STAT 408
Applied Regression Analysis

Miles Xi

Department of Mathematics and Statistics

Loyola University Chicago

Fall 2022

Introduction to R Programming Language: Part 1

R Programming Language

- R is a programming language for statistical computing, machine learning, and data science

- R's official website https://www.r-project.org/
  - R is free
  - Numerous packages for cutting-edge statistical and machine learning techniques
  - Very good at data visualization
  - Easy to learn

R Programming Language

- Let’s install R in your computer
- Install the correct version based on your operating system
- R works on Windows, Mac, and Linux
- R doesn’t work on Chrome book, iOS, or Android device

RStudio

- RStudio is an integrated development environment (IDE) for R

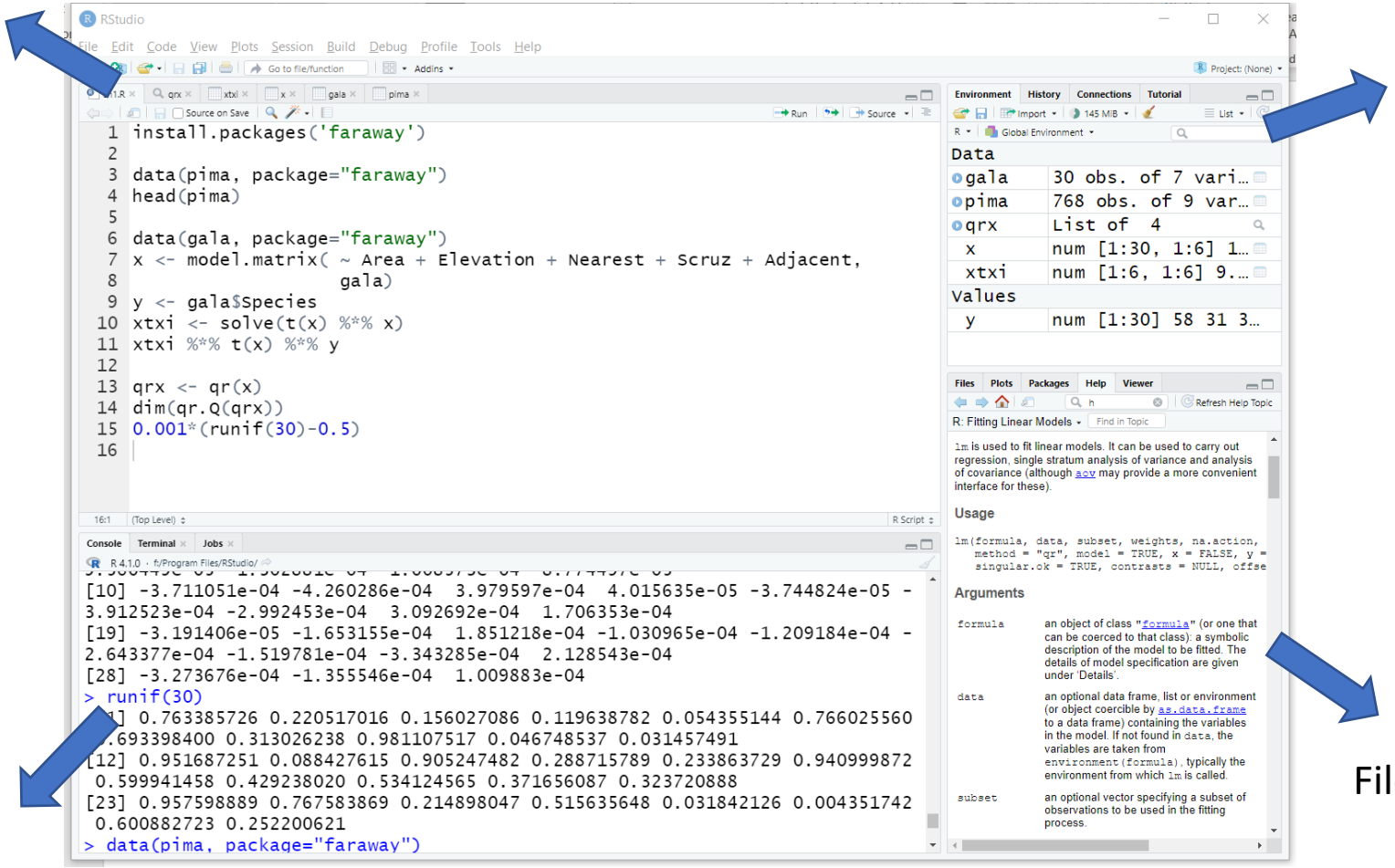
- RStudio's website https://www.rstudio.com/
  - RStudio is free
  - All R coding will be done through RStudio

- R and RStudio will be the major programming tools for this class

- Let's install R in your computer

The Interface of RStudio

Source



Environment

Files/Plots/Packages/Help

Console

R programming language

- Any programming language has two essential elements
  - Data type
  - Operation
- Major data types in R
  - Single value: numerical, character, factor, …
  - Vector
  - Matrix and array
  - List
  - Data frame
- Major operations in R
  - Numerical operation: +, -, *, /
  - Function: c(), cbind(), lm(), …
  - Logic control: if… else, for loop, …

Basic Syntax

- To output text in R, use single or double quotes

  “Hello World!”

- To output numbers, just type the number

  10

- Add numbers together

  5 + 5

- print() function explicitly print the contents to console

  print("Hello World!")

Basic Syntax

- Two ways to run code in RStudio
    1. The run button
    2. Ctrl + Enter

- Run single line: the current cursor position

- Run multiple line: select lines first and run

Comment

- Comment is to explain R code, and to make it more readable

- Comments starts with a #

- When executing the R-code, R will ignore anything that starts with #

```
# This is a comment
“Hello World!”
"Hello World!" # This is a comment
```

Variable

- Variables are containers for storing data values

- To assign a value to a variable, use the <- sign (“Alt” + “-”)

```
name <- "John"
age <- 40
```

- To print the variable value, just type the variable name

```
name         # output "John"
print(age)  # output 40
```

Data Type

- R includes the following basic data types

1. numeric: 10.5, 55, 787
2. Integer: 1L, 55L, 100L ("L" declares this as an integer)
3. Character (string): "k", "R is exciting", "FALSE", "11.5"
4. Logical: TRUE (T), FALSE (F)
5. complex: 9 + 3i ("i" is the imaginary part)

Data Type

```
# numeric
x <- 10.5

# integer
x <- 1000L

# character/string
x <- "R is exciting"

# logical
x <- TRUE
```

- class() function is to check the data type of one variable

Data Type Conversion

- Convert from one type to another with the following functions
  - as.numeric()
  - as.integer()

```
x <- 1L       # integer
y <- 2        # numeric

# convert from integer to numeric:
a <- as.numeric(x)

# convert from numeric to integer:
b <- as.integer(y)

# print the class name of a and b
class(a)
class(b)
```

Math Operation

- Basic operation: +, - , *, /, ^

```
10 + 5
10 – 5
10 * 5
10 / 5
10 ^ 5        # power
```

- Math Functions

```
max(5, 10, 15)      # maximum
min(5, 10, 15)      # minimum
sqrt(16)            # square root
abs(-4.7)           # absolute value
ceiling(1.4)        # round up to the nearest integer
floor(1.4)          # round down to the nearest integer
```

Logical Value

- Logical value is the result of comparing two values

```
10 > 9       # TRUE because 10 is greater than 9
10 == 9      # FALSE because 10 is not equal to 9
10 < 9       # FALSE because 10 is greater than 9
10 != 9      # TRUE because 10 is not equal to 9
```

- Compare two variables

```
a <- 10
b <- 9
a > b
```

If… else…

- If… else… makes a decision about which command to run, based on a logical value

```
a <- 200
b <- 33
if (b > a) {
  print("b is greater than a")
} else {
  print("b is not greater than a")
}
```

- “else” part is optional

- The pair of curly brackets { } define the scopes in the code

And and Or

- The “&” symbol means “and”

```
a <- 200
b <- 33
c <- 500
if (a > b & c > a){
  print("Both conditions are true")
}
```

- The “|” symbol means “or”

```
if (a > b | a > c){
  print("At least one of the conditions is true")
}
```

While Loop

- While loop executes a set of statements as long as a condition is TRUE

```
i <- 1
while (i < 6) {
  print(i)
  i <- i + 1
}
```

For Loop

- For loop iterates over a sequence and execute statements

```
# x iterates from 1 to 10
for (x in 1:10) {
  print(x)
}
```

Function

- A function is a block of code with a name
  - We need to “call” the function to let it run
  - We pass parameters into a function
  - A function can return data as a result

```
# create a function with the name my_function
my_function <- function() {
  print("Hello World!")
}
my_function()       # call the function named my_function
```

Function

- A function with parameters

```
my_function <- function(fname) {
  print(fname)        # paste function combine two strings
}

my_function("Peter")
my_function("Lois")
my_function("Stewie")
```