

# Curso de R para meteorologia IAG/USP

*Sergio Ibarra-Espinosa, Amanda Rehbein e possivelmente outros (u r invited to collaborate)*

*2018-04-29*



# Contents

<b>1</b>	<b>Pre-requisitos do sistema</b>	<b>5</b>
1.1	Pacotes usados neste curso . . . . .	5
1.2	Colaborar . . . . .	6
1.3	Aportar com dados . . . . .	6
<b>2</b>	<b>Intro</b>	<b>7</b>
<b>3</b>	<b>R!</b>	<b>9</b>
3.1	Objetos de R . . . . .	9
3.2	Classe . . . . .	9
3.3	Vetores . . . . .	9
3.4	Convertir objetos com <code>as</code> . . . . .	10
3.5	Matrizes e a função <code>matrix</code> . . . . .	10
3.6	Array . . . . .	10
3.7	<code>list</code> . . . . .	12
3.8	Tempo e Data . . . . .	12
3.9	Fatores . . . . .	13
3.10	<code>data.frames</code> . . . . .	14
<b>4</b>	<b>Importando e exportando dados em R</b>	<b>17</b>
4.1	<code>data.frames</code> . . . . .	17
4.2	<code>data.table</code> . . . . .	21
<b>5</b>	<b>Applications</b>	<b>23</b>
5.1	Example one . . . . .	23
5.2	Example two . . . . .	23
<b>6</b>	<b>Final Words</b>	<b>25</b>



# Chapter 1

## Pre-requisitos do sistema

Em Windows, instale além do R, Rtools <https://cran.r-project.org/bin/windows/Rtools/>

Em MAC instale netcdf e:

```
brew unlink gdal
brew tap osgeo/osgeo4mac && brew tap --repair
brew install proj
brew install geos
brew install udunits
brew install gdal2 --with-armadillo --with-complete --with-libkml --with-unsupported
brew link --force gdal2
```

Em Ubuntu:

```
- sudo add-apt-repository ppa:ubuntugis/ubuntugis-unstable --yes
- sudo apt-get --yes --force-yes update -qq
# install tmap dependencies
- sudo apt-get install --yes libprotobuf-dev protobuf-compiler libv8-3.14-dev
# install tmap dependencies; for 16.04 libjq-dev this ppa is needed:
- sudo add-apt-repository -y ppa:opencpu/jq
- sudo apt-get --yes --force-yes update -qq
- sudo apt-get install libjq-dev
# units/udunits2 dependency:
- sudo apt-get install --yes libudunits2-dev
# sf dependencies:
- sudo apt-get install --yes libproj-dev libgeos-dev libgdal-dev libnetcdf-dev netcdf-bin gdal-bin
```

### 1.1 Pacotes usados neste curso

Para fazer este curso instale os seguintes pacotes como indicado:

```
install.packages("devtools")
devtools::install_github("tidyverse/tidyverse")
devtools::install_github("r-spatial/sf")
devtools::install_github("r-spatial/mapview")
devtools::install_github("r-spatial/stars")
install.packages(c("raster", "sp", "rgdal", "maptools", "ncdf4"))
install.packages(c("cptcity", "data.table", "openair"))
```

- devtools é um pacote para instalar pacotes de diferentes repositórios
- tidyverse é o universo de pacotes do Hadley Wickham. A instalação tem que ser usando devtools, pois precisamos plotar os objetos espaciais sf usando geom\_sf.
- sf e mapview, stars, raster, sp, rgdal e maptools são para a parte espacial. Lembrar que os objetos em meteorologias são espaço-temporais.
- ncdf4 é um pacote para manipular arquivos NetCDF.
- cptcity é um pacote que tem 7140 paletas de cores do arquivo web cpt-city (<http://soliton.vm.bytemark.co.uk/pub/cpt-city/index.html>).
- openair é um pacote para trabalhar com dados de qualidade do ar e meteorologia.

Se faltarem dependências de sistema, instale elas e instale os pacotes.

## 1.2 Colaborar

A forma preferida de colaboração é com pull-requests em <https://github.com/ibarraespinosa/cursoR/pull/new/master>. Lembre de aplicar a Guia de Estilo de R de Google (<https://google.github.io/styleguide/Rguide.xml>) ou com o formato de formatR <https://yihui.name/formatr/>. Em poucas palavras, lembre que seu código vai ser lido por seres humanos. Se quiser tem acesso no repositório deste curso, me contate. Tem um botão para editar qualquer página.

## 1.3 Aportar com dados

Se você tem dados para fazer este curso mais legal, por favor, edite este arquivo e com pull request, eu vou fazer um merge para poder.

1. NCEP: [ftp://nomads.ncdc.noaa.gov/GFS/analysis\\_only/](ftp://nomads.ncdc.noaa.gov/GFS/analysis_only/)
- 2.
- 3.

## Chapter 2

# Intro

Este curso é para pos, então vamos ver conteúdo rapidamente e se não dá tempo, este curso está online no site <https://github.com/atmoschem/cursorIAG>.

Eu tento usar BASE sempre que posso, e se não dá aí vou para outros paradigmas.

Outros pacotes de BASE: utils, stats, datasets, graphics, grDevices, grid, methods, tools, parallel, compiler, splines, tcltk, stats4.

Veja outros pacotes.

Este curso está baseado no livro R Programming for Data Science.

Vamos usar Rstudio

### Dica:

- Se não sabe como usar uma função, escreva: `?função`.
- As funções têm argumentos, use **TAB** para ver eles numa função.

Vamos lá!





# Chapter 3

## R!

- Quase em qualquer sistema operacional mas eu vou focar em Linux.
- Muita documentação:
- Intro.
- I/O.
- Quer fazer um pacote? Veja, aqui e aqui.
- Stackoverflow provides a great source of resources.

### 3.1 Objetos de R

- Character a
- numeric 1
- integer 1
- complex 0+1i
- logical TRUE

### 3.2 Classe

`class` função permite ver a classe dos objetos

### 3.3 Vetores

- `c("A", "C", "D")`
- `1:5 = c(1, 2, 3, 4, 5)`
- `c(TRUE, FALSE)`
- `c(1i, -1i)`
- `c(1, "C", "D")` qual é a classe???
- `c(1, NA, "D")` qual é a classe???
- `c(1, NA, NaN)` qual é a classe???

### 3.4 Converter objetos com as

```
as.numeric(c(1, "C", "D"))
```

```
## Warning: NAs introduzidos por coerção
```

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

### 3.5 Matrizes e a função matrix

[linhas, colunas]

- permitidos elementos **da mesma classe!**

vamos ver os argumentos da função `matrix`

```
args(matrix)
```

```
## function (data = NA, nrow = 1, ncol = 1, byrow = FALSE, dimnames = NULL)
```

```
## NULL
```

usando TAB

```
(m <- matrix(data = 0, nrow = 4, ncol = 4))
```

```
##      [,1] [,2] [,3] [,4]
## [1,]  0   0   0   0
## [2,]  0   0   0   0
## [3,]  0   0   0   0
## [4,]  0   0   0   0
```

```
(m1 <- matrix(data = 1:(4*4), nrow = 4, ncol = 4))
```

```
##      [,1] [,2] [,3] [,4]
## [1,]  1   5   9  13
## [2,]  2   6  10  14
## [3,]  3   7  11  15
## [4,]  4   8  12  16
```

```
dim(m1)
```

```
## [1] 4 4
```

```
(m2 <- matrix(data = 1:(4*4), nrow = 4, ncol = 4, byrow = TRUE))
```

```
##      [,1] [,2] [,3] [,4]
## [1,]  1   2   3   4
## [2,]  5   6   7   8
## [3,]  9  10  11  12
## [4,] 13  14  15  16
```

### 3.6 Array

É como uma matriz de matrizes de matrizes de matrizes..... and so on.

```
args(array)
```

```
## function (data = NA, dim = length(data), dimnames = NULL)
## NULL
```

lembre usar TAB

```
(a <- array(data = 0, dim = c(1,1)))
```

```
##      [,1]
## [1,]    0
```

```
class(a)
```

```
## [1] "matrix"
```

```
(a <- array(data = 0, dim = c(1,1,1)))
```

```
## , , 1
##
##      [,1]
## [1,]    0
```

```
class(a)
```

```
## [1] "array"
```

```
(a <- array(data = 0, dim = c(2,2,2)))
```

```
## , , 1
##
##      [,1] [,2]
## [1,]    0    0
## [2,]    0    0
```

```
##
## , , 2
##
##      [,1] [,2]
## [1,]    0    0
## [2,]    0    0
```

```
(a <- array(data = 0, dim = c(2,4,4)))
```

```
## , , 1
##
##      [,1] [,2] [,3] [,4]
## [1,]    0    0    0    0
## [2,]    0    0    0    0
```

```
##
## , , 2
##
##      [,1] [,2] [,3] [,4]
## [1,]    0    0    0    0
## [2,]    0    0    0    0
```

```
##
## , , 3
##
##      [,1] [,2] [,3] [,4]
## [1,]    0    0    0    0
## [2,]    0    0    0    0
##
```

```
## , , 4
##
##      [,1] [,2] [,3] [,4]
## [1,]    0    0    0    0
## [2,]    0    0    0    0

dim(a)

## [1] 2 4 4

(a <- array(data = 0, dim = c(2, 2,2,2)))
```

### 3.7 list

As listas são como sacolas, e dentro delas, tu pode colocar mais sacolas... então, tu pode ter sacolas, dentro de sacolas, dentro de sacolas... ou

```
list(list(list(list(1)))
```

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

a diferença das matrizes, tu pode colocar qualquer coisa nas listas, por exemplo: funções, characters, etc.

```
(x <- list(1, "a", TRUE, 1 + 4i))
```

```
## [[1]]
## [1] 1
##
## [[2]]
## [1] "a"
##
## [[3]]
## [1] TRUE
##
## [[4]]
## [1] 1+4i
```

### 3.8 Tempo e Data

R tem classes de tempo e data:

```
(a <- ISOdate(year = 2018, month = 4, day = 5))
```

```
## [1] "2018-04-05 12:00:00 GMT"
```

```
class(a)
```

```
## [1] "POSIXct" "POSIXt"
```

```
(b <- ISOdate(year = 2018, month = 4, day = 5, tz = "Americas/Sao_Paulo"))
```

```
## [1] "2018-04-05 12:00:00 Americas"
```

tempo

```
(d <- ISOdatetime(year = 2018, month = 4, day = 5, hour = 0, min = 0, sec = 0,
  tz = "Americas/Sao_Paulo"))
```

```
## [1] "2018-04-05 Americas"
```

O pacote nanotime permite trabalhar com nano segundos.

Da pra fazer sequencias:

```
hoje <- Sys.time()
(a <- seq.POSIXt(from = hoje, by = 3600, length.out = 24))
```

```
## [1] "2018-04-29 18:46:07 -03" "2018-04-29 19:46:07 -03"
## [3] "2018-04-29 20:46:07 -03" "2018-04-29 21:46:07 -03"
## [5] "2018-04-29 22:46:07 -03" "2018-04-29 23:46:07 -03"
## [7] "2018-04-30 00:46:07 -03" "2018-04-30 01:46:07 -03"
## [9] "2018-04-30 02:46:07 -03" "2018-04-30 03:46:07 -03"
## [11] "2018-04-30 04:46:07 -03" "2018-04-30 05:46:07 -03"
## [13] "2018-04-30 06:46:07 -03" "2018-04-30 07:46:07 -03"
## [15] "2018-04-30 08:46:07 -03" "2018-04-30 09:46:07 -03"
## [17] "2018-04-30 10:46:07 -03" "2018-04-30 11:46:07 -03"
## [19] "2018-04-30 12:46:07 -03" "2018-04-30 13:46:07 -03"
## [21] "2018-04-30 14:46:07 -03" "2018-04-30 15:46:07 -03"
## [23] "2018-04-30 16:46:07 -03" "2018-04-30 17:46:07 -03"
```

funções bacana: **weekdays**, **month**, **julian**

```
weekdays(a)
```

```
## [1] "domingo" "domingo" "domingo" "domingo" "domingo" "domingo" "segunda"
## [8] "segunda" "segunda" "segunda" "segunda" "segunda" "segunda" "segunda"
## [15] "segunda" "segunda" "segunda" "segunda" "segunda" "segunda" "segunda"
## [22] "segunda" "segunda" "segunda"
```

```
months(a)
```

```
## [1] "abril" "abril" "abril" "abril" "abril" "abril" "abril" "abril"
## [9] "abril" "abril" "abril" "abril" "abril" "abril" "abril" "abril"
## [17] "abril" "abril" "abril" "abril" "abril" "abril" "abril" "abril"
```

```
julian(a) #olha ?julian... dias desde origin
```

```
## Time differences in days
## [1] 17650.91 17650.95 17650.99 17651.03 17651.07 17651.12 17651.16
## [8] 17651.20 17651.24 17651.28 17651.32 17651.37 17651.41 17651.45
## [15] 17651.49 17651.53 17651.57 17651.62 17651.66 17651.70 17651.74
## [22] 17651.78 17651.82 17651.87
## attr(,"origin")
## [1] "1970-01-01 GMT"
```

olha [https://en.wikipedia.org/wiki/Julian\\_day](https://en.wikipedia.org/wiki/Julian_day):

## 3.9 Fatores

Os **factors** podem ser um pouco infernais. Olha R INFERNO

Usados para representar categorias, ejemplo clasico para nos, dias da semana.

```

a <- seq.POSIXt(from = hoje, by = 3600, length.out = 24*7)
aa <- weekdays(a)
class(aa)

## [1] "character"

factor(aa)

## [1] domingo domingo domingo domingo domingo domingo segunda segunda
## [9] segunda segunda segunda segunda segunda segunda segunda segunda
## [17] segunda segunda segunda segunda segunda segunda segunda segunda
## [25] segunda segunda segunda segunda segunda segunda segunda terça terça
## [33] terça terça terça terça terça terça terça terça
## [41] terça terça terça terça terça terça terça terça
## [49] terça terça terça terça terça terça terça quarta quarta
## [57] quarta quarta quarta quarta quarta quarta quarta quarta
## [65] quarta quarta quarta quarta quarta quarta quarta quarta
## [73] quarta quarta quarta quarta quarta quarta quinta quinta
## [81] quinta quinta quinta quinta quinta quinta quinta quinta
## [89] quinta quinta quinta quinta quinta quinta quinta quinta
## [97] quinta quinta quinta quinta quinta quinta sexta sexta
## [105] sexta sexta sexta sexta sexta sexta sexta sexta
## [113] sexta sexta sexta sexta sexta sexta sexta sexta
## [121] sexta sexta sexta sexta sexta sexta sexta sábado sábado
## [129] sábado sábado sábado sábado sábado sábado sábado sábado
## [137] sábado sábado sábado sábado sábado sábado sábado sábado
## [145] sábado sábado sábado sábado sábado sábado sábado domingo domingo
## [153] domingo domingo domingo domingo domingo domingo domingo domingo
## [161] domingo domingo domingo domingo domingo domingo domingo domingo
## Levels: domingo quarta quinta sábado segunda sexta terça

```

olha os **Levels**

Então:

```

ab <- factor(x = aa,
             levels = c("Monday", "Tuesday", "Wednesday", "Thursday",
                       "Friday", "Saturday", "Sunday"))
levels(ab)

## [1] "Monday"    "Tuesday"   "Wednesday" "Thursday"  "Friday"    "Saturday"
## [7] "Sunday"

```

### 3.10 data.frames

*lembre ?data.frame*

São como planilha EXCEL.... mais o menos

É uma classe bem especial, tem elementos de matriz mas o modo é lista

```

(df <- data.frame(a = 1:3))

## a
## 1 1
## 2 2
## 3 3

```

```
names(df)
```

```
## [1] "a"
```

```
class(df)
```

```
## [1] "data.frame"
```

```
mode(df)
```

```
## [1] "list"
```

Então

```
nrow(df)
```

```
## [1] 3
```

```
ncol(df)
```

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

```
dim(df)
```

```
## [1] 3 1
```





## Chapter 4

# Importando e exportando dados em R

### 4.1 data-frames

Probavelmente um dos primeiros objetos que vamos usar quando começamos usar R. Pensa num data-frame como uma planilha de Libreoffice (o excel). Os data-frame pode ser criados como foi visto na seção anterior. O principal, é que temos varias funções para ler data-frames no R, entre elas

- `read.csv`
- `read.csv2`
- `read.table`

Agora vamos a ler dados do repositório usando `read.table`, mas primeiro vamos lembrar que se tu precisar ver a ajuda da função, tem que escrever no R `?read.table`. Então, agora vamos ver os argumentos da função:

```
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, nrow = -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
```

Aqui vem-se os valores default dos argumentos da função `read.table`. O terceiro argumento é `sep`, com valores por default = “,”.

```
df <- read.table("https://raw.githubusercontent.com/ibarraespinosa/cursor/master/dados/NOXIPEN2014.txt")
```

Agora vamos usar as funções `head` and `tail` para ver as primeiras e as ultimas 6 linhas do data-frame.

```
head(df)
```

```
##   TipodeRede TipodeMonitoramento      Tipo      Data Hora
## 2 Automático          CETESB Dados Primários 01/01/2014 01:00
## 3 Automático          CETESB Dados Primários 01/01/2014 02:00
## 4 Automático          CETESB Dados Primários 01/01/2014 03:00
## 5 Automático          CETESB Dados Primários 01/01/2014 04:00
## 6 Automático          CETESB Dados Primários 01/01/2014 05:00
## 7 Automático          CETESB Dados Primários 01/01/2014 06:00
```

```
##      CodigoEstação      NomeEstação      NomeParâmetro
## 2          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 3          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 4          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 5          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 6          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 7          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
##      UnidadeMedida MediaHoraria MediaMovel Valido
## 2          ppb          9          -      Não
## 3          ppb          9          -      Sim
## 4          ppb          5          -      Sim
## 5          ppb          4          -      Sim
## 6          ppb          5          -      Sim
## 7          ppb          5          -      Sim
```

```
tail(df)
```

```
##      TipodeRede TipodeMonitoramento      Tipo      Data      Hora
## 8577 Automático      CETESB Dados Primários 01/01/2015 19:00
## 8578 Automático      CETESB Dados Primários 01/01/2015 20:00
## 8579 Automático      CETESB Dados Primários 01/01/2015 21:00
## 8580 Automático      CETESB Dados Primários 01/01/2015 22:00
## 8581 Automático      CETESB Dados Primários 01/01/2015 23:00
## 8582 Automático      CETESB Dados Primários 01/01/2015 24:00
##      CodigoEstação      NomeEstação      NomeParâmetro
## 8577          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 8578          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 8579          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 8580          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 8581          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 8582          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
##      UnidadeMedida MediaHoraria MediaMovel Valido
## 8577          ppb          3          -      Sim
## 8578          ppb          8          -      Sim
## 8579          ppb          11         -      Sim
## 8580          ppb          11         -      Sim
## 8581          ppb          16         -      Sim
## 8582          ppb          NA         -      Sim
```

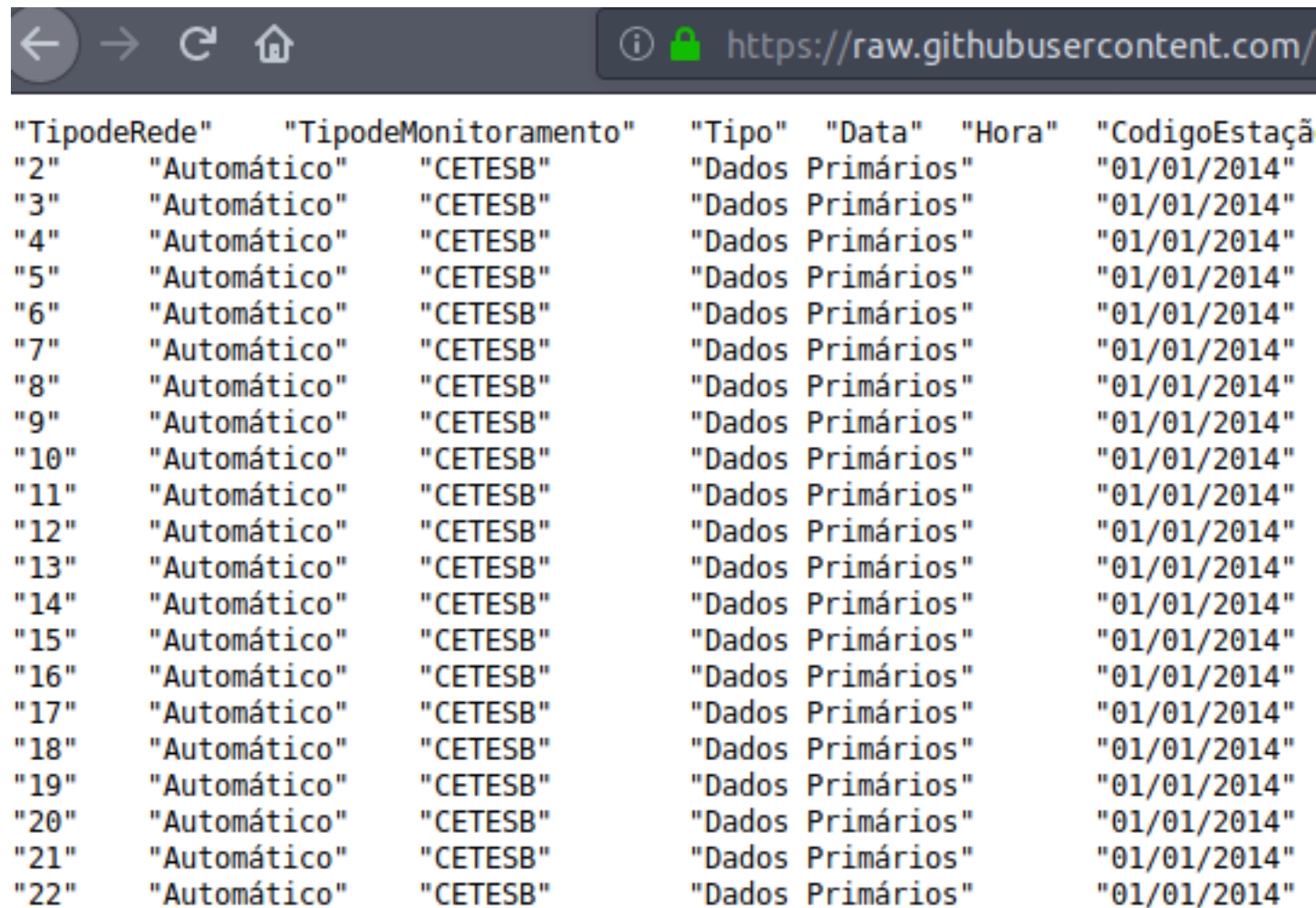
Agora vamos ler os mesmos dados com outro formato e testar se `read.table` funciona do mesmo jeito

```
df2 <- read.table("https://raw.githubusercontent.com/ibarraespinosa/cursoR/master/dados/NOXIPEN2014v2.t
# Error in scan(file = file, what = what, sep = sep, quote = quote, dec = dec, :
# linha 1 não tinha 6 elementos
```

Vemos a mensagem de error, mas o que quer dizer.

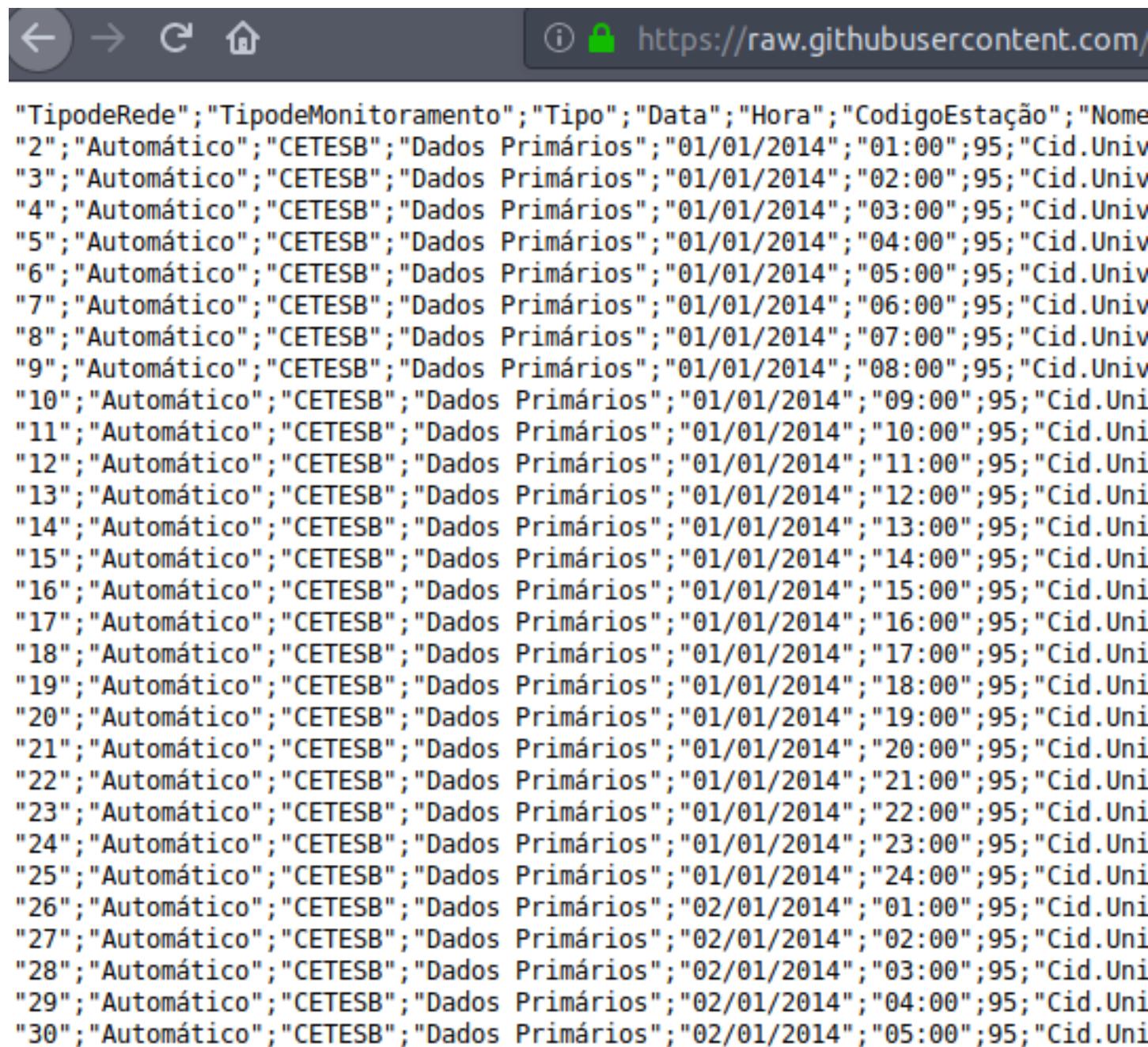
**Se tu receber um banco de dados tipo .txt e quer abrir no R... ABRE ELE COM BLOCO DE NOTAS PRIMEIRO!!!**

O primeiro arquivo:



"TipodeRede"	"TipodeMonitoramento"	"Tipo"	"Data"	"Hora"	"CodigoEstação"
"2"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"3"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"4"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"5"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"6"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"7"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"8"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"9"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"10"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"11"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"12"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"13"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"14"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"15"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"16"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"17"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"18"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"19"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"20"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"21"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"
"22"	"Automático"	"CETESB"	"Dados Primários"		"01/01/2014"

O segundo arquivo é:



```

"TipodeRede";"TipodeMonitoramento";"Tipo";"Data";"Hora";"CodigoEstação";"Nome
"2";"Automático";"CETESB";"Dados Primários";"01/01/2014";"01:00";95;"Cid.Univ
"3";"Automático";"CETESB";"Dados Primários";"01/01/2014";"02:00";95;"Cid.Univ
"4";"Automático";"CETESB";"Dados Primários";"01/01/2014";"03:00";95;"Cid.Univ
"5";"Automático";"CETESB";"Dados Primários";"01/01/2014";"04:00";95;"Cid.Univ
"6";"Automático";"CETESB";"Dados Primários";"01/01/2014";"05:00";95;"Cid.Univ
"7";"Automático";"CETESB";"Dados Primários";"01/01/2014";"06:00";95;"Cid.Univ
"8";"Automático";"CETESB";"Dados Primários";"01/01/2014";"07:00";95;"Cid.Univ
"9";"Automático";"CETESB";"Dados Primários";"01/01/2014";"08:00";95;"Cid.Univ
"10";"Automático";"CETESB";"Dados Primários";"01/01/2014";"09:00";95;"Cid.Uni
"11";"Automático";"CETESB";"Dados Primários";"01/01/2014";"10:00";95;"Cid.Uni
"12";"Automático";"CETESB";"Dados Primários";"01/01/2014";"11:00";95;"Cid.Uni
"13";"Automático";"CETESB";"Dados Primários";"01/01/2014";"12:00";95;"Cid.Uni
"14";"Automático";"CETESB";"Dados Primários";"01/01/2014";"13:00";95;"Cid.Uni
"15";"Automático";"CETESB";"Dados Primários";"01/01/2014";"14:00";95;"Cid.Uni
"16";"Automático";"CETESB";"Dados Primários";"01/01/2014";"15:00";95;"Cid.Uni
"17";"Automático";"CETESB";"Dados Primários";"01/01/2014";"16:00";95;"Cid.Uni
"18";"Automático";"CETESB";"Dados Primários";"01/01/2014";"17:00";95;"Cid.Uni
"19";"Automático";"CETESB";"Dados Primários";"01/01/2014";"18:00";95;"Cid.Uni
"20";"Automático";"CETESB";"Dados Primários";"01/01/2014";"19:00";95;"Cid.Uni
"21";"Automático";"CETESB";"Dados Primários";"01/01/2014";"20:00";95;"Cid.Uni
"22";"Automático";"CETESB";"Dados Primários";"01/01/2014";"21:00";95;"Cid.Uni
"23";"Automático";"CETESB";"Dados Primários";"01/01/2014";"22:00";95;"Cid.Uni
"24";"Automático";"CETESB";"Dados Primários";"01/01/2014";"23:00";95;"Cid.Uni
"25";"Automático";"CETESB";"Dados Primários";"01/01/2014";"24:00";95;"Cid.Uni
"26";"Automático";"CETESB";"Dados Primários";"02/01/2014";"01:00";95;"Cid.Uni
"27";"Automático";"CETESB";"Dados Primários";"02/01/2014";"02:00";95;"Cid.Uni
"28";"Automático";"CETESB";"Dados Primários";"02/01/2014";"03:00";95;"Cid.Uni
"29";"Automático";"CETESB";"Dados Primários";"02/01/2014";"04:00";95;"Cid.Uni
"30";"Automático";"CETESB";"Dados Primários";"02/01/2014";"05:00";95;"Cid.Uni

```

qual é a diferença?

Como vemos o segundo arquivo tem separação de “;”, então, temos que lero arquivo assim:

```
df2 <- read.table("https://raw.githubusercontent.com/ibarraespinosa/cursorR/master/dados/NOXIPEN2014v2.t
head(df2)
```

```
##   TipodeRede TipodeMonitoramento      Tipo      Data Hora
## 2 Automático          CETESB Dados Primários 01/01/2014 01:00
## 3 Automático          CETESB Dados Primários 01/01/2014 02:00
## 4 Automático          CETESB Dados Primários 01/01/2014 03:00
## 5 Automático          CETESB Dados Primários 01/01/2014 04:00
## 6 Automático          CETESB Dados Primários 01/01/2014 05:00
```

```
## 7 Automático          CETESB Dados Primários 01/01/2014 06:00
##  CodigoEstação        NomeEstação          NomeParâmetro
## 2          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 3          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 4          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 5          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 6          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 7          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
##  UnidadedeMedida MediaHoraria MediaMovel Valido
## 2          ppb          9          -      Não
## 3          ppb          9          -      Sim
## 4          ppb          5          -      Sim
## 5          ppb          4          -      Sim
## 6          ppb          5          -      Sim
## 7          ppb          5          -      Sim
```

```
tail(df2)
```

```
##      TipodeRede TipodeMonitoramento      Tipo      Data  Hora
## 8577 Automático          CETESB Dados Primários 01/01/2015 19:00
## 8578 Automático          CETESB Dados Primários 01/01/2015 20:00
## 8579 Automático          CETESB Dados Primários 01/01/2015 21:00
## 8580 Automático          CETESB Dados Primários 01/01/2015 22:00
## 8581 Automático          CETESB Dados Primários 01/01/2015 23:00
## 8582 Automático          CETESB Dados Primários 01/01/2015 24:00
##  CodigoEstação        NomeEstação          NomeParâmetro
## 8577          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 8578          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 8579          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 8580          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 8581          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
## 8582          95 Cid.Universitária-USP-Ipen NOx (Óxidos de Nitrogênio)
##  UnidadedeMedida MediaHoraria MediaMovel Valido
## 8577          ppb          3          -      Sim
## 8578          ppb          8          -      Sim
## 8579          ppb         11          -      Sim
## 8580          ppb         11          -      Sim
## 8581          ppb         16          -      Sim
## 8582          ppb         NA          -      Sim
```

## 4.2 data.table

`data.table` é um pacote que apresenta a classe `data.table`, que é como uma versão melhorada da classe `data.frame`. O termo específico é que `data-table` tem herencia (inherits) da classe `data.frame`.



## Chapter 5

# Applications

Some *significant* applications are demonstrated in this chapter.

### 5.1 Example one

### 5.2 Example two





## Chapter 6

# Final Words

We have finished a nice book.