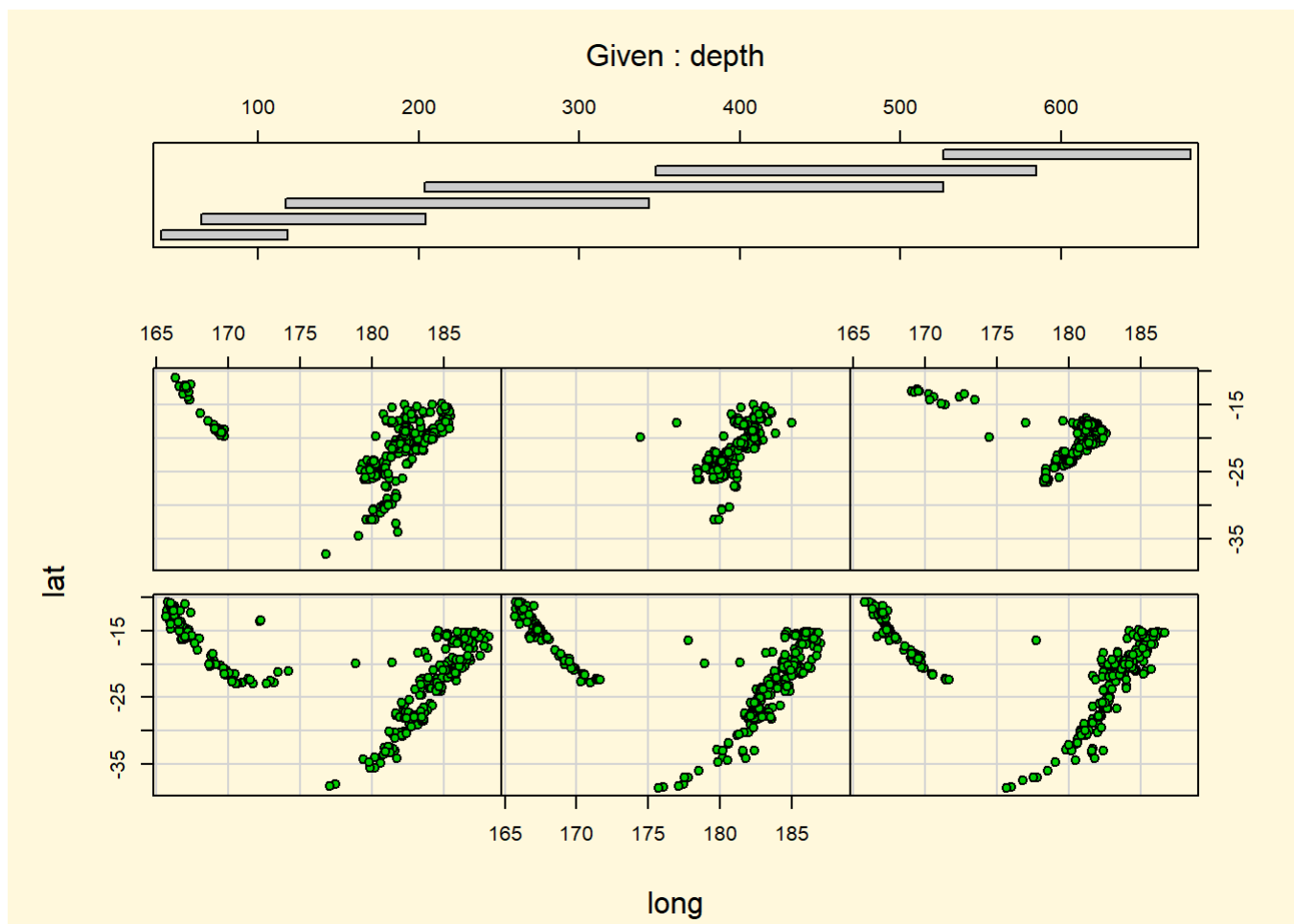
```
##
## > ## Contour plotting
## > ## This produces a topographic map of one of Auckland's many volcanic "peaks".
## >
## > x <- 10*1:nrow(volcano)
##
## > y <- 10*1:ncol(volcano)
##
## > lev <- pretty(range(volcano), 10)
##
## > par(bg = "lightcyan")
##
## > pin <- par("pin")
##
## > xdelta <- diff(range(x))
##
## > ydelta <- diff(range(y))
##
## > xscale <- pin[1]/xdelta
##
## > yscale <- pin[2]/ydelta
##
## > scale <- min(xscale, yscale)
##
## > xadd <- 0.5*(pin[1]/scale - xdelta)
##
## > yadd <- 0.5*(pin[2]/scale - ydelta)
##
## > plot(numeric(0), numeric(0),
## +      xlim = range(x)+c(-1,1)*xadd, ylim = range(y)+c(-1,1)*yadd,
## +      type = "n", ann = FALSE)
```

A Topographic Map of Maunga Whau

10 Meter Contour Spacing



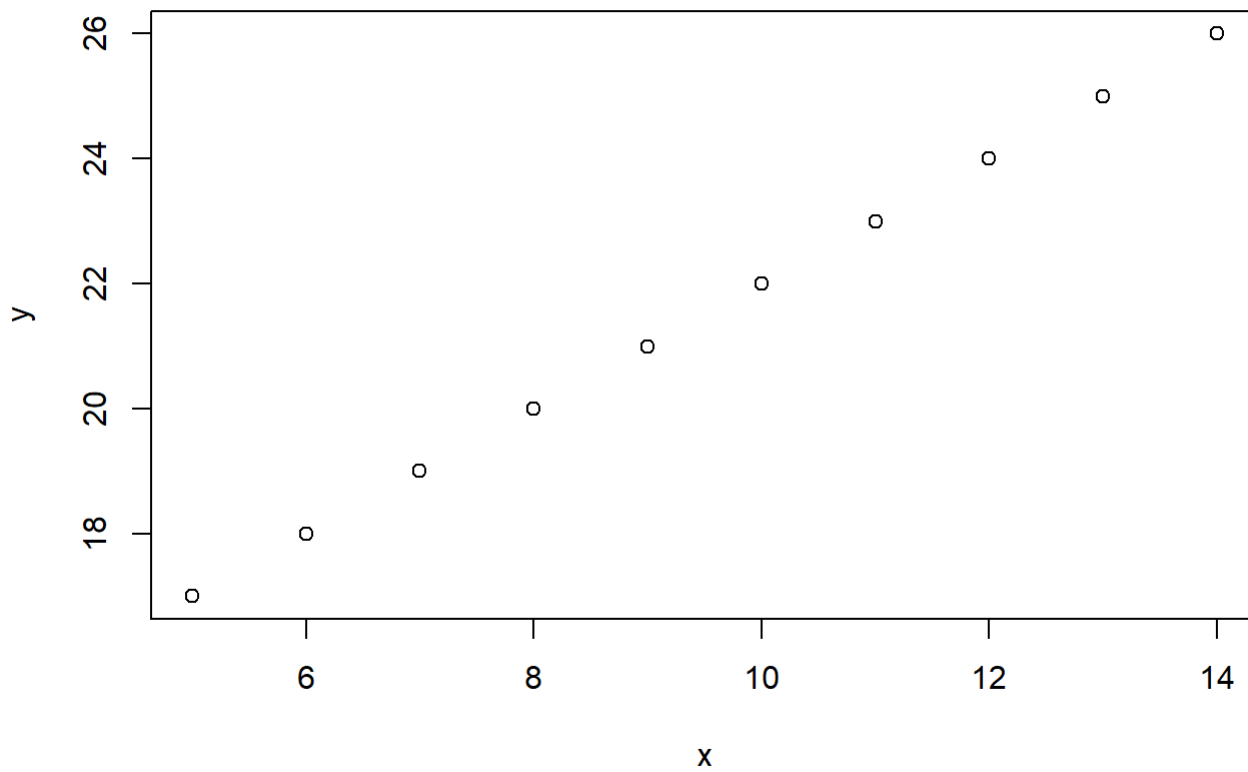
```
##
## > usr <- par("usr")
##
## > rect(usr[1], usr[3], usr[2], usr[4], col="green3")
##
## > contour(x, y, volcano, levels = lev, col="yellow", lty="solid", add=TRUE)
##
## > box()
##
## > title("A Topographic Map of Maunga Whau", font= 4)
##
## > title(xlab = "Meters North", ylab = "Meters West", font= 3)
##
## > mtext("10 Meter Contour Spacing", side=3, line=0.35, outer=FALSE,
## +      at = mean(par("usr")[1:2]), cex=0.7, font=3)
##
## > ## Conditioning plots
## >
## > par(bg="cornsilk")
##
## > coplot(lat ~ long | depth, data = quakes, pch = 21, bg = "green3")
```



```
##  
## > par(opar)
```

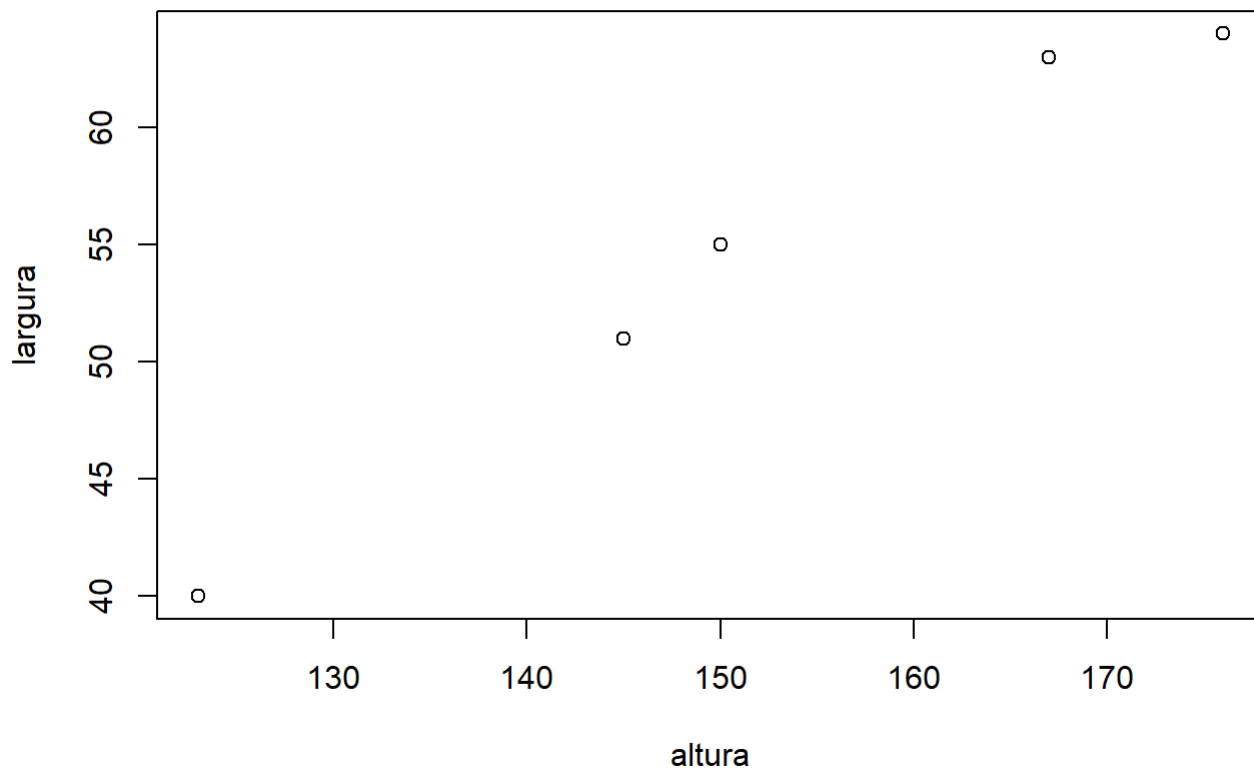
Criando variaveis e gerando um grafico simples

```
x = 5:14  
y = 17:26  
plot(x,y)
```



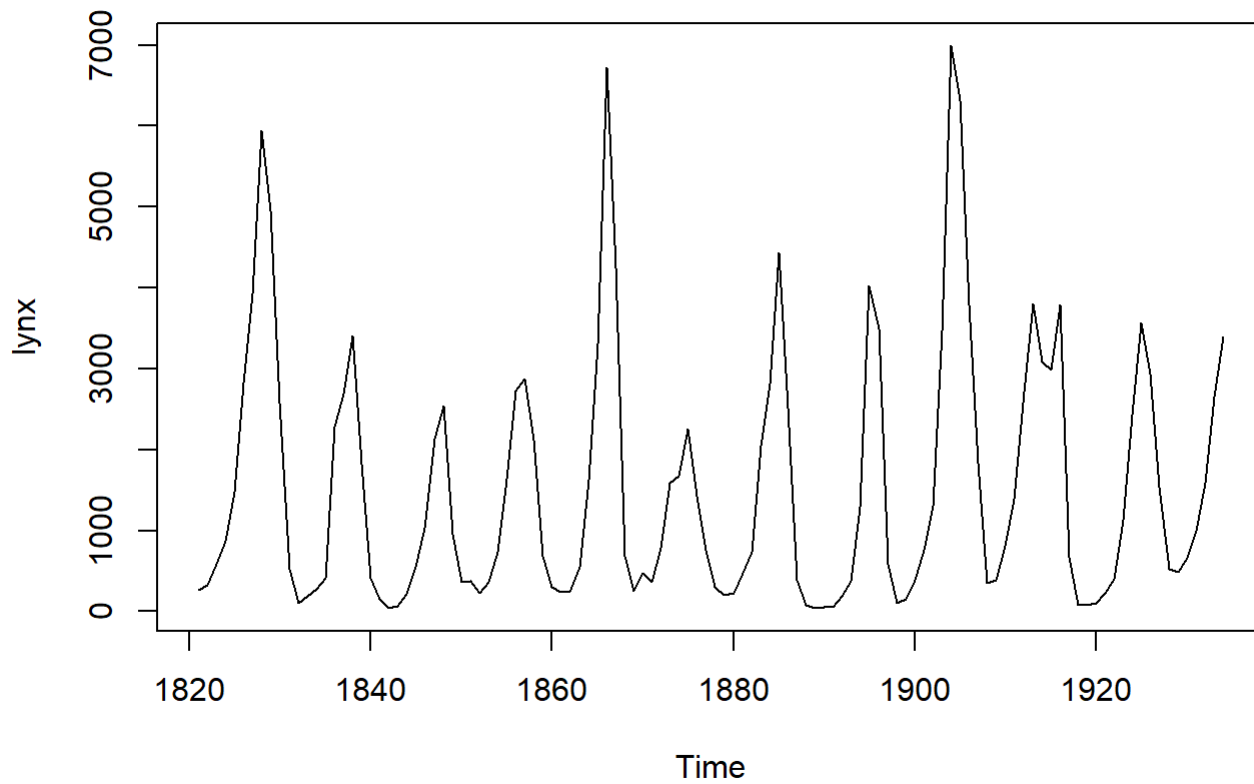
Com vetores

```
altura = c(145,167,176,123,150)
largura = c(51,63,64,40,55)
plot(altura, largura)
```

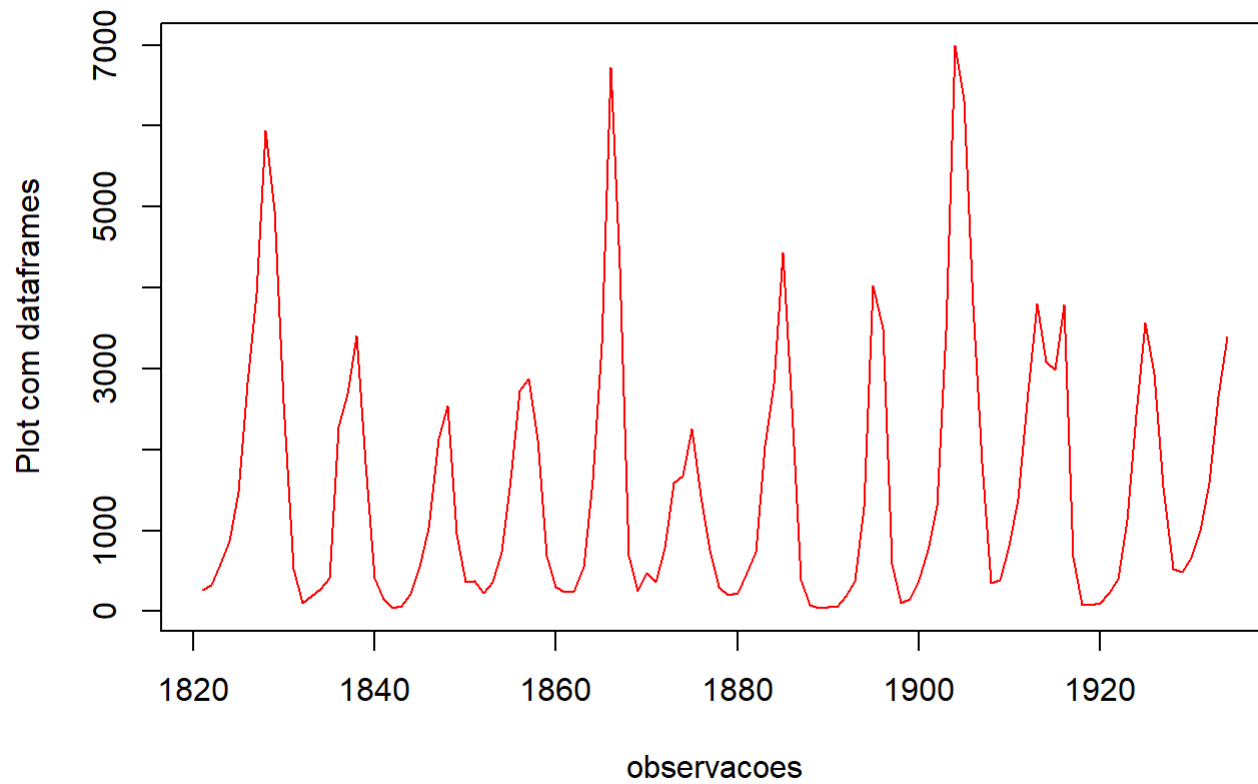


Plotando um dataframe

```
plot(lynx)
```

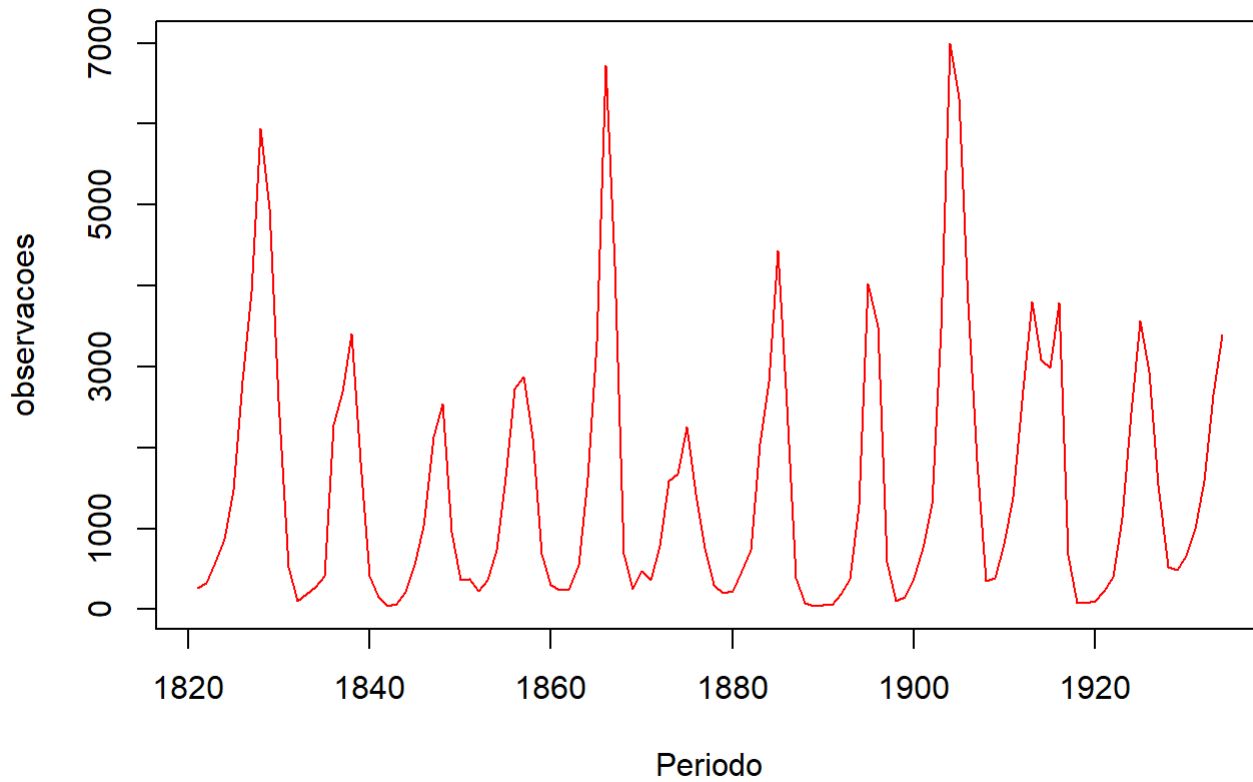



```
plot(lynx, ylab = "Plot com dataframes", xlab="observacoes", col = 'red',  
     col.main = 52, cex=1.5)
```



```
plot(lynx, main = "Plotando um dataframe" ,ylab = "observacoes", xlab="Período",  
     col = 'red', col.main = 52, cex=1.5)
```

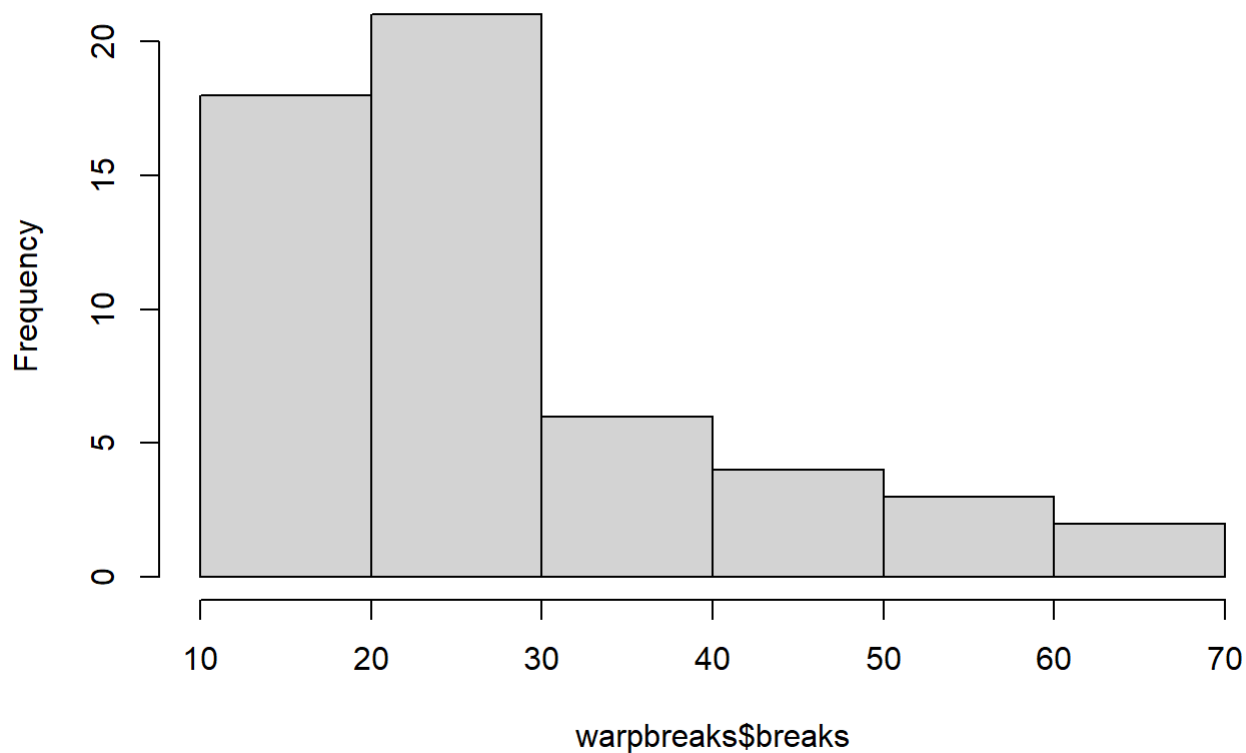
Plotando um dataframe



Plotando um histograma

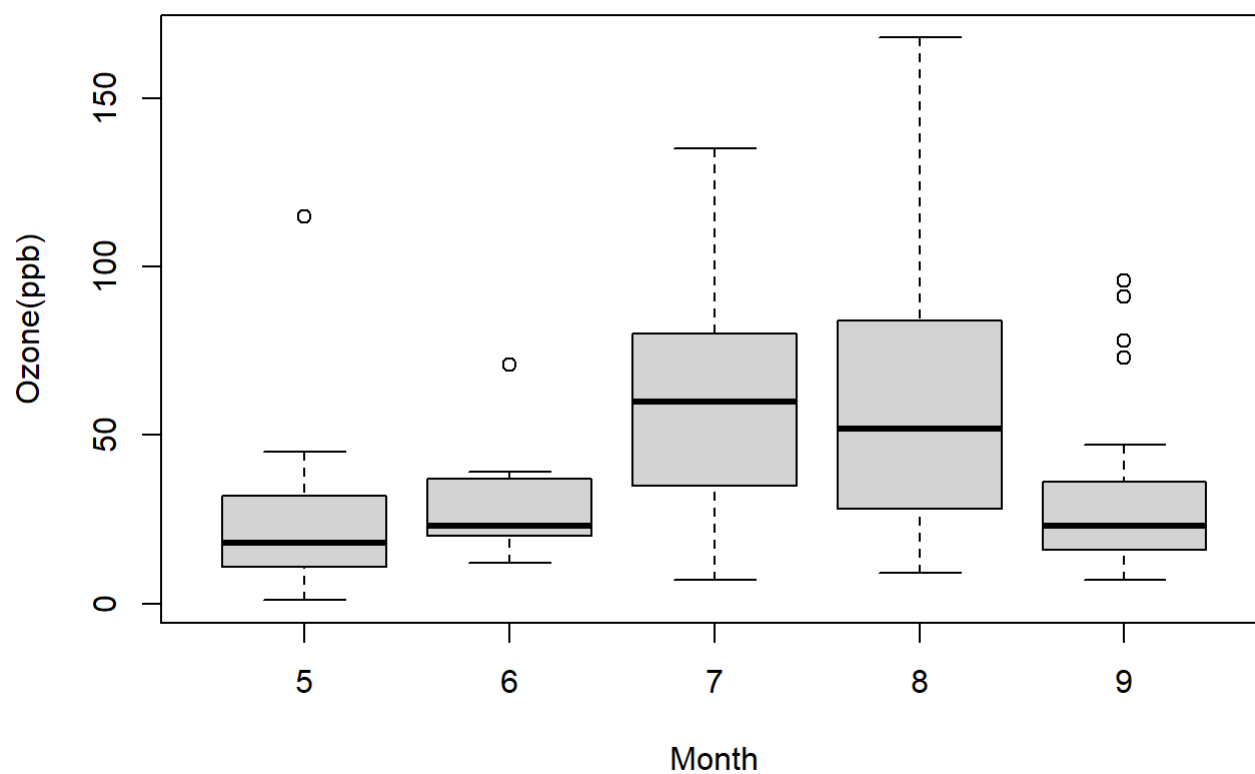
```
hist(warpbreaks$breaks)
```

Histogram of warpbreaks\$breaks



Plotando um boxplot

```
#airquality<-transform(airquality, Month=factor(Month)) #transformando a coluna Month em fator  
boxplot(Ozone~Month, airquality, xlab = 'Month', ylab='Ozone(ppb)') #plotando o boxplot
```



Parametros usados para plotagem de graficos

```
#col - cor do plotting
#lty - tipo de linha
#lwd - largura da linha
#pch - simbolo do plot
#xlab - label do eixo x
#ylab - label do eixo y
#las - orientacao dos labels
#bg - backgroud color
#mfrow - numero de plots por linha
#mfcoll - numero de plots por coluna
```

Funcoes basicas

```
#plots()
#lines()
#points()
#text()
#title()
```

Exemplos

```
par()
```

```
## $xlog
## [1] FALSE
##
## $ylog
## [1] FALSE
##
## $adj
## [1] 0.5
##
## $ann
## [1] TRUE
##
## $ask
## [1] FALSE
##
## $bg
## [1] "white"
##
## $bty
## [1] "o"
##
## $cex
## [1] 1
##
## $cex.axis
## [1] 1
##
## $cex.lab
## [1] 1
##
## $cex.main
## [1] 1.2
##
## $cex.sub
## [1] 1
##
## $cin
## [1] 0.15 0.20
##
## $col
## [1] "black"
##
## $col.axis
## [1] "black"
##
## $col.lab
## [1] "black"
##
## $col.main
## [1] "black"
##
## $col.sub
## [1] "black"
```

```
##
## $cra
## [1] 28.8 38.4
##
## $crt
## [1] 0
##
## $csi
## [1] 0.2
##
## $cxy
## [1] 0.02604167 0.06329115
##
## $din
## [1] 6.999999 4.999999
##
## $err
## [1] 0
##
## $family
## [1] ""
##
## $fg
## [1] "black"
##
## $fig
## [1] 0 1 0 1
##
## $fin
## [1] 6.999999 4.999999
##
## $font
## [1] 1
##
## $font.axis
## [1] 1
##
## $font.lab
## [1] 1
##
## $font.main
## [1] 2
##
## $font.sub
## [1] 1
##
## $lab
## [1] 5 5 7
##
## $las
## [1] 0
##
## $lend
## [1] "round"
```

```
##
## $lheight
## [1] 1
##
## $ljoin
## [1] "round"
##
## $lmitre
## [1] 10
##
## $lty
## [1] "solid"
##
## $lwd
## [1] 1
##
## $mai
## [1] 1.02 0.82 0.82 0.42
##
## $mar
## [1] 5.1 4.1 4.1 2.1
##
## $mex
## [1] 1
##
## $mfcol
## [1] 1 1
##
## $mfg
## [1] 1 1 1 1
##
## $mfrow
## [1] 1 1
##
## $mgp
## [1] 3 1 0
##
## $mkh
## [1] 0.001
##
## $new
## [1] FALSE
##
## $oma
## [1] 0 0 0 0
##
## $omd
## [1] 0 1 0 1
##
## $omi
## [1] 0 0 0 0
##
## $page
## [1] TRUE
```



```
##
## $pch
## [1] 1
##
## $pin
## [1] 5.759999 3.159999
##
## $plt
## [1] 0.1171429 0.9400000 0.2040000 0.8360000
##
## $ps
## [1] 12
##
## $pty
## [1] "m"
##
## $smo
## [1] 1
##
## $srt
## [1] 0
##
## $tck
## [1] NA
##
## $tcl
## [1] -0.5
##
## $usr
## [1] 0 1 0 1
##
## $xaxp
## [1] 0 1 5
##
## $xaxs
## [1] "r"
##
## $xaxt
## [1] "s"
##
## $xpd
## [1] FALSE
##
## $yaxp
## [1] 0 1 5
##
## $yaxs
## [1] "r"
##
## $yaxt
## [1] "s"
##
## $ylbias
## [1] 0.2
```

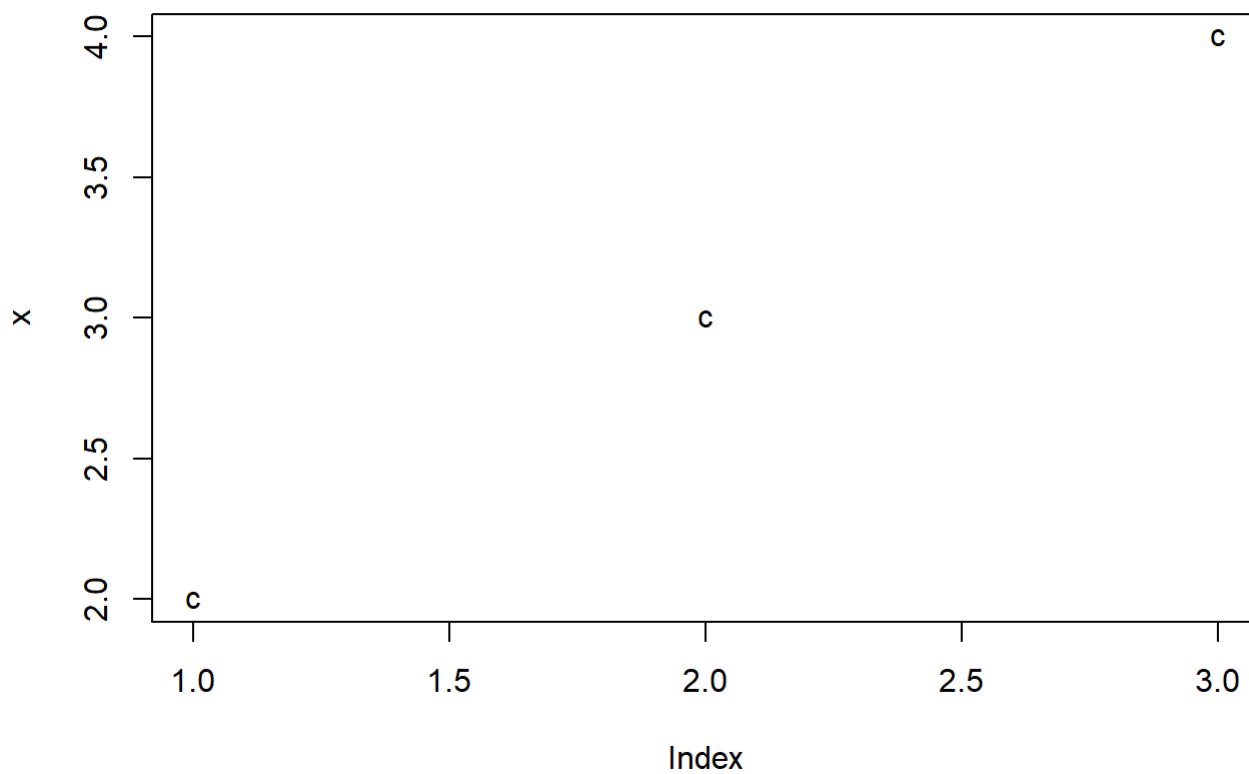
```
par('pch')
```

```
## [1] 1
```

```
par('lty')
```

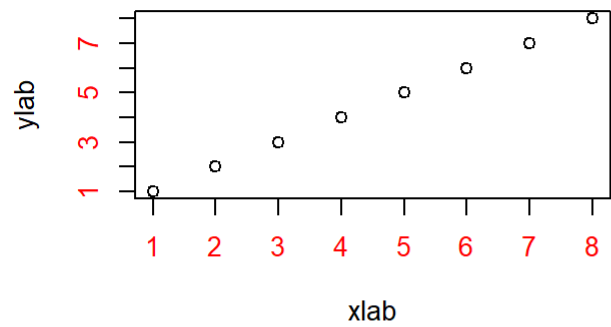
```
## [1] "solid"
```

```
x=2:4  
plot(x, pch='c')
```

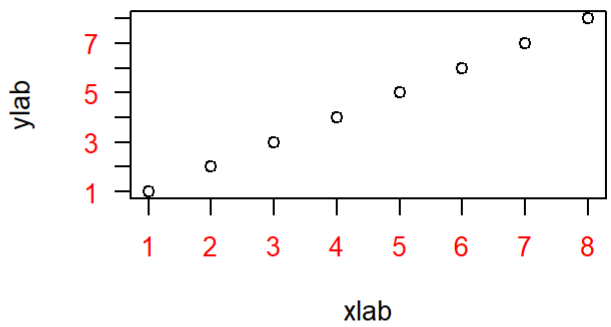


```
par(mfrow=c(2,2), col.axis='red')  
plot(1:8, las='0', xlab = 'xlab', ylab = 'ylab', main = 'LAS 0')  
plot(1:8, las='1', xlab = 'xlab', ylab = 'ylab', main = 'LAS 1')  
plot(1:8, las='2', xlab = 'xlab', ylab = 'ylab', main = 'LAS 2')  
plot(1:8, las='3', xlab = 'xlab', ylab = 'ylab', main = 'LAS 3')
```

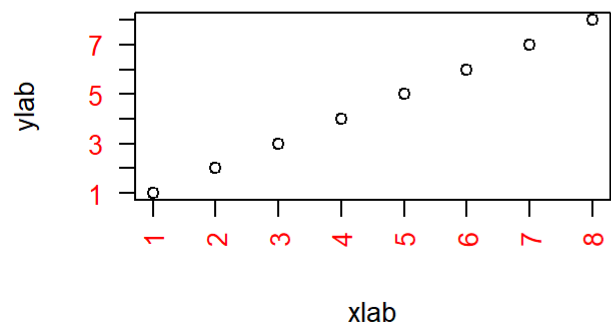
LAS 0



LAS 1



LAS 2



LAS 3

