

Abrindo dados no R

Leonardo Sangali Barone

March 27, 2017

Abrindo dados no R

Neste tutorial vamos cobrir uma série de métodos disponíveis para abrirmos arquivos de texto e excel no R. Vamos dar atenção aos argumentos das funções de forma a solucionar dificuldades de abertura de dados com diferentes características ou em sistemas operacionais variados.

Pacotes no R

Antes de avançarmos à tarefa principal, vamos aprender um pouco mais sobre pacotes. Já foi destacado diversas vezes que uma das vantagens do R é a existência de uma comunidade produtiva e que desenvolve continuamente novas funcionalidades, tudo em código aberto.

Para instalarmos um novo pacote de R que esteja disponível no CRAN – “The Comprehensive R Archive Network” – utilizamos a função *install.packages*. Veja o exemplo com o pacote *beepr*:

```
install.packages("beepr")
```

Note que o nome do pacote deve estar em parêntese. Além disso, é possível que você tenha sido perguntad@ sobre de qual servidor do CRAN você quer baixar o pacote. A escolha em nada muda o resultado, exceto pelo tempo de duração.

Uma vez que um pacote foi instalado, ele está disponível em seu computador, mas não ainda para uso. Apenas depois de executarmos a função *library* é que teremos o pacote em nossa “biblioteca” de funções.

```
library(beepr)
```

Você pode dispensar as aspas ao usar a função *library*, pois é opcional. A função *require* é semelhante a *library* e a ignoraremos.

Abrindo dados com sa funções do pacote utils

Quando você inicia uma nova sessão de R, alguns pacotes já estão automaticamente carregados. *utils* é um deles, e ele contém as funções mais conhecidas de abertura de dados em arquivos de texto.

A principal função é *read.table*. Use a função *args* para explorar seus argumentos:

```
args(read.table)
```

```
## function (file, header = FALSE, sep = "", quote = "\"'", dec = ".",
##     numerals = c("allow.loss", "warn.loss", "no.loss"), row.names,
##     col.names, as.is = !stringsAsFactors, na.strings = "NA",
##     colClasses = NA, nrows = -1, skip = 0, check.names = TRUE,
##     fill = !blank.lines.skip, strip.white = FALSE, blank.lines.skip = TRUE,
##     comment.char = "#", allowEscapes = FALSE, flush = FALSE,
##     stringsAsFactors = default.stringsAsFactors(), fileEncoding = "",
##     encoding = "unknown", text, skipNul = FALSE)
## NULL
```

É imprescindível que a função *read.table* receba como primeiro argumento um arquivo de dados. Note que o caminho para o arquivo deve estar completo (ex: “C:\\User\\Documents\\file.txt”) ou ele deve estar no seu **working directory** (wd). Mas como eu descubro meu wd?

```
getwd()
```

E como eu altero meu wd?

```
setwd("C:\\User\\Documents")
```

Simples e muito útil para evitar escrever “labirintos de pastas” ao importar dados.

Um detalhe fundamental para quem usa Windows: há uma