Statistical Programming in R

CS50 for MBAs

Really Rewarding Regressions in R

Anita Xu
ranfeixu@college.harvard.edu
4/21/16

What is R?

- Programming language developed by two professors at the University of Auckland
- Data analysis software for data scientists, statisticians for predictive modeling and statistical computing
- Has many packages for importing and manipulating data, statistical modeling, machine learning, visualization
- Popularity has increased substantially in recent years
- Object-oriented language, weakly typed, and syntax similar to C

Why use R? (vs. Excel)

- Large data sets
- Graphing or data visualization
- Statistical analysis
- Explore the data

Let’s get started!

Fundamentals

- Assign variables with <-
- No type declarations
- Indexing from 1 (rather than 0)
- Commenting #
- Use colon to denote ranges (1:4)
- Mod, Multiplication, Integer division with %%, *, and x/y

R Objects: Scalars

- Basic calculations

```
> 3+7
[1] 10
> pi*2
[1] 6.283185
> 2^5
[1] 32
> sqrt(25)
[1] 5
```

- Variables

```
> x <- 5
> y <- -3
> z <- x + y
> z
[1] 2
```

R Objects: Vectors

- Vectors
  - Constructed in form c(x1,x2,x3..)
  - Cannot be mixed type
  - Summary(x) gives mean, median, min, max
  - Can be integers, strings, booleans, floats

```
> z <- c(3,5,7,9)
> x <- c(3,5)
> y <- c(7,9)
> z <- c(x,y)
> z
[1] 3 5 7 9
> x+y
[1] 10 14
> x*y #pointwise arithmetic
[1] 21 45
> a <- c(2,5,7)
> a+3 #mixed types
[1]  5  8 10
> b <- 10:30
> summary(b)
Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  10   15      20      20      25      30
```

R Objects: Vectors

- Sequences

```
> x <- 2:10
> x
[1]  2  3  4  5  6  7  8  9 10
> seq(2,10,by=2)
[1]  2  4  6  8 10
> seq(2,10,length=4)
[1]  2.000000  4.666667  7.333333 10.000000
> mean(x)
> var(x)
```

R Objects: Matrices

- Initialized with matrix(data, nrow = rows, ncol = columns)
- Data is a vector. Fills matrix first up to down, then left ot right
- Matrix multiplication: x %*% y
- Find eigenvalue: eigen(x)

R Objects: Matrices

```
> a <- matrix(1:6, nrow=2, ncol = 3)
> a
     [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
> b <- matrix(2:7, nrow = 3, ncol = 2)
> b
     [,1] [,2]
[1,]    2    5
[2,]    3    6
[3,]    4    7
> a  %*% b
     [,1] [,2]
[1,]   31   58
[2,]   40   76
```

Data Frames

- Type of list where each value is a vector of the same length
- Represents data table
- Can access columns with hello["column name"] or name[column index]
- Access rows with row index, followed by comma: hello[row index,]
- Print header and first few rows: head(hello)
- Print row and column names: rownames(hello), colnames(hello)

Data Frames: cont.

```
> numbers = c(7,8,9)
> strings = c("aa","bb","cc")
> bools = c(TRUE, FALSE,TRUE)
> df = data.frame(numbers,strings,bools)
> df
  numbers strings bools
1       7      aa  TRUE
2       8      bb FALSE
3       9      cc  TRUE
```

Functions

- Name <- function(arg1 [=default] ,…, argn[=default]) {…}
- Call the function as Name([arg1=]value_1, … [argn=]value_n)

Hello <- Function ( )

{

return

}

Functions

Example:

```
> mult <- function(x,y)
{return (x+y/2)}

> mult(10,12)

[1] 16

> mult(x=10,y=12)

[1] 16
```

```
> mult <- function(x,y=4)
{return (x+y/2)} #set default
y

> mult(5)

[1] 7

> mult(5,10)

[1] 10
```

Data Import and Export

- Read data into data frames: read.csv() [CSV files], read.xls() [Excel], read.table() [text files .txt]
- Files need to be in R's working directory. Use getwd() command to get directory path, use setwd() to set path to directory

```
> fname = file.choose()

> fname

[1] "/Users/Downloads/bruins2015.csv"

> data = read.csv(fname,header=T)
```

Statistical Computations

- Suppose X~ Normal(0,1)
  - Simulate 5 random observations from X:

    ```
    > rnorm(n=5,mean=0,sd=1)
    [1]  0.01111921 -0.72778332 -0.30454563 -1.00801027
    0.98772552
    ```

  - "d" returns height of probability density function (PDF), "p" returns the cumulative density function (CDF), "q" returns quantiles of the inverse CDF, "r" returns randomly generated numbers from the distribution

    ```
    > pnorm(q=0,mean=0,sd=1)
    [1] 0.5
    ```

Statistical Computations

- Similar with other distributions: Poisson, Gamma, Uniform, Binomial
  - qpois
  - dgamma
  - runif
  - pbinom

Really Rewarding Regressions with R

- Multi-linear Regressions!
- Syntax: fit<- lm(y~ x1+x2+…+xn , data = mydata)
- Summary(fit) #show results
- y is the dependent variable, x1,…,xn are the independent variables; these are either vectors or are column headers of the data frame mydata
- More complicated regressions: fit <- lm(y^2*5 ~ log(x1) + x2^3, data= mydata)

Plotting

- General: plot(x,y)
- Plot takes two vectors of same length
- Other optional arguments: xlab,ylab (axis titles), main (plot title), colors, etc.
- Can add fit best-fit lines
- Histograms with hist(x)
- Boxplot with boxplot(y)

Demo Time!

Additional Resources

- R Documentation Manual
- Text: “An Introduction to R”- W.N. Venables
- Website: “Advanced R” by Hadley Wickham
  - http://adv-r.had.co.nz/

Thanks!

http://images.clipartpanda.com/smiley-face-clip-art-thumbs-up-clipart-two-thumbs-up-happy-smiley-emoticon-512x512-eec6.png