A (very) short introduction to R

Wei(Becky) Lin

July 8, 2013

1 Introduction

R is a powerful language and environment for statistical computing and graphics. It is a public domain ("GNU") project which is similar to the commercial S language and enviorment which was developed at Bell Laboratories(Formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. R is much used as an educational language and research tool.

The main advantages of R are the facts that R is free and there is a lot of help available. It is quite similar to other programming languages such as MatLab(not free), but more user-friendly than language such as C++ or Fortan.

2 Let the games begin

2.1 Install R

To install R on your computer(legally for free!), go to the home website of R:

http://www.r-project.org

and do the following (assuming you work on a windows computer):

- click download "CRAN" in the left bar
- choose a download site, recommend you try the link for university of Toronto
- choose "Download R for Windows" as target (if you use different operation system, choose your one)
- click "base"
- choose "Download R 3.0.1 for Windows" and choose default answers for all questions.
- click the downloaded file and install it step by step.

After finishing this setup, you should see and "R" icon on your desktop. Clicking this would start up the standard interface.

2.2 Getting Started With R

Click on the R icon, R is started, there is a console waiting for input. At the prompt(>), you can enter numbers and perform calculations. such as

```
>2+3
[1] 5
```

It is better to assign values to variables with the assignment operator "=". Typing the variable by itself at the prompt will print out the value. Not that another form of assignment operator "< −" is also in use

```
> x=2*3 # same as x <- 2*3
> x
[1] 6
```

At anytime, if you want to clean the screen please press Ctrl+L.

How to quit from R?

To quit from R, enter "q()" after the prompt sign. i.e

```
>q()
```

How to get help in R?

There is a embedded function "rpois()" to generate poisson variables, to get the help on how to use it, you could type the following command after the prompt sign

```
>help(rpois) # or you could use ?rpois() instead
```

All text after the pound sign "#" within the same line is considered a comment. The R won't run anything that you put as comments. For R help in html format, you could go to Help on the R toolbar and select R language(html), a html version of help will be available.

How to use functions in R?

R functions are invoked by its name, then followed by the parenthesis, and zero or more arguments, for examples, c() and rpois(). The following apply the function c to combine four numeric values into a vector and assigns it to grade variable. Also, how to access the component of grade and find the sum, mean and variance are presented.

```
> grade=c(91,90,77,24) # assign 4 marks to the variable grade
> grade # display the components of grade
[1] 91 90 77 24
>  grade[3] # what is the third component in grade?
[1] 77
> sum(grade) # summation of all marks
[1] 282
> mean(grade) # find the mean of the mark
[1] 70.5
```

```
> var(grade)   # find the variance of the mark
[1] 1001.667
>
> # manually calculation to show show var() gives the sample variance
>sum( (grade-mean(grade))^2 ) /(4-1)
[1] 1001.667
```

How to write a script, save it and run it in R?

To write a script, you could click the new script under the file menu, which will open a new window waiting for input. An R script is simply a text file containing (almost) the same commands that you would enter on the command line of R. Once you finish the input of the command lines you then save it a .R or .r file. For example, I save above commands as a Rtut.R

```
grade=c(91,90,77,24) # assign 4 marks to the variable grade
grade # display the components of grade
grade[3] # what is the third component in grade?


sum(grade) # summation of all marks
mean(grade) # find the mean of the mark
var(grade)   # find the variance of the mark
# manually calculation to show show var() gives the sample variance
sum( (grade-mean(grade))^2 ) /(4-1)
```

To run a R script file, there has two ways to do it. You could open the script file and copy the command line and paste at the prompt. Or if could use function source() to run it. For the second method, you need to give a path to where the R script stored. eg.

```
>source("D:\\Rtut.R")
```

More on how to use R?

The following links can also be very useful: (1).

`http://cran.r-project.org/doc/manuals/R-intro.pdf`

A full manual. (2).

`http://cran.r-project.org/doc/contrib/`

Short-refcard.pdf A short reference card. (3).

`http://zoonek2.free.fr/UNIX/48 R/all.html`

A very rich source of examples. (4).

`http://rwiki.sciviews.org/doku.php`

A typical user wiki. (5).

`http://www.statmethods.net/`

Also called Quick-R. Gives very productive direct help. Also for users coming from other programming languages.

2.3 Poisson Example

Here is the Poisson simulation example with comments. You could use any editor to open the file and copy all the lines then paste it at the prompt in R to run it. Or, you could copy line by line, and run one line at the time to see more detail how each line works.

```
# IMPORTANT: If you want to reproduce your results,
# you will have to set the seed of the random number generator.
# There is an R function set.seed()
# TO DO: rum the set.seed(1) in the following code and see what happens
#        to the result in calculating the mean and var for L1 and L2

set.seed(1)

# Claim two vector with names L1 and L2
L1=L2=NULL


#-----------------------Simulation 1---------------------------------
# loop with 1000 iterations
for(i in 1:1000)
{
x=rpois(20,5)  # generates n=20 from poisson with lambda=5 and
               #assigns it to x. Try: > ?rpois()
L1[i]=mean(x)  # assigns mean of x to L1 for each iteration
L2[i]=mean((x-mean(x))^2) # assigns sum(x[i]-mean(x))/n (MSE)  to L2
}
mean(L1) # mean of the vector with length 1000, L1
var(L1)  # variance of the vector with length 1000 ,L1
mean(L2)
var(L2)

#-----------------------Simulation 2---------------------------------
L1=L2=NULL
for(i in 1:1000)
{x=rpois(20,10)  #simulated 20 data points from poisson with lambda=10
L1[i]=mean(x)
L2[i]=mean((x-mean(x))^2)
}
mean(L1)
var(L1)
mean(L2)
var(L2)


#-----------------------Simulation 3---------------------------------
L1=L2=NULL
for(i in 1:1000)
{x=rpois(100,5)
L1[i]=mean(x)
```

```
L2[i]=mean((x-mean(x))^2)
}
mean(L1)
var(L1)
mean(L2)
var(L2)


#------------------------Simulation 4---------------------------------
L1=L2=NULL
for(i in 1:1000)
{x=rpois(100,10)
L1[i]=mean(x)
L2[i]=mean((x-mean(x))^2)
}
mean(L1)
var(L1)
mean(L2)
var(L2)
```