HowTo Get data from the EUROSTAT

A. Blejec

November 18, 2014

Contents

```
Warning: package 'Hmisc' was built under R version 3.0.3
Warning: package 'lattice' was built under R version 3.0.3
```

1 Introduction

To make reproducible reports from the EUROSTAT data, I will explore the possibility to download data from their repository. They provide data in different formats, with footnotes and labels included or not included. The most promising for automatic retreival is the so called TSV format which is in fact tab separated file. This file is included in the .zip or .gz file.

2 Finding the file name

We will explore it later.

3 Get the data

Asumming that we have the desired file code, we can get it by composing the URL:

```
> fcode <- "tps00001"
> fcode <- "tps00025"  # Life expectancy at birth, by sex
> # http://epp.eurostat.ec.europa.eu/portal/page/portal/product_details/datase
>
> lfn <- paste(fcode, ".tsv", sep = "")
> upre <- "http://epp.eurostat.ec.europa.eu/NavTree_prod/everybody/BulkDownloa
> upost <- ".tsv.gz"
> furl <- paste(upre, fcode, upost, sep = "")
> furl
[1] "http://epp.eurostat.ec.europa.eu/NavTree_prod/everybody/BulkDownloadListi

> temp <- tempfile()
> download.file(furl, temp)
> data <- read.table(gzfile(temp), sep = "\t", header = TRUE, row.names = 1,
+     na.strings = ": ")
> unlink(temp)
```

Show the data structure

```
> str(data)
```

```
'data.frame':        96 obs. of  12 variables:
 $ X2001: num  NA 81.7 NA 81.2 75.4 NA 83.2 81.4 78.5 81.4 ...
 $ X2002: Factor w/ 60 levels "64.4 ","65.6 ",..: NA 53 NA 49 22 NA 59 48 39 5
 $ X2003: num  NA 81.5 NA 81.1 75.9 NA 83.2 81.2 78.6 81.3 ...
 $ X2004: num  NA 82.1 NA 81.9 76.2 NA 83.8 81.8 79.1 81.9 ...
 $ X2005: num  NA 82.2 NA 81.9 76.2 NA 84 80.8 79.2 82 ...
 $ X2006: num  76 82.8 75.4 82.3 76.3 NA 84.2 82 79.9 82.4 ...
 $ X2007: Factor w/ 70 levels "61.4 ","61.8 ",..: 32 65 23 62 30 NA 69 58 54 6
 $ X2008: num  76.7 83.3 76.3 82.6 77 NA 84.6 82.9 80.5 82.7 ...
 $ X2009: num  76.7 83.2 76.3 82.8 77.4 NA 84.6 83.5 80.5 82.8 ...
 $ X2010: num  NA 83.5 76 83 77.4 NA 84.9 83.9 80.9 83 ...
 $ X2011: Factor w/ 71 levels "64.7 ","66.0 ",..: NA 63 NA 61 27 22 69 56 51 5
 $ X2012: Factor w/ 69 levels "66.1 ","66.6 ",..: NA 59 21 53 29 28 66 57 47 5
```

First few lines

```
> head(data)
```

```
           X2001 X2002 X2003 X2004 X2005 X2006  X2007 X2008 X2009
Y_LT1,F,AM    NA  <NA>    NA    NA    NA  76.0  76.8   76.7  76.7
Y_LT1,F,AT  81.7 81.7   81.5  82.1  82.2  82.8  83.1   83.3  83.2
Y_LT1,F,AZ    NA  <NA>    NA    NA    NA  75.4  75.5   76.3  76.3
Y_LT1,F,BE  81.2 81.2   81.1  81.9  81.9  82.3  82.6   82.6  82.8
Y_LT1,F,BG  75.4 75.5   75.9  76.2  76.2  76.3 76.6 b  77.0  77.4
Y_LT1,F,BY    NA  <NA>    NA    NA    NA    NA   <NA>    NA    NA
           X2010  X2011 X2012
Y_LT1,F,AM    NA   <NA>  <NA>
Y_LT1,F,AT  83.5  83.8  83.6
Y_LT1,F,AZ  76.0   <NA> 76.6
Y_LT1,F,BE  83.0 83.3 b 83.1
Y_LT1,F,BG  77.4  77.8  77.9
Y_LT1,F,BY    NA  76.9  77.8
```

4 Clean the data

Some data contain labels and can be suffixed with a space. So we have to clean the data.

```
> X <- data
> X <- as.data.frame(apply(X, 2, function(x) as.numeric(sapply(x, FUN = functi
+     "", u)))))
> dimnames(X)[[1]] <- dimnames(data)[[1]]
```

Show the structure

```
> str(X)
```

```
'data.frame':        96 obs. of  12 variables:
 $ X2001: num  NA 81.7 NA 81.2 75.4 NA 83.2 81.4 78.5 81.4 ...
 $ X2002: num  NA 81.7 NA 81.2 75.5 NA 83.2 81 78.7 81.3 ...
 $ X2003: num  NA 81.5 NA 81.1 75.9 NA 83.2 81.2 78.6 81.3 ...
 $ X2004: num  NA 82.1 NA 81.9 76.2 NA 83.8 81.8 79.1 81.9 ...
 $ X2005: num  NA 82.2 NA 81.9 76.2 NA 84 80.8 79.2 82 ...
 $ X2006: num  76 82.8 75.4 82.3 76.3 NA 84.2 82 79.9 82.4 ...
 $ X2007: num  76.8 83.1 75.5 82.6 76.6 NA 84.4 82.1 80.2 82.7 ...
 $ X2008: num  76.7 83.3 76.3 82.6 77 NA 84.6 82.9 80.5 82.7 ...
 $ X2009: num  76.7 83.2 76.3 82.8 77.4 NA 84.6 83.5 80.5 82.8 ...
 $ X2010: num  NA 83.5 76 83 77.4 NA 84.9 83.9 80.9 83 ...
 $ X2011: num  NA 83.8 NA 83.3 77.8 76.9 85 83.1 81.1 83.2 ...
 $ X2012: num  NA 83.6 76.6 83.1 77.9 77.8 84.9 83.4 81.2 83.3 ...
```

and first few lines

```
> head(X)
```

```
           X2001 X2002 X2003 X2004 X2005 X2006 X2007 X2008 X2009
Y_LT1,F,AM    NA    NA    NA    NA    NA  76.0  76.8  76.7  76.7
Y_LT1,F,AT  81.7  81.7  81.5  82.1  82.2  82.8  83.1  83.3  83.2
Y_LT1,F,AZ    NA    NA    NA    NA    NA  75.4  75.5  76.3  76.3
Y_LT1,F,BE  81.2  81.2  81.1  81.9  81.9  82.3  82.6  82.6  82.8
Y_LT1,F,BG  75.4  75.5  75.9  76.2  76.2  76.3  76.6  77.0  77.4
Y_LT1,F,BY    NA    NA    NA    NA    NA    NA    NA    NA    NA
           X2010 X2011 X2012
Y_LT1,F,AM    NA    NA    NA
Y_LT1,F,AT  83.5  83.8  83.6
Y_LT1,F,AZ  76.0    NA  76.6
Y_LT1,F,BE  83.0  83.3  83.1
Y_LT1,F,BG  77.4  77.8  77.9
Y_LT1,F,BY    NA  76.9  77.8
```

4.1 Parse meta data

Meta data are encoded in the first column, used as the row names. We can parse the data.

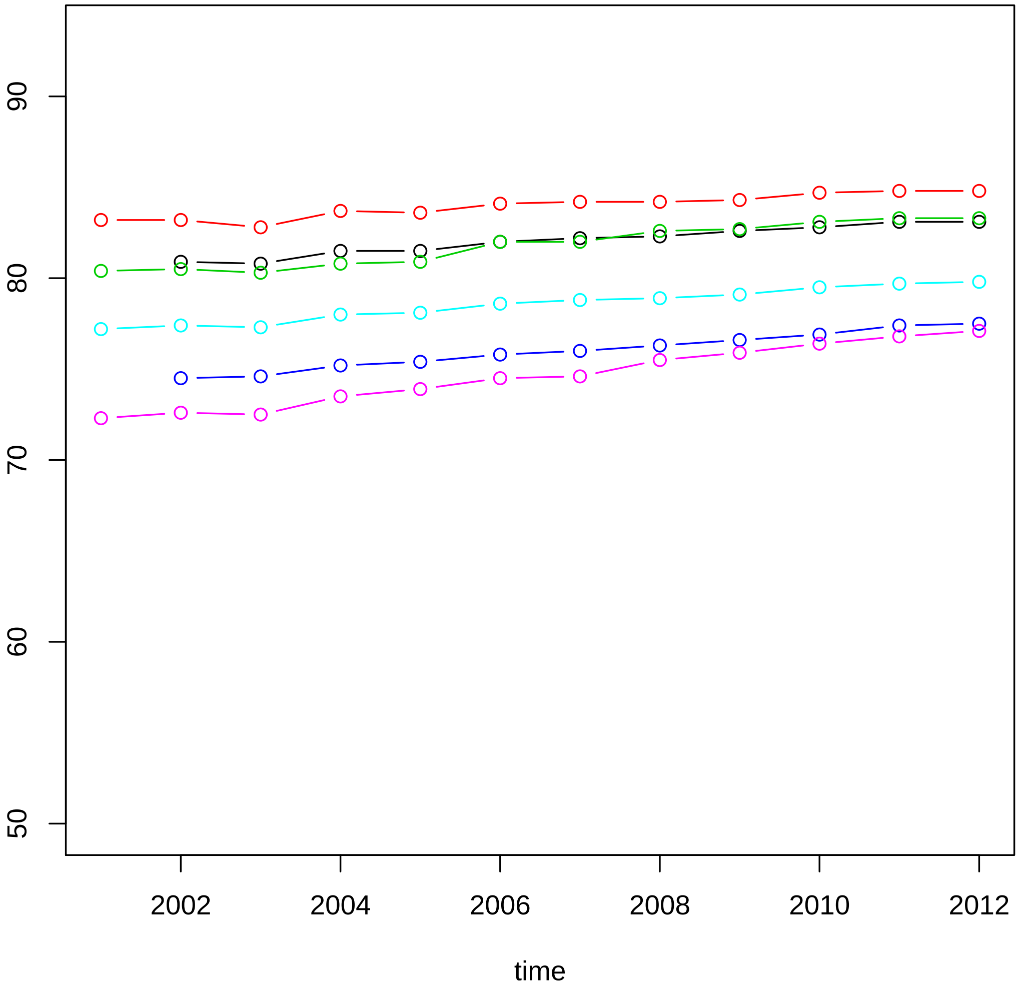
```
> meta <- sapply(dimnames(X)[[1]], strsplit, split = ",")
> meta <- matrix(unlist(meta), ncol = length(meta[[1]]), byrow = TRUE)
> dimnames(meta)[[1]] <- dimnames(X)[[1]]
> head(meta)
```

```
           [,1]    [,2] [,3]
Y_LT1,F,AM "Y_LT1" "F"  "AM"
Y_LT1,F,AT "Y_LT1" "F"  "AT"
Y_LT1,F,AZ "Y_LT1" "F"  "AZ"
Y_LT1,F,BE "Y_LT1" "F"  "BE"
Y_LT1,F,BG "Y_LT1" "F"  "BG"
Y_LT1,F,BY "Y_LT1" "F"  "BY"
```

Select specific country data

```
> countries <- c("SI", "EU28", "IT")
> filter <- meta[, 3] %in% countries
> Y <- X[filter, ]
> time <- as.numeric(gsub("X", "", dimnames(X)[[2]]))

> ylim <- c(50, max(Y, na.rm = TRUE) * 1.1)
> plot(time, time, ylim = ylim, xlim = range(time), ylab = "")
> for (i in 1:dim(Y)[1]) lines(time, Y[i, ], col = i, type = "b")
```



SessionInfo

Windows 7 x64 (build 7601) Service Pack 1

- R version 3.0.2 (2013-09-25), x86_64-w64-mingw32
- Locale: LC_COLLATE=Slovenian_Slovenia.1250, LC_CTYPE=Slovenian_Slovenia.1250, LC_MONETARY=Slovenian_Slovenia.1250, LC_NUMERIC=C, LC_TIME=Slovenian_Slovenia.1250
- Base packages: base, datasets, graphics, grDevices, grid, methods, splines, stats, utils
- Other packages: Formula 1.1-1, Hmisc 3.14-4, knitr 1.6, lattice 0.20-27, survival 2.37-7
- Loaded via a namespace (and not attached): cluster 1.14.4, evaluate 0.5.5, formatR 0.10, latticeExtra 0.6-26, RColorBrewer 1.0-5, stringr 0.6.2, tools 3.0.2

Project path: D:/_Y/R/Rome
Main file : ../doc/getEurostat.Rnw

View as vignette

Project files can be viewed by pasting this code to R console:

```
> projectName <-"Rome";  mainFile <-"getEurostat"


> commandArgs()
> library(tkWidgets)
> openPDF(file.path(dirname(getwd()),"doc",
> paste(mainFile,"PDF",sep=".")))
> viewVignette("viewVignette", projectName, #
> file.path("../doc",paste(mainFile,"Rnw",sep=".")))
> #
```