



#####

#

#R tutorial de bolso

#Renata de Lara Muylaert

#Nível “easy curious”

#

#####

#####

#versão 3 de Setembro de 2015



Sumário

| | |
|--|----|
| Introdução | 5 |
| #Escolhendo o diretório de trabalho..... | 5 |
| #utilizando janelas | 5 |
| #utilizando comandos..... | 5 |
| #Inserindo dados, criando um data.frame | 6 |
| #maneira 1: read.csv..... | 6 |
| #maneira 2: read.table | 6 |
| #maneira 3: read.delim..... | 6 |
| #maneira 4: clipboard | 6 |
| #arrumando alguma célula no seu data.frame | 7 |
| #Removendo colunas e linhas de diferentes maneiras | 7 |
| #retirando colunas..... | 7 |
| #retirando a segunda coluna | 7 |
| #subset..... | 8 |
| #linhas..... | 8 |
| #removendo linhas por indexação simples | 8 |
| #subset..... | 8 |
| #which | 9 |
| #Sobre NA..... | 9 |
| #retirando linhas com NA da sua tabela | 9 |
| #maneira 1 | 9 |
| #maneira 2: NA.omit() | 9 |
| #maneira 3: is.na()..... | 9 |
| #substituindo NA por 0 na sua tabela..... | 10 |
| #Binarizando seus dados | 10 |
| #binarizando com ifelse | 10 |
| #binarizando com decostand | 10 |
| # s2 Ifelse: Categorizando seus dados a partir de uma coluna de dados contínuos..... | 10 |
| #Exportando qualquer objeto como um arquivo TXT..... | 11 |



| | |
|--|----|
| #Sequências | 11 |
| #gerando sequências com diferentes distribuições | 11 |
| #Concatenando colunas em uma nova coluna | 12 |
| #Adicionando colunas na rabeira dos seus dados: | 12 |
| #Adicionando linhas na rabeira dos seus dados: | 12 |
| #Citando o R | 12 |
| #Blogs e sites legais..... | 13 |



Agradecimentos

Este tutorial foi elaborado em 2015 após algum treinamento autodidata em R e sofrimento em conjunto com meus amigos leecianos (principalmente Calebe Mendes e Maurício Vancine). Agradeço a Milton Cezar Ribeiro, Pavel Dodonov e Arleu Viana por todas as conversas e aprendizados sobre R. Marco Mello, Arleu Viana e Maurício Vancine revisaram esta versão.

Licença Creative Commons

O conteúdo deste guia pode ser reproduzido à vontade, desde que não seja modificado e não seja usado para fins comerciais. O guia pode ser citado utilizando meu nome (Renata de Lara Muylaert) e direcionando para o site no qual ele se encontra <http://www.leec.eco.br/>. Se este guia ajudar você a elaborar algum trabalho científico, por favor, mencione isso nos agradecimentos.





Introdução

Este tutorial simplificado é para você que precisa usar ou escolheu usar o R, ou que nunca consegue se lembrar de algumas coisinhas e fica empacado no seu data.frame. O tutorial não foi feito no formato de “.R” direto, pois queria que ele tivesse um visual mais amigável e um índice organizado. Além disso, este tutorial pode de fato ser impresso como uma versão de bolso. Em breve um tutorial nível “*normal curious*” será lançado na sequência deste. A ideia é que vocês possam ler o tutorial e aplicá-lo aos seus dados imediatamente. Ele não é um tutorial do tipo “*example*” do R e, portanto, não fornece dados *a priori*, o que caracteriza qualquer nível de tutorial “*curious*”. A ideia é que o leitor utilize os seus próprios dados e treine para memorizar os códigos, a fim de não precisar mais de um tutorial no nível “*easy curious*” e estar pronto para o nível “*normal curious*”. O data.frame imaginário básico que utilizamos tem o nome “dados”. O leitor deverá adaptá-lo aos seus dados, modificando o nome do objeto como preferir. Além disso, o leitor deve ter o mínimo discernimento sobre diferentes tipos de objetos no R e sua estrutura. Espero que gostem! Sugestões podem ser enviadas para o e-mail renatamuy@gmail.com. Bom trabalho!

#Escolhendo o diretório de trabalho

#o diretório é a pasta na qual se encontram os seus dados, aqueles que você precisa para trabalhar! É no diretório de trabalho que também serão criados e armazenados os arquivos de saída das análises, caso você converta os objetivos em arquivos .txt (TXT).

#utilizando janelas

```
setwd(choose.dir())
```

#utilizando comandos

```
setwd("C://Users//Documents//LEEC")
```



#Inserindo dados, criando um data.frame

#maneira 1: read.csv

#Esse comando irá permitir que você encontre o arquivo através de uma janela.

```
dados<-read.csv(file.choose(), sep=";", header=TRUE)
```

#maneira 2: read.table

#separado por tab e com rótulos de coluna

```
dados<-read.table("meusdados.txt", header=T, sep="\t")
```

#maneira 3: read.delim

#com nome de linhas e de colunas dessa vez...

```
dados<- read.delim("meusdados.txt", row.names=1, header=TRUE)
```

#maneira 4: clipboard

#dados estão no bloco de notas

#selecione seus dados no bloco de notas

#aperte copy clicando sobre seus dados no bloco de notas

#dê o comando no r

```
dados<-read.delim("clipboard", header=TRUE)
```

#O comando `str(dados)` é muito útil para ver se você importou seus dados corretamente!

#Se você estiver trabalhando no seu script no Windows, aperte control+R para mandar rodar seu código. Se você estiver em um Mac, use command+enter.



#arrumando alguma célula no seu data.frame

```
fix(dados)
```

```
edit(dados)
```

#Removendo colunas e linhas de diferentes maneiras

#retirando colunas

```
dados$colunaquequerotirar<- NULL
```

#retirando a segunda coluna

#maneira 1

```
dados[2] <- NULL
```

#maneira 2

```
dados <- dados[,-2]
```

#maneira 3

```
dados <- dados[-2] #o segundo elemento em um data.frame é a  
segunda #coluna
```

#Removendo múltiplas colunas em sequência

```
dados[1:2] <- list(NULL) # funciona
```

```
dados[1:2] <- NULL # não funciona!
```

#Retirando colunas 2 e 3. Cautela ao fazer subset de uma matrix, pois
você pode acabar tendo um vetor

```
dados <- dados[,-(2:3)] # é um vector
```

```
dados <- dados[,-(2:3), drop=FALSE] # ainda é um data.frame
```



#subset

#para remover uma sequência, entre colunas 1 e 3

```
dados2 <- subset( dados, select = -c( 1:3 ))
```

#Removendo colunas específicas, que estão em sequência

```
dados2 <- subset( dados, select = -c( deessacoluna : aessacoluna )  
#nomes sem aspas mesmo
```

#Removendo colunas específicas, que não necessariamente estão em sequência

```
dados2 <- subset( dados, select = -c(essacoluna, essacoluna ) ) #nome  
da coluna sem aspas!
```

#linhas

#removendo linhas por indexação simples

#removendo a linha 1

```
dados = dados[-1,]
```

Removendo linhas 2 e 3

```
dados <- dados[c(-2,-3),]
```

#só as que tem valor F em saocarlos

```
dados[dados$local=="saocarlos" & dados$h=="F",]
```

#só as diferentes de cinquenta em saocarlos

```
dados[dados$local=="saocarlos" & dados$area!= 50,]
```

#só as linhas que tem carrapato diferente de zero e com área maior que 150 no paisagem saocarlos

```
dados[dados$paisagem=="saocarlos" & dados$carrapato != 0  
& dados$areaha>150,]
```

#subset

#Removendo ou mantendo linhas de um determinado valor, que no caso são os fragmentos F1

```
dados_f1= subset(dados, fragmento=="F1") #uso aspas, pois é factor
```

#Removendo ou mantendo linhas de um determinado valor, que no caso são #valores diferentes de zero

```
dados_sem_zero= subset(dados, remocao!=0) #s2 subset!
```



#which

#Removendo linhas de uma categoria específica

```
#data.frame dados
```

```
dados
```

| Local | Ponto | Riqueza |
|-------|--------|---------|
| A | dentro | 1 |
| A | fora | 5 |
| B | dentro | 3 |
| B | fora | 14 |
| C | dentro | 6 |
| C | fora | 11 |

```
#Retirando todos os pontos de dentro
```

```
dentro=which(dados$Ponto!="dentro")
```

```
dentro=dados[dentro,]
```

| Local | Ponto | Riqueza |
|-------|-------|---------|
| A | fora | 5 |
| B | fora | 14 |
| C | fora | 11 |

#Sobre NA

#retirando linhas com NA da sua tabela

#maneira 1

```
Linhas_com_NA <- apply(dados, 1, function(x){any(is.na(x))})
```

```
#retirando linhas com NA
```

```
Dados_sem_NA<- dados[!Linhas_com_NA ,]
```

#maneira 2: NA.omit()

```
na.omit(dados)
```

#maneira 3: is.na()

```
Dados_sem_NA_para_uma_coluna<-dados[!is.na(dados$umacoluna),]
```



#substituindo NA por 0 na sua tabela

```
#is.na()
```

```
dados[is.na(dados)]<-0
```

#Binarizando seus dados

#binarizando com ifelse

#Caso dados\$palm sejam maiores ou iguais a 1, faça-os ser igual a 1, caso contrário, faça-os serem zero!

```
dados$palm_bin<- ifelse(dados$palm>= 1, 1, 0)
```

```
dados$palm_bin
```

#binarizando com decostand

```
Install.packages("vegan")
```

```
require(vegan)
```

```
dados_bin<- decostand(dados, "pa")
```

s2 Ifelse: Categorizando seus dados a partir de uma coluna de dados contínuos

```
#criando a coluna
```

```
dados$F150<- NULL
```

```
#Quero separar minha coluna que contém área de fragmentos em três  
#categorias: 1)"menor_que_60": áreas com menos de 60 há, 2)  
"intermediário" #áreas iguais ou maiores a 60 há e menores ou iguais a  
150 e 3) #"maior_que_150" áreas maiores que 150 ha.
```

```
dados$F_150 <- ifelse(dados$A_FRAG1 < 60, "menor que60 ha",
```

```
ifelse(dados$A_FRAG1 >= 60 & dados$A_FRAG1 <=150 , "intermediario",
```

```
ifelse(dados$A_FRAG1 > 150, "maior que 150 ha", NA)))
```



#Exportando qualquer objeto como um arquivo TXT

```
#quote=FALSE quer dizer "sem aspas", ou sem aspas rs..  
#append= FALSE cria um novo arquivo  
#sep="" garante que haverá separação por espaço  
write.table(dados, file="exportei.txt", sep= " ", quote= FALSE, append=  
FALSE, na= "NA")
```

#Sequências

```
#Função seq, com os argumentos: de, até, de quanto em quanto.  
seq(from=0, to=10, by=0.1)  
#simplificada  
seq(0,10,0.1)  
#você pode usar o argumento length.out= quantos elementos terá a  
#sequência  
d=seq(0,10, length.out=13) # a sequência terá 13 elementos
```

#gerando sequências com diferentes distribuições

```
#gerando números, distribuição normal, com média igual a 1 e desvio  
igual a 1
```

```
rnorm(n=15, mean=1, sd=1)
```

```
#A função runif gera uma distribuição uniforme
```

```
#15 valores, mínimo podendo valer 0, máximo podendo valer 2
```

```
runif(n=15, min=0, max=2)
```



#Concatenando colunas em uma nova coluna

```
dados$Coisas_concatenadas<-NULL
```

```
#Dados representados
```

| Local | Ponto | Coisas_concatenadas |
|-------|-------|---------------------|
| 1 | 3 | NULL |

```
#concatenando
```

```
dados$Coisas_concatenadas<- paste(dados$Local, dados$Ponto,  
sep="_")
```

```
#Representação final
```

| Local | Ponto | Coisas_concatenadas |
|-------|-------|---------------------|
| 1 | 3 | 1_3 |

#Adicionando colunas na rabeira dos seus dados:

```
Dados_com_coluna_a_mais<- cbind(dados, colunaquequeroadicionar)
```

#Adicionando linhas na rabeira dos seus dados:

```
Dados_com_linha_a_mais<- rbind(dados, linhaquequeroadicionar)
```

#Citando o R

```
citation()
```



#Blogs e sites legais

<http://recologia.com.br/>

<http://pt.stackoverflow.com/>

<http://rfunction.com/>

<https://cantinhodor.wordpress.com/>

<http://www.r-bloggers.com/>