First R Experience

Paul E. Johnson

1Department of Political Science
2Center for Research Methods and Data Analysis, University of Kansas

August 12, 2015
Overview

1. Get Started
2. Interactively Test
3. Quick Look at R Graphics
4. How to ask for help
Outline

1. Get Started

2. Interactively Test

3. Quick Look at R Graphics

4. How to ask for help
Install R

R is more-or-less easily installable for all major operating systems.

- R freely downloadable from any CRAN MIRROR, such as http://rweb.crmda.ku.edu/cran
- Tips about installing R for Windows http://crmda.ku.edu/windows-admin-tips
- Later, get a GOOD PROGRAM EDITOR.
  - Mac’s R.app is OK
  - Windows editor is not adequate
  - Try Emacs or RStudio.
Run R: Linux R Terminal Looks like this

On my Linux system, the “R console”:

```
R version 3.1.2 (2014-10-31) -- "Pumpkin Helmet"
Copyright (C) 2014 The R Foundation for Statistical Computing
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.

> x <- rnorm(1000, m = 50, s = 30)
> mean(x)
[1] 49.48432
> sd(x)
[1] 28.39534
> 
```
Windows R Terminal Similar

On Fan Jia’s Windows computer:

```
R version 2.15.1 (2012-06-22) -- "Roasted Marshmallows"
Copyright (C) 2012 The R Foundation for Statistical Computing
ISBN 3-900051-07-0
Platform: x86_64-pc-mingw32/x64 (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.

> |
```
Outline

1. Get Started

2. Interactively Test

3. Quick Look at R Graphics

4. How to ask for help
Type At the > Prompt!

- The > prompt is a quick “try this out” opportunity.
- Nothing you type there is “remembered” for future session.
- Once you get past this initial experience, you need to learn to write R scripts to remember your steps.
Start the R Terminal. Try this!

```r
> x <- c(3, 5, 7, 9, 11)
> x
[1]  3  5  7  9 11

> mean(x)
[1] 7

> var(x)
[1] 10
```
Start the R Terminal

- Note that

```
> x
[1] 3 5 7 9 11
```

- Is the exact same as

```
> print(x)
[1] 3 5 7 9 11
```

- Because typing a thing’s name tells R to use the print function to display it.
Your first statistics in the R Terminal

```r
> y <- c(4, -2, 3, 5, 1)
> mean(x)

[1] 7

> var(x)

[1] 10
```
Your first plot, launched from the R Terminal

This should launch a graphic image in a separate display window

```r
> plot(y ~ x, main = "My First Plot")
```

![My First Plot](image)
Your first random samples, from the R Terminal

- Lets draw 40 observations, from two different “normal” (or Gaussian) distributions
  
  ```
  > x <- rnorm(40, m = 7, sd = 10)
  > y <- rnorm(40, m = 11, sd = 17)
  ```

- Check the first few values with head:

  ```
  > head(x)
  
  
  > head(y)
  
  ```
Tidbit about Random Numbers and Reproducability

- If you run “rnorm(40, m = 7, sd = 10)” over again, you will get a different (unpredictable) set of numbers every time.
- It is possible to re-draw the same random numbers, over and over, if you re-set the random number generator.

```r
> set.seed(66565)  # put in any integer you like
> x <- rnorm(40, m = 7, sd = 10)
> y <- rnorm(40, m = 11, sd = 17)
```

- Homework: Use the `head()` function to view `x` and `y`, then run those 3 lines over a few times.
Let's bundle them together into a "data.frame" object

```r
> dat1 <- data.frame(x, y)
> head(dat1)

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.502011</td>
<td>33.52677</td>
</tr>
<tr>
<td>22.013442</td>
<td>26.59070</td>
</tr>
<tr>
<td>-4.754001</td>
<td>-13.40854</td>
</tr>
<tr>
<td>-2.395734</td>
<td>-23.85941</td>
</tr>
<tr>
<td>9.430454</td>
<td>16.74289</td>
</tr>
<tr>
<td>11.431225</td>
<td>24.25179</td>
</tr>
</tbody>
</table>
```
Outline

1. Get Started
2. Interactively Test
3. Quick Look at R Graphics
4. How to ask for help
A histogram is a basic type of R Graphic

```r
> hist(dat1$x, prob = TRUE, main = "My First Histogram")
```

My First Histogram

---

The code above creates a histogram of the data in `dat1$x`. The `hist` function in R is used to generate the histogram, with `prob = TRUE` to ensure the histogram is a density histogram rather than a count histogram. The `main` argument is used to title the histogram as "My First Histogram".
Want more detail on histograms?

- Read the R documentation
  
  ```
  > ?hist
  ```

- Run their examples
  
  ```
  > example(hist)
  ```
Plot the y against x variable

> plot(y ~ x, data = dat1)
Start the R Terminal

```r
> plot(y ~ x, main = "Here comes a line!")
> mod1 <- lm(y ~ x, dat = dat1)
> abline(mod1)
```

Here comes a line!
Outline

1. Get Started
2. Interactively Test
3. Quick Look at R Graphics
4. How to ask for help
When you ask for help

- We get emails asking for help and people don’t tell us either
  1. What they did or
  2. Where they are.

- We waste time writing back asking
  1. What did you do?
  2. What kind of computer do you have?
  3. What were the error messages, exactly?
  4. What were you trying to do?
1. Give sessionInfo()

If you are writing to ask anybody for help about R, they need these bits of information.

1. ALWAYS provide the output of sessionInfo().

```r
> sessionInfo()
```

```
R version 3.1.2 (2014-10-31)
Platform: x86_64-pc-linux-gnu (64-bit)

locale:
 [1] LC_CTYPE=en_US.UTF-8    LC_NUMERIC=C
      LC_TIME=en_US.UTF-8
      LC_COLLATE=en_US.UTF-8  LC_MONETARY=
      en_US.UTF-8
```
1. Give sessionInfo() ...

[6] LC_MESSAGES=en_US.UTF-8  LC_PAPER=en_US.UTF-8  LC_NAME=C  LC_ADDRESS=C  LC_TELEPHONE=C

attached base packages:
[1] stats  graphics  grDevices  utils  datasets  base

loaded via a namespace (and not attached):
[1] tools_3.1.2
2. Provide a Runable Example

2. Provide a small, single-purpose example script that shows the problem you are trying to solve.

- It is tempting to just copy 100s of lines of disorganized code.
- Produce a small, clear example of the problem you are trying to solve.

3. Provide the error output from R when you have trouble.