

First R-03

Paul E. Johnson¹ ²

¹Department of Political Science

²Center for Research Methods and Data Analysis, University of Kansas

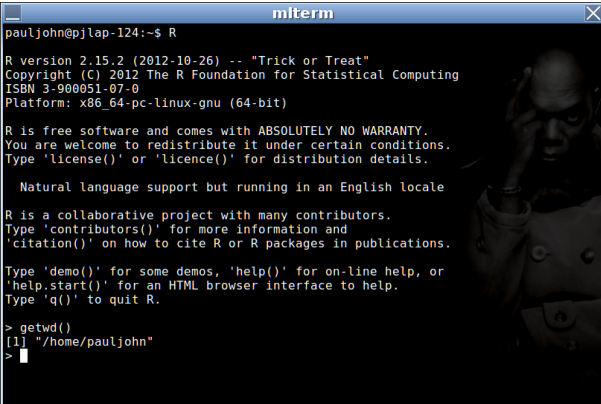
May 18, 2013

Introduction

- These notes are about the “path” and the “current working directory”
- Key concepts
 - “current working directory”
 - Relative and absolute file paths
- Key R functions that are emphasized
 - `getwd()`
 - `setwd()`

When R starts, Where are you?

- When you start R, “where are you”?
- This example: I started R in my Linux home directory, `/home/pauljohn`
- `getwd()` \iff get Working Directory (WD)!



```
pauljohn@pjlap-124:~$ R
R version 2.15.2 (2012-10-26) -- "Trick or Treat"
Copyright (C) 2012 The R Foundation for Statistical Computing
ISBN 3-900051-07-0
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> getwd()
[1] "/home/pauljohn"
>
```

When R starts, Where are you?

- Suppose instead I create a directory `/home/pauljohn/tmp/funnyDirname`
- I change into that directory, and start R

```
mlterm
R version 2.15.2 (2012-10-26) -- "Trick or Treat"
Copyright (C) 2012 The R Foundation for Statistical Computing
ISBN 3-900051-07-0
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> getwd()
[1] "/home/pauljohn/tmp/funnyDirname"
>
```

- R notices the difference, so the current WD becomes R's WD

Working Directory Means...

- Everything R reads (and writes) has to be (or go) “somewhere”
- When you read or write a file WITHOUT SPECIFYING A FULL PATH, R looks in the current working directory.
- The FULL PATH would be (long, boring, typo-prone) like
Mac: “/user/your-name/Documents/psych790/exercise-1/mydata.dat”
Windows: “C:/users/your-name-
here/Documents/psych790/exercise-1/mydata.dat”
- Less error prone to keep data and output in current working directory, or in a directory that is “relative” to it (explanation below).

Path

- Graphical Computer Interfaces are making it tougher to know “where am I?”.
- Windows launcher icons have a “Start In” argument...
- If you launch R from a menu or icon, where does it “put you?”
- The `getwd()` command displays the current working directory

```
> getwd ()
```

Windows Slashes

- Misty Water Colored Memories of MS Windows
 - Olden days: R would think your working directory was “C:\Program Files\R” or something silly like that.
 - Now, perhaps it guesses “C:\user\your-name-here\Documents” or such.
- **ALTHOUGH** Windows uses the backslash for a separator, inside R we **DO NOT** use the backslash, even on Windows. Instead, that path would be typed in as “C:/usr/your-name-here/Documents”.

Please Be organized

- Be like me. Make a directory structure
 - Macintosh:
`/user/your-name-here/documents/ps706/First-R-03`
 - Windows:
`C:\users\your-name-here\Documents\ps706\First-R-03`
 - Linux/Unix: `/home/your-name/documents/ps706/First-R-03`
- Each separate project is in a separate folder.

Relative file paths

Suppose your directories are like so

```
school
  pols706
    homework
      ex-1
      ex-2
    test
      test-1
      test-2
```

Set working directory:

```
"school/ ... /test/test-2"
```

- Ask for "mydata.dat", R looks in CWD `"./test-2"`

- Want mydata.dat stored in "test"? Use file name `"../mydata.dat"`

Notice the 2 dots

`"../"` means "go up one level"

- Want myditty.dat in "test/test-1"? Use file name `"../test-1/myditty.dat"`
- Want myex1.dat from ex-1? Ask for `"../.. /homework/ex-1 /myex1.dat"`

Be the Boss of R!

- It is OK to ask R where it is working.

```
> getwd()
```

- Better than asking, is Telling R where to work. Be assertive!
- I manage that by starting R from a terminal (or actually, an editor in that directory), and R notices where I was when I started.

Set the Working Directory

- Perhaps you don't start R from a terminal, as I do. You can still tell R where it is supposed to work.
- For that, you need to develop a little “routine”.
- But it's a GOOD routine, generally valuable in all kinds of research computing.

Option 1: `setwd("some-full-path-name")`

- This is the difficult way. The first line in any R program is `setwd()`. Almost all R novices think is the most obvious way.
- Start R, let it open up anywhere it wants, and
 - Use `setwd()` to make R change its working directory

```
> setwd("some-directory-name-here")
```
 - Use “`setwd()`”, with a full directory name IN QUOTES.
 - Windows:

```
> setwd("c:/users/your-name-here/
Documents/ps706/First-R-03")
```
 - YES, I mean forward slashes, even in Windows.
- FYI: For other Windows tips, please see the FAQ for Windows that is distributed with R documentation. (don't forget “`help.start()`”).

Method #2 (Smarter way)

- Make a directory
- Put an R file in that directory.
- Open that file in an editor (even the R editor window will do, if your OS has one)
- R will figure out where the file was and assume that is the working directory.
- Works with Emacs (and ESS), Notepad++ (with NPPTOR), or RStudio.

Relative File Locations are OK

- Suppose your default working directory is “/home/your-name/”
- You could keep a data file “mydata.dat” in the directory “/home/your-name/ps706”
- Start R, Use the `setwd()` command on that relative path:

```
> setwd( " ps706 " )
```

- If all your input is in “/home/your-name/ps706”, then it will all be available.

Another idea: relative directories above and below

- Another idea: set the working directory
“/home/your-name/ps706/exercise_1”.
 - Gives access to all files in that directory
- Can still refer to the data file in “one directory above” as

```
“ .. /mydata.dat ”
```
- If you moved the data into
“/home/your-name/ps706/exercise_1/data”, you could
 - start R and set the working directory
“/home/your-name/ps706/exercise_1”
 - then load a data file with a relative name
“ .. /data/mydata.dat”.

Create a Subdirectory for your plots

- In R you can also create subfolders and then write output in them.

```
> myPlotDir <- "outdir"  
> dir.create(myPlotDir)
```


Just one more thing before finishing

- R has separate functions to write graphs in files with various formats (pdf, postscript, png, jpeg)
- The “pull down” Save As menu seems convenient, but it is not “scriptable” or flexible.
- So learn to write commands that can save output images automatically

Recall PDF example from Frist-R-01

```
> x <- rgamma(100, 1.5, 2.2)
> y <- rnorm(100) + 0.2 * x
> pdf("RegPlot1.pdf", height=6, width=6, paper="special")
> plot(y ~ x, main="Here comes a line of best fit!")
> mod1 <- lm(y ~ x)
> abline(mod1)
> dev.off() #turns off pdf output device
```

That creates a pdf and writes it in the Current Working Directory.

You can write the files in other locations

```
> x <- rgamma(100, 1.5, 2.2)
> y <- rnorm(100) + 0.2 * x
> dir.create("plots2")
> pdf("plots2/RegPlot1.pdf", height=6, width=6, paper="
  special")
> plot(y ~ x, main = "I'm saved in plots folder!")
> mod1 <- lm(y ~ x)
> abline(mod1)
> dev.off() #turns off pdf output device
```

Creates a directory “plots2” and writes the file in there

- In addition to creating plots2 in the working directory, can place the new directory elsewhere
 - relative placement, above: “../plots2/RegPlot1.pdf”
 - absolute path:
“C:/user/long—annoying—directory/plots2/RegPlot1.pdf”