

XV Semana Nacional de Ciência e Tecnologia de Itacoatiara - XV SNCT-ITA

Introdução ao R

Prof. Dr. Ricardo Perdiz (Luz da Floresta)



Objetivo

Ensinar conhecimentos básicos necessários para manipulação de dados em ambiente R.



O que é o R?

R é uma **linguagem** e **ambiente** para análise estatística e produção de gráficos de alta qualidade.

R pode ser utilizado para manipular e analisar dados, criar gráficos, e simulação computacional.

R pode ter extensão de funções através do uso de **pacotes**, baixados por meio do **Comprehensive R Archive Network (CRAN)**

Por que usar o R???

1. É grátis
2. Gráficos excelentes
3. Diversidade de ferramentas (=pacotes) para fazer análises estatísticas
4. Liberdade para criação de funções

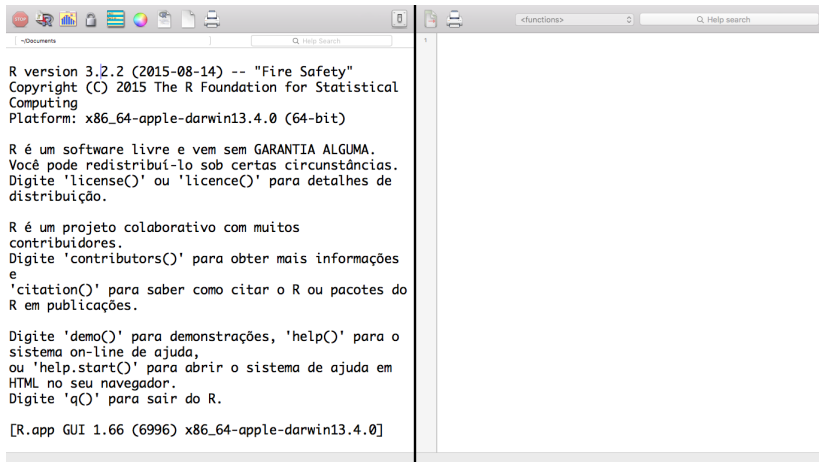
Como obter o R?

Visitar a página do projeto www.r-project.org

Verificar o tutorial para instalação na **página da disciplina**

Qual é a cara do R?

R no MAC OS X



```
R version 3.2.2 (2015-08-14) -- "Fire Safety"
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin13.4.0 (64-bit)

R é um software livre e vem sem GARANTIA ALGUMA.
Você pode redistribuí-lo sob certas circunstâncias.
Digite 'license()' ou 'licence()' para detalhes de
distribuição.

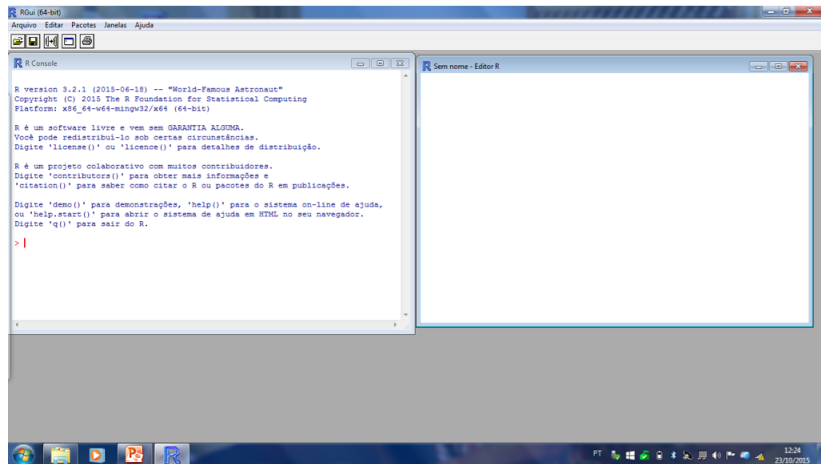
R é um projeto colaborativo com muitos
contribuidores.
Digite 'contributors()' para obter mais informações
e
'citation()' para saber como citar o R ou pacotes do
R em publicações.

Digite 'demo()' para demonstrações, 'help()' para o
sistema on-line de ajuda,
ou 'help.start()' para abrir o sistema de ajuda em
HTML no seu navegador.
Digite 'q()' para sair do R.

[R.app GUI 1.66 (6996) x86_64-apple-darwin13.4.0]
```

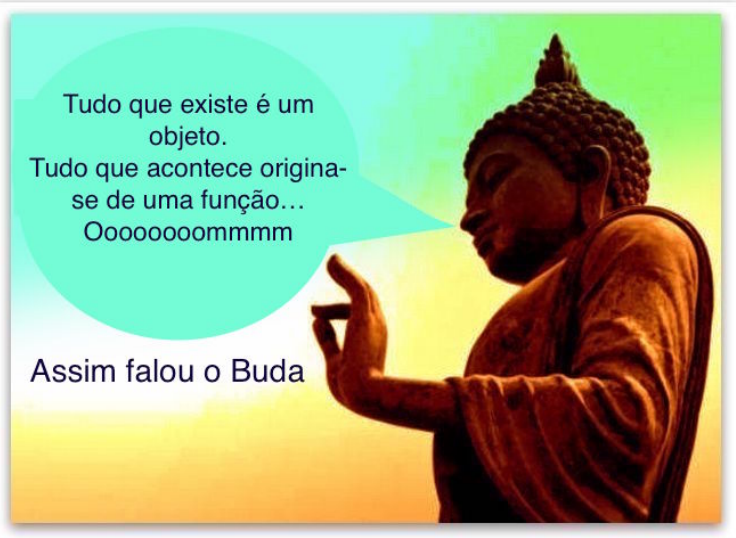
Qual é a cara do R?

R no Windows



Como é a linguagem do R

- *'Orientada a objetos' - object-oriented programming*

A photograph of a Buddha statue in profile, facing left, set against a bright, hazy background of yellow and green. A large, light blue speech bubble originates from the Buddha's mouth. Inside the bubble, there is text in Portuguese. Below the bubble, the text 'Assim falou o Buda' is written.

Tudo que existe é um
objeto.
Tudo que acontece origina-
se de uma função...
Oooooooooommm

Assim falou o Buda

Como é a linguagem do R

- ▶ Escrever em R é como escrever em português, inglês etc
 - ▶ Comunicação - ação - meios a um fim
- ▶ Programar, em geral, consiste em definir *funções* e *variáveis*
- ▶ *Variáveis* podem ser números ou palavras; guardamos esses valores para uso posterior
- ▶ *Funções* executam comandos, por meio de *argumentos*
- ▶ Comandos geram resultados

Variáveis

Variáveis em R são assinaladas utilizando a notação `** <- **`

```
> string <- 'Olá, Roraima!'
> meu_numero <- 25
```

```
> print(string)
```

```
## [1] "Olá, Roraima!"
```

```
> print(meu_numero)
```

```
## [1] 25
```

Funções e argumentos

- ▶ **Função** -> é um elemento em R que requer uma ação do computador. Cumpre determinada tarefa.
- ▶ **Argumento** -> especifica ou modifica como uma função trabalha. É especificado dentro de “parênteses” após o nome da função

```
> seq(from = 0,to = 10,by =1)
```

```
## [1] 0 1 2 3 4 5 6 7 8 9 10
```

Como é a linguagem do R

- Anatomia de uma linha de comando

```
> x <-sample(1:10,20,replace=TRUE)
> print(x)
```

```
## [1] 2 5 2 5 8 6 8 9 2 3 9 8 3 1 9 3 8 9 4 6
```

```
> mean(x)
```

```
## [1] 5.5
```

Detalhes *nomenclaturais* em R

- ▶ Separador de decimal é ponto (= "."); vírgula separa **argumentos**
- ▶ O símbolo de *igual* "=" atribui valores aos **argumentos**
- ▶ Nomes de **argumentos** podem ser omitidos, desde que os valores estejam seguindo a ordem pré-determinada da função. Ver o '?help'

O meu melhor amigo é o **HELP**

Dúvidas são comuns quando programamos. O R possui um excelente recurso, o '**help**'. Uma das maneiras é digitar "?" + "*nomeDaFunção*"

```
?seq
```

O meu melhor amigo é o **HELP**

`seq {base}`

R Documentation

Sequence Generation

Description

Generate regular sequences. `seq` is a standard generic with a default method. `seq.int` is a primitive which can be much faster but has a few restrictions. `seq_along` and `seq_len` are very fast primitives for two common cases.

Usage

```
seq(...)  
  
## Default S3 method:  
seq(from = 1, to = 1, by = ((to - from)/(length.out - 1)),  
    length.out = NULL, along.with = NULL, ...)  
  
seq.int(from, to, by, length.out, along.with, ...)  
  
seq_along(along.with)  
seq_len(length.out)
```

Arguments

Como é a linguagem do R

```
> a <- "Eu "  
> b <- "sou "  
> c <- "Feliz"  
> print(a)
```

```
## [1] "Eu "
```

```
> b
```

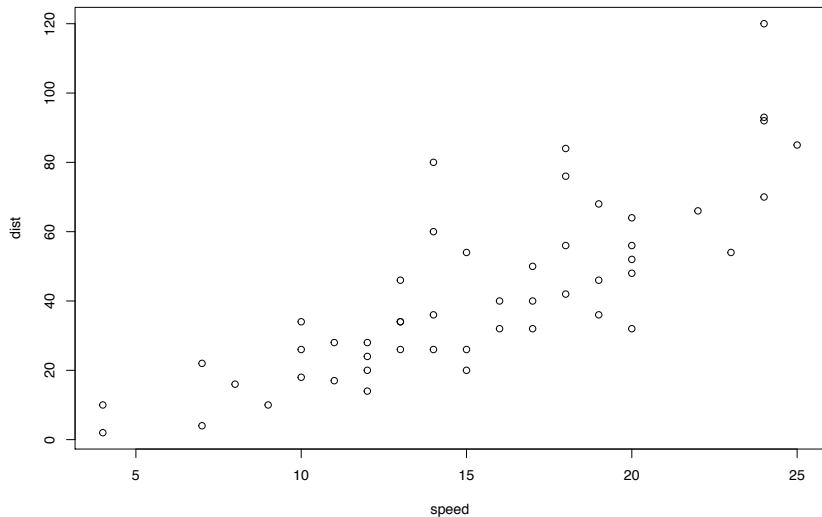
```
## [1] "sou "
```

```
> print(paste(a,b,c,sep=""))
```

```
## [1] "Eu sou Feliz"
```

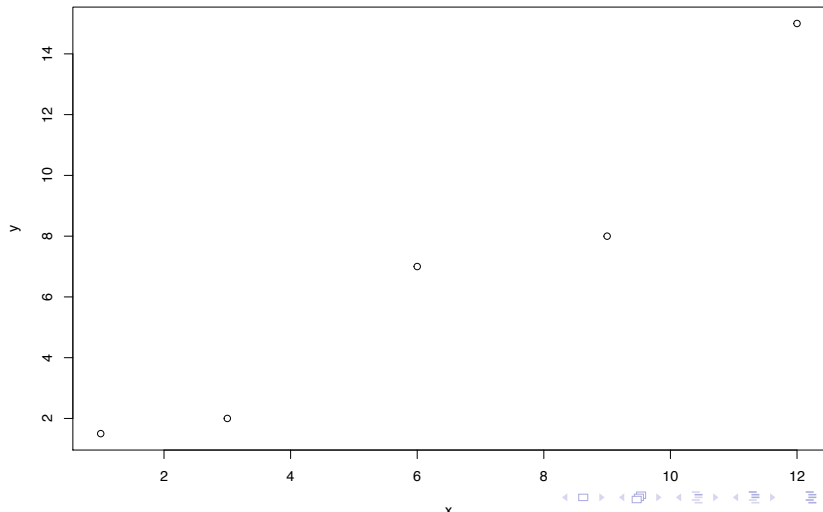

Gráficos em R

► Plots

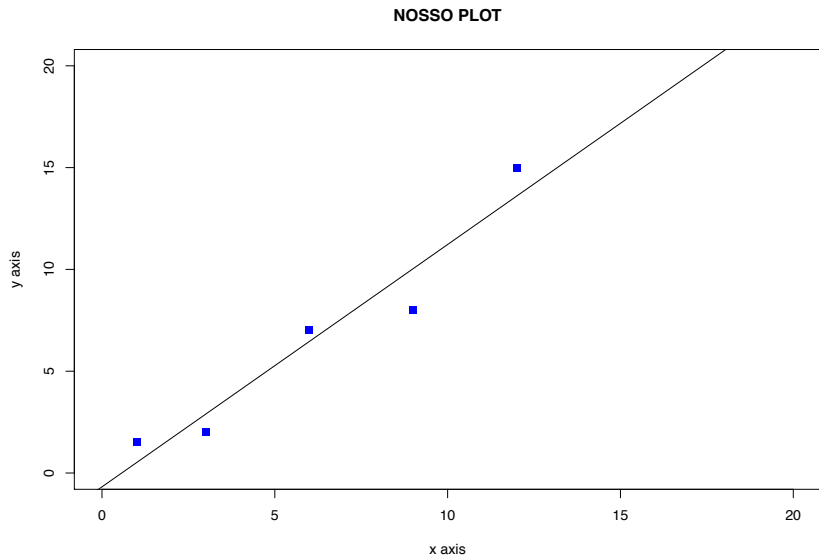


Gráficos em R

```
x <- c(1,3,6,9,12)
y <- c(1.5,2,7,8,15)
plot(x,y)
```

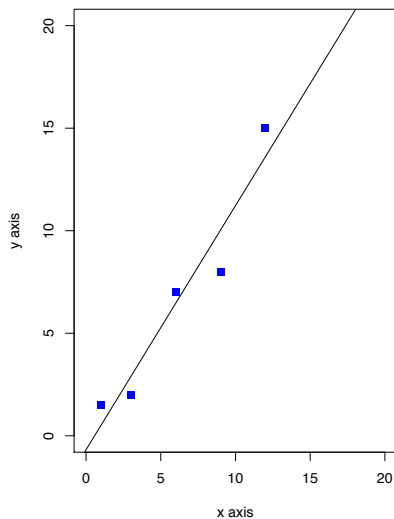
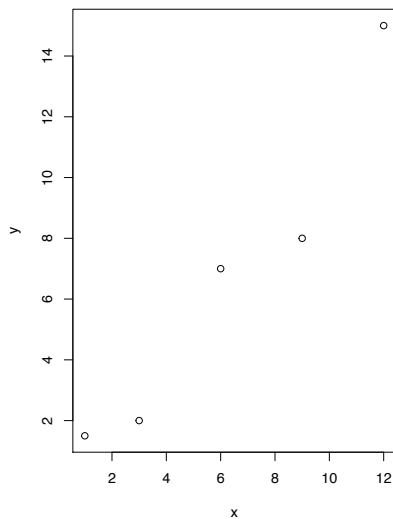


Gráficos em R



Gráficos em R

NOSSO PLOT



Gráficos em R

- ▶ Com o código abaixo...

```
map(regions=países,col='gray95',fill=T,xlim=x1,ylim=y1)
abline(h=0,lwd=0.5,lty="dotted")
for(i in 1:length(spp)){
  points(cpa$Longitude[cpa$SP==spp[i]],cpa$Latitude
        [cpa$SP==spp[i]],pch=pontos[i],col=
        cores.map[i],bg=cores.map[i],cex=
        cex.ponto[val.exp[i]])
}
par(cex=1, las=1)
map.scale(-55, ratio=F, cex=1.2)
map.axes()
axis(side=4, las=1)
axis(side=3, las=1)
legend(posicoes[2], 11.5, legend=spp, pch=pontos, col=
      cores.map, pt.bg=cores.map, cex=1.2,
      x.intersp=0.4, text.font=3)
```

Gráficos em R

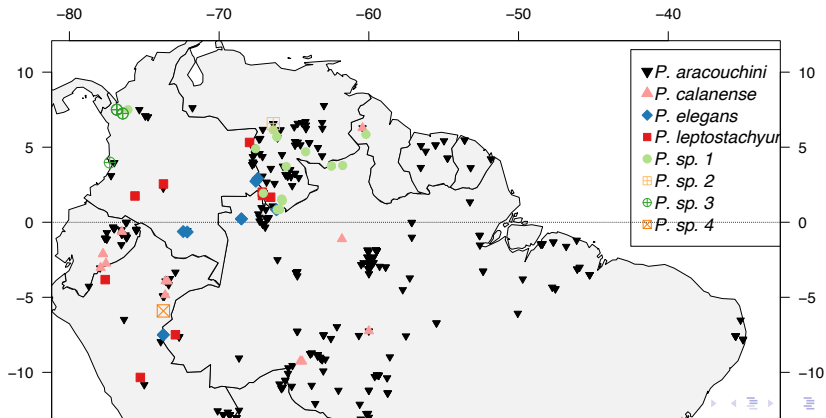
► ...temos um mapa

```
##
```

```
## # ATTENTION: maps v3.0 has an updated 'world' map.
```

```
## # Many country borders and names have changed since 199
```

```
## # Type '?world' or 'news(package="maps")'. See README_v
```



Vamos 'codeaR'!!!

