TBA2102

Introduction to R

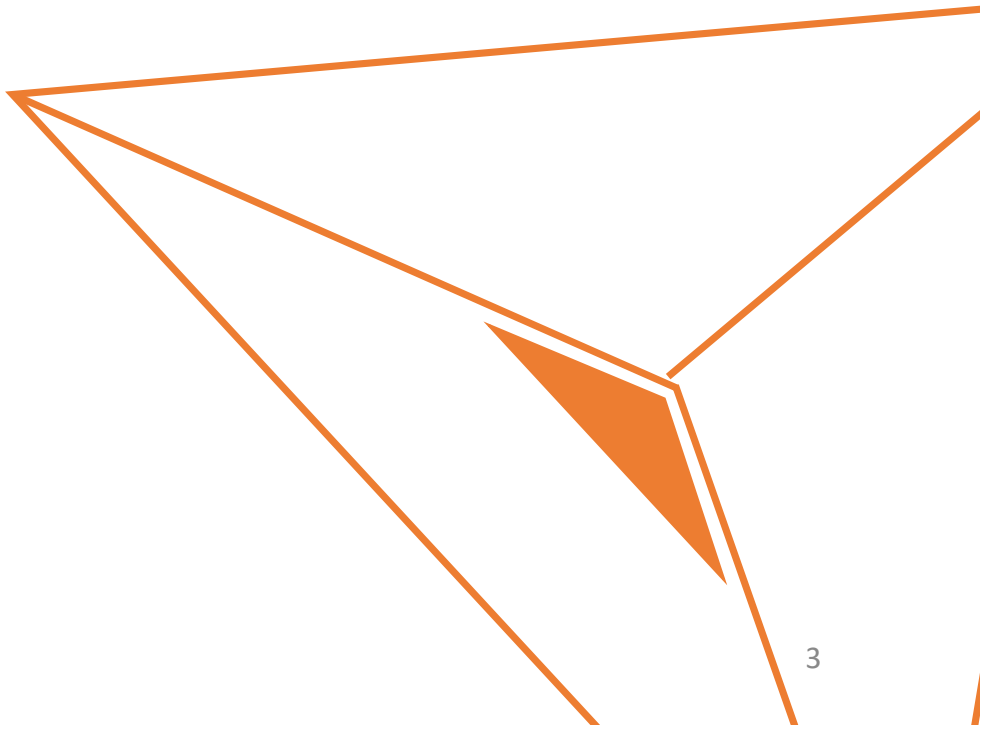
1.0 INTRODUCTION

Outline

- R Basics
- Calculations, Expressions, Variables
- Data Structure
- Read/write data
- Graphics

Part I

R Basics



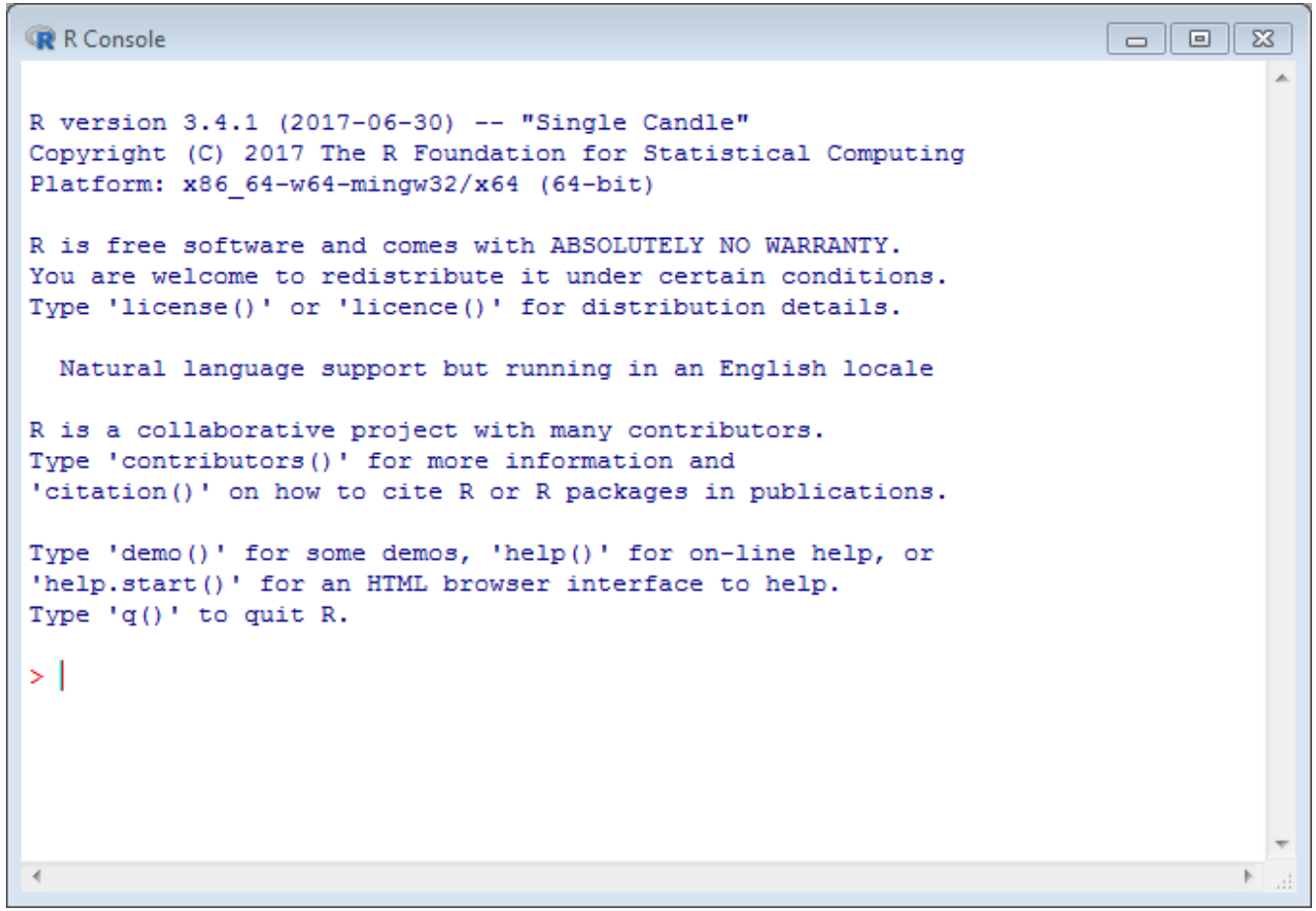
PART I

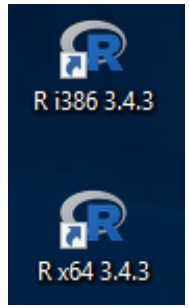
1.1 What is R?

- **What is R?**
  - A language and environment for data analysis and graphics
- **Benefits of R**
  - Provides all the capabilities of a programming language.
  - Flexible, easy, and friendly graphical capabilities that can be displayed on the video display of your computer or stored in different file formats.
  - Supports getting data from a wide variety of sources, including text files, database management systems, web XML files, and other repositories.
  - Data storage facility to store large amounts of data effectively in the memory for data analysis.
  - Free!
  - Large number of free packages available for data analysis.
  - Supported by vibrant user community

1.2 PART I

R Console



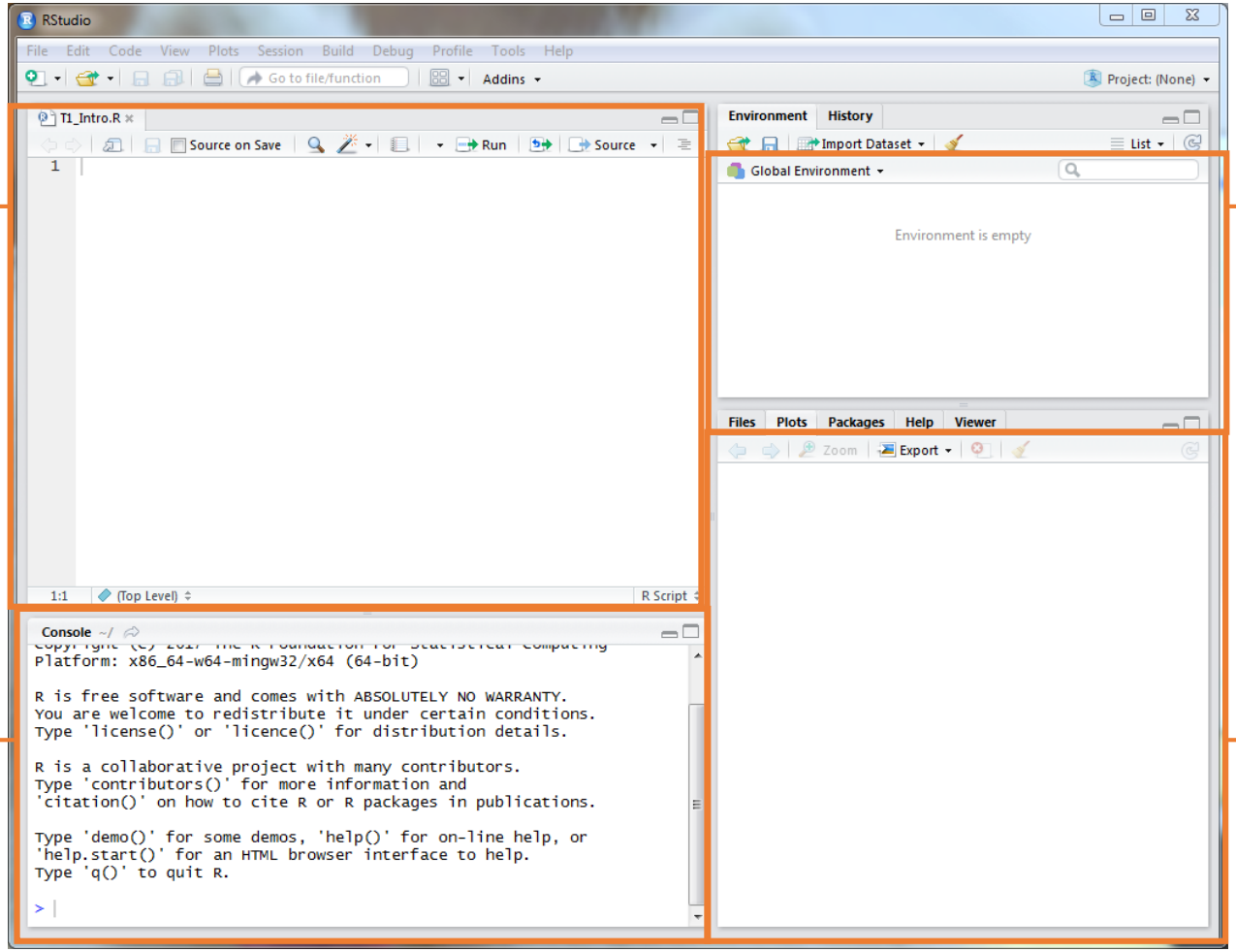


Read Page 21-22 of ***Business Analytics Using R – A Practical Approach***

PART I

1.3 RStudio

- A free integrated development environment (IDE) for R
- Have some features that makes it easier to work with
- *Note*: R still needs to be installed before RStudio



- Working Directory
  - The folder on your computer in which you are currently working.
  - R will read and write files from/to this folder.

```
> setwd("~/Documents/BT1101")  # set working directory in Mac
> setwd("D:/BT1101")    # set working directory in Windows
> getwd()               # get current directory
```

  - Use RStudio dropdown menu to select working directory

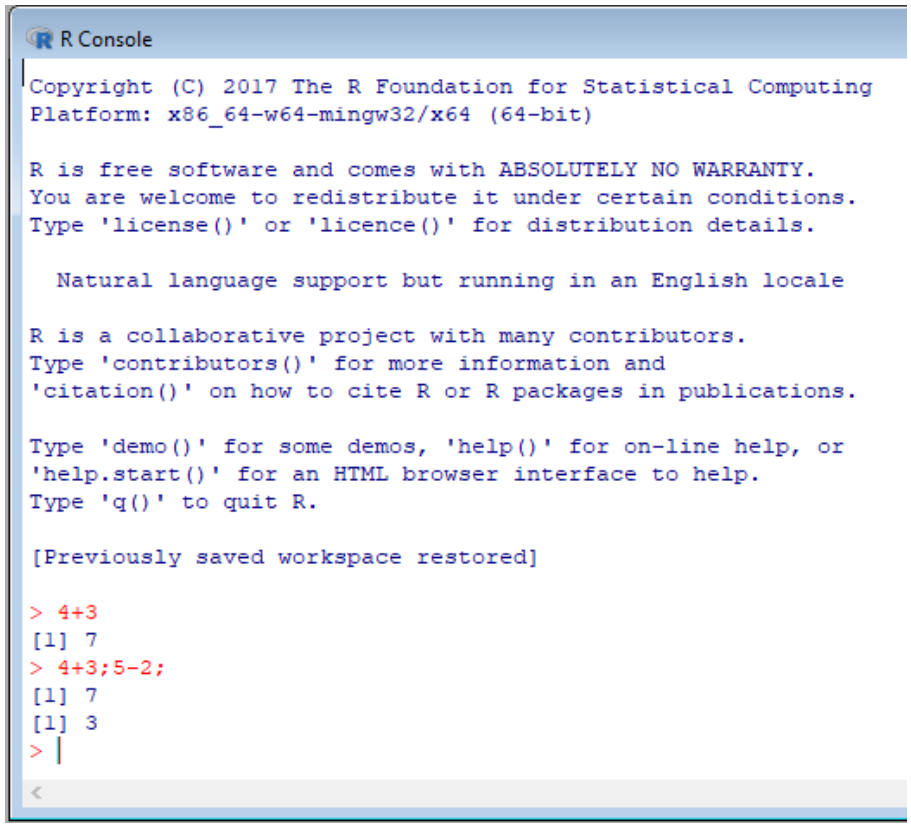
```
> Session → Set Working Directory → Choose Directory
> Files Pane → Navigating to a Directory → Clicking “More”
  → “Set as Working Directory”
```

1.5 PART I
RStudio Tutorial

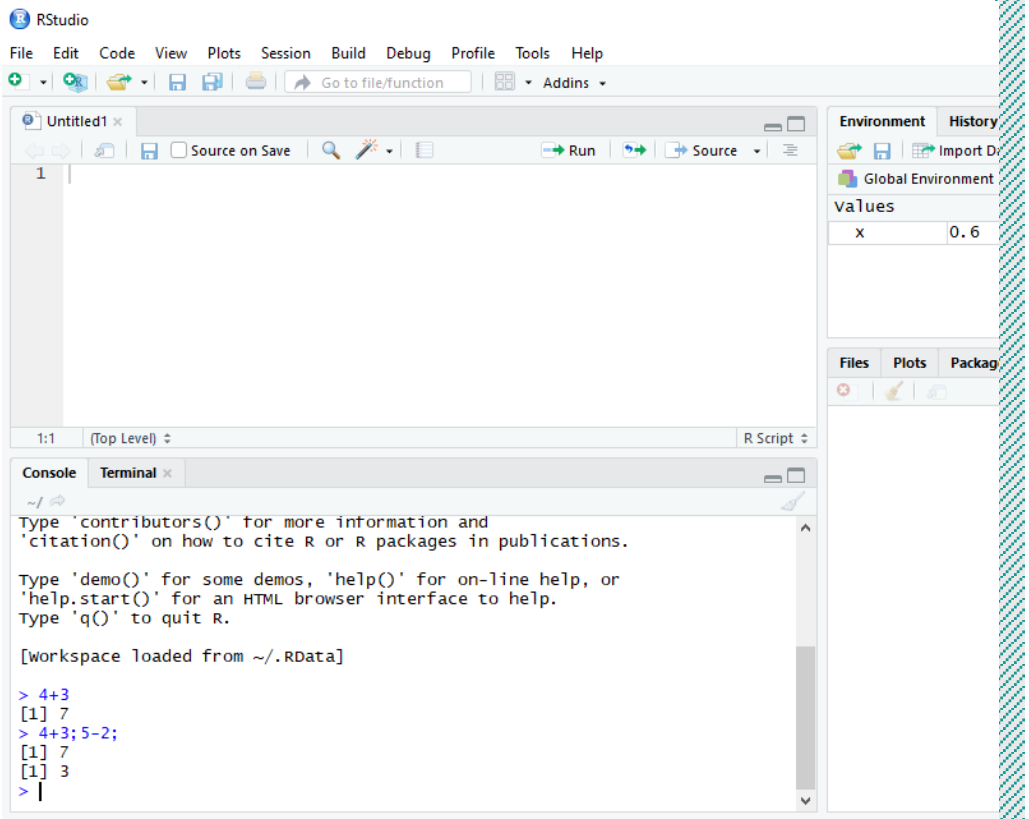
- Go through Datacamp tutorial on “Working with the Rstudio IDE”
  - https://www.datacamp.com/courses/working-with-the-rstudio-ide-part-1
  - Watch the videos in Chapter 1 Orientation:
    - > Introduction
    - > Install R and RStudio
    - > RStudio’s panes
    - > The source pane
    - > The View() function
    - > The environment pane
    - > The history pane
    - > The files pane
    - > The plots pane and the packages tab
    - > The help pane
    - > The viewer pane
    - > Wrap-up

Read Page 23-25 of ***Business Analytics Using R – A Practical Approach***

- Commands are separated either by a semi-colon (‘;’), or by a newline.



R Console

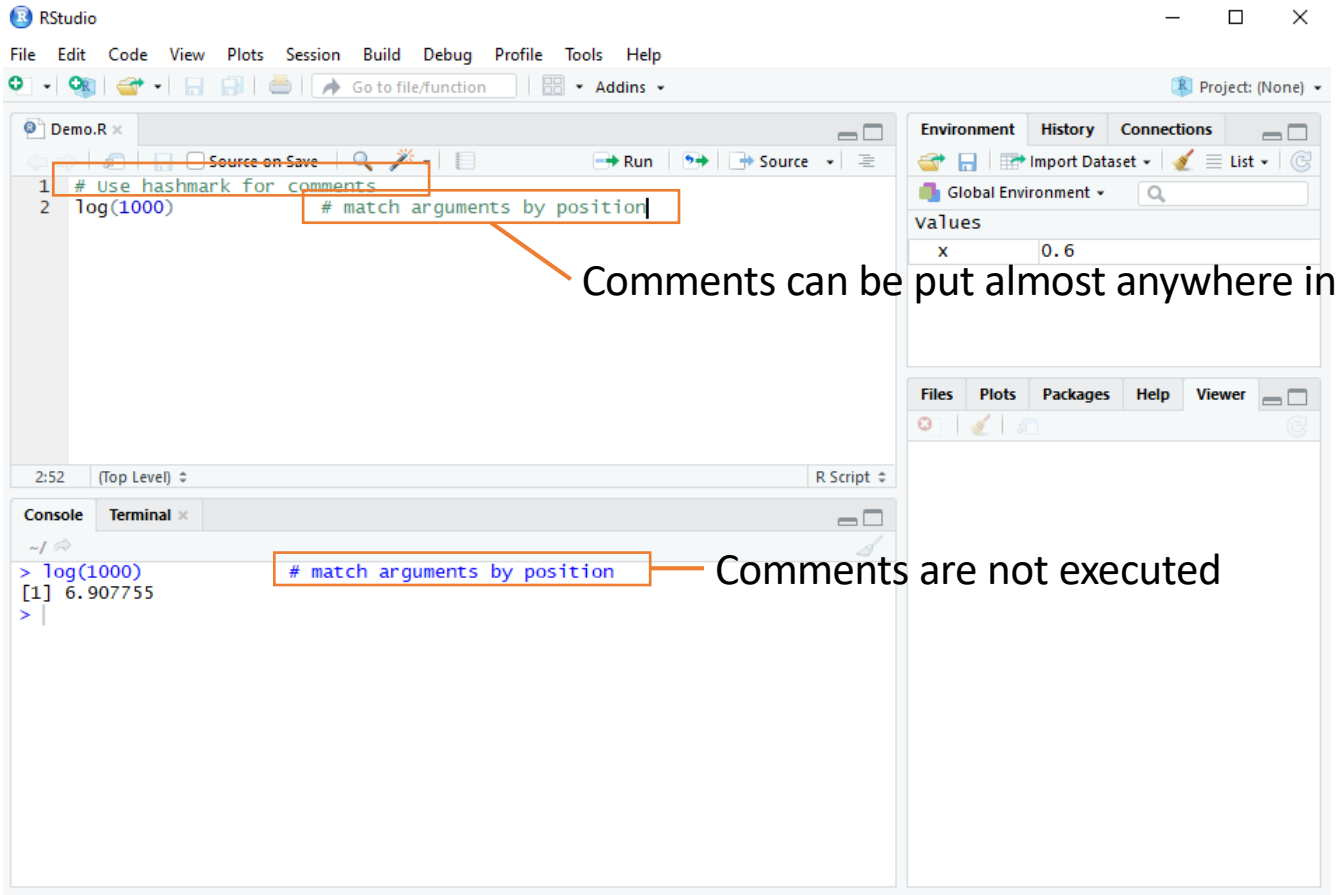


RStudio

1.7 PART I

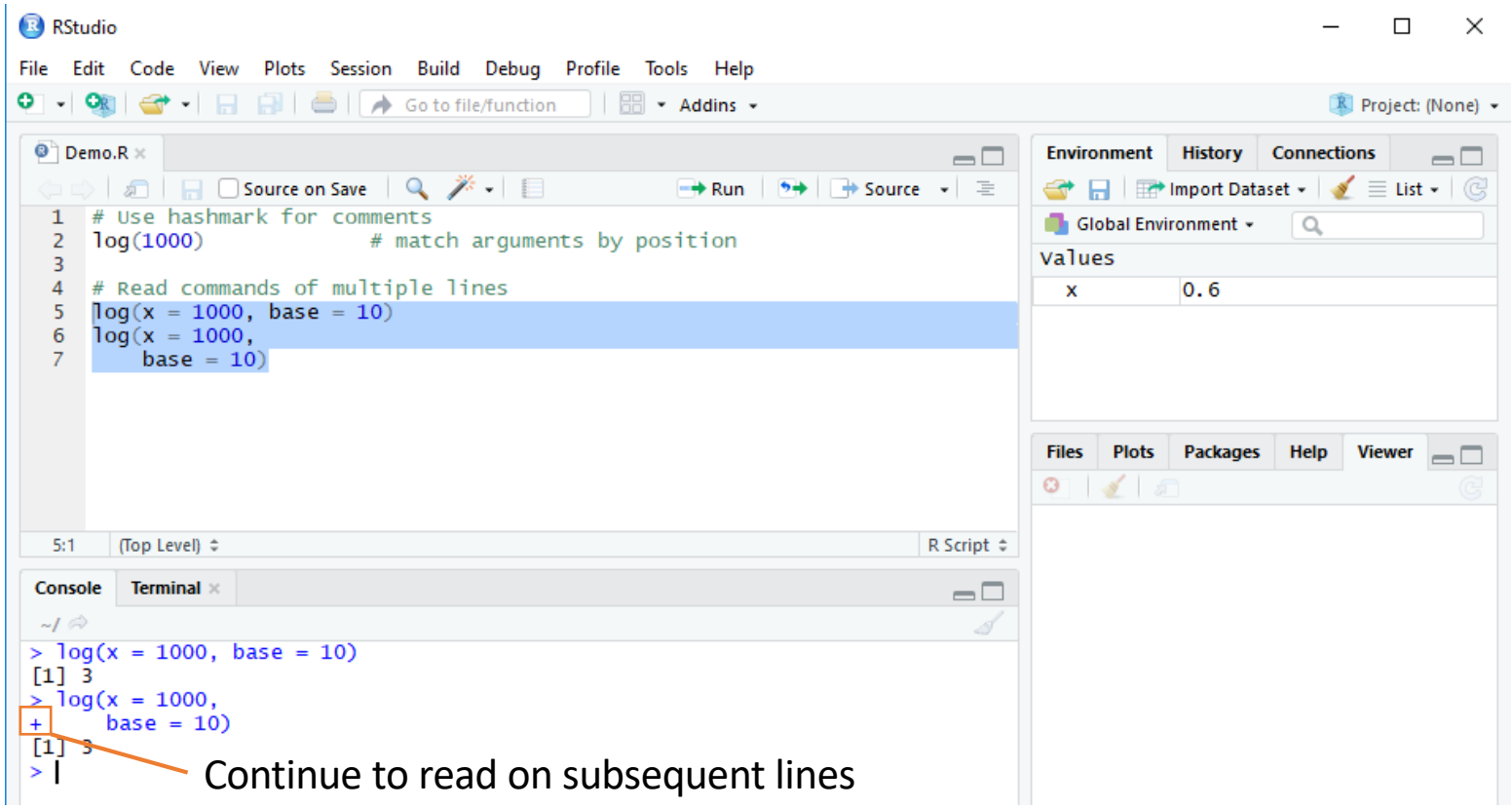
R Commands Cont.

- Comments can be put almost anywhere, starting with a hashmark (‘#’), everything to the end of the line is a comment.



1.8 PART I
R Commands Cont.

- If a command is not complete at the end of a line, R will give a different prompt (+) on subsequent lines, and continue to read input until the command is syntactically complete.



1.9 Functions

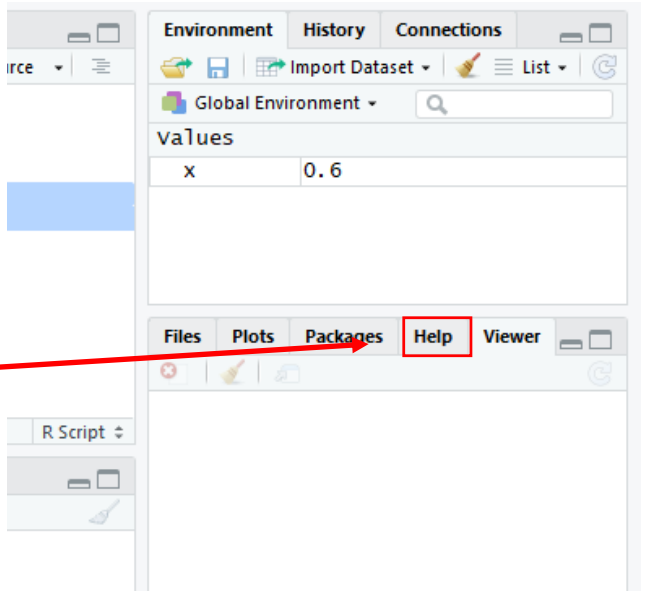
- R allows function calls similar to other languages
- Example: `log(x, base)`

```
> log(1000)              # match arguments by position
[1] 6.907755
```

- Few ways to supply function arguments

```
> log(x=1000, base=10) # match arguments by name
[1] 3
> log(base=10, x=1000) # match arguments by name
[1] 3
> log(1000, base=10)   # match by both position and
  name
[1] 3
```

1.10 PART I
Getting Help



- R has an inbuilt help facility
  - Click on Rstudio help tab

    > Review the video: https://campus.datacamp.com/courses/working-with-the-rstudio-ide-part-1/orientation?ex=27
  - To get more information on the function `log`, the command is:

    ```
    > help(log)
    ```
  - An alternative is

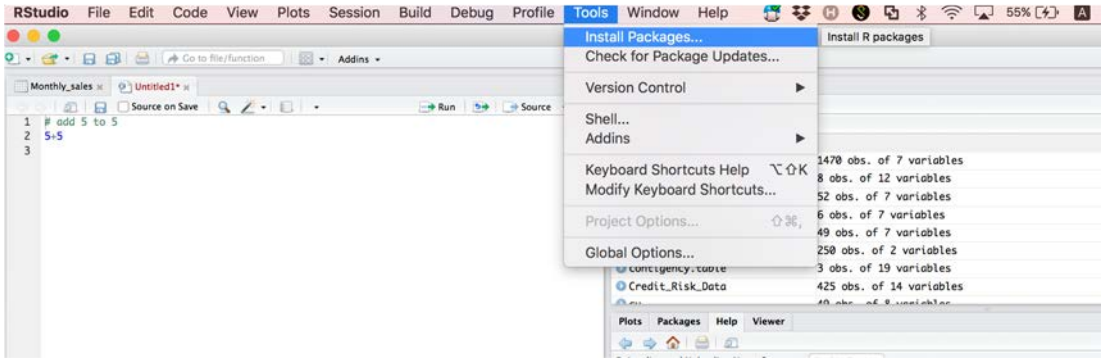
    ```
    > ?log
    ```
  - For a feature specified by special characters, the argument must be enclosed in double or single quotes:

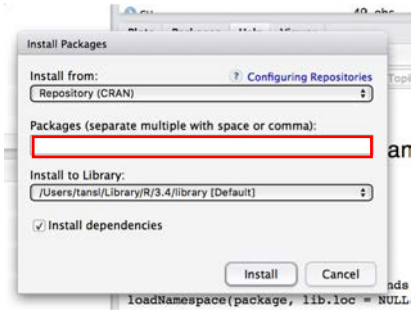
    ```
    > help("[[")
    ```

1.11 Package & Library

- Package
  - A collection of R functions, data, and compiled code in a well-defined format
  - Install package

    - Dropdown menu- install package:





    - > install.packages("dplyr")
  - Once installed, they have to be loaded into the session to be used.

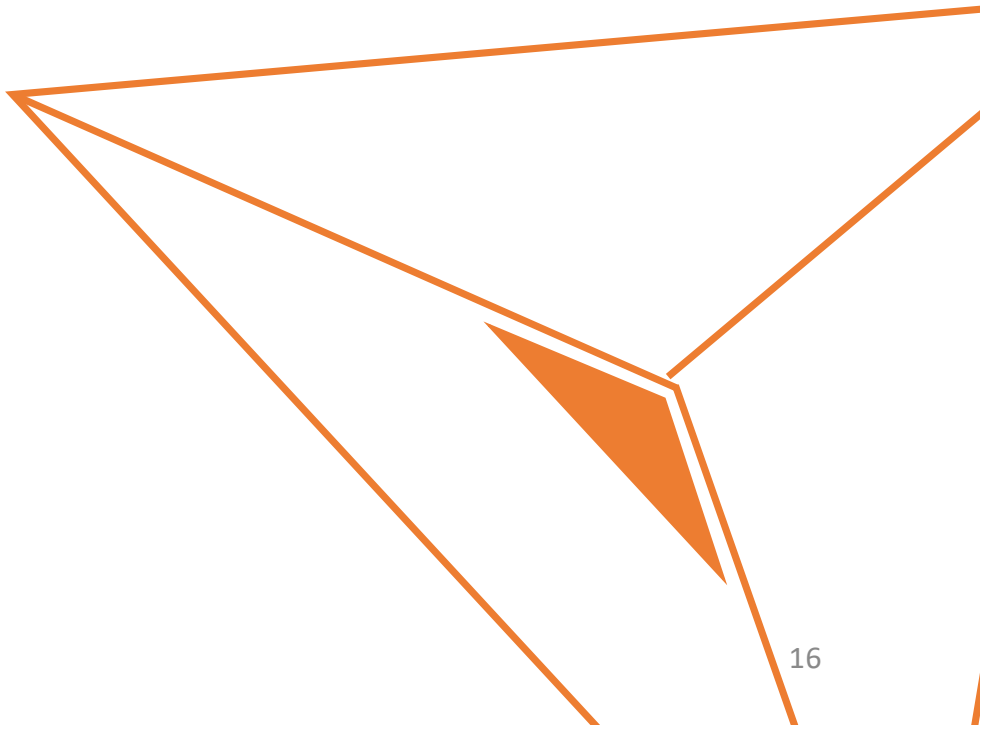
    > library("dplyr")
- Library
  - Where various functional packages are stored

1.12 Package & Library

- Commonly used packages in this course
  - Read excel files: `readxl`
  - Restructure and aggregate data:`dplyr, tidyr`
  - Skewness and kurtosis: `psych`
  - Pivot table: `rpivotTable`
  - Trendlines: `spatialEco`
  - Forecasting; `forecast`, `TTR`
  - Parallel coordinates chart: `GGally`, `ggparallel`
  - Scatterplot matrix: `car`
  - Classification: `class`
  - Linear optimization: `lpSolve`, `lpSolveAPI`

Part II

Calculations, Expressions, Variables



PART II

2.1 Operators

- Arithmetic Operators

Operator	Description
+	Addition
-	Subtraction
*	Multiplication
/	Division
^ or **	Exponentiation

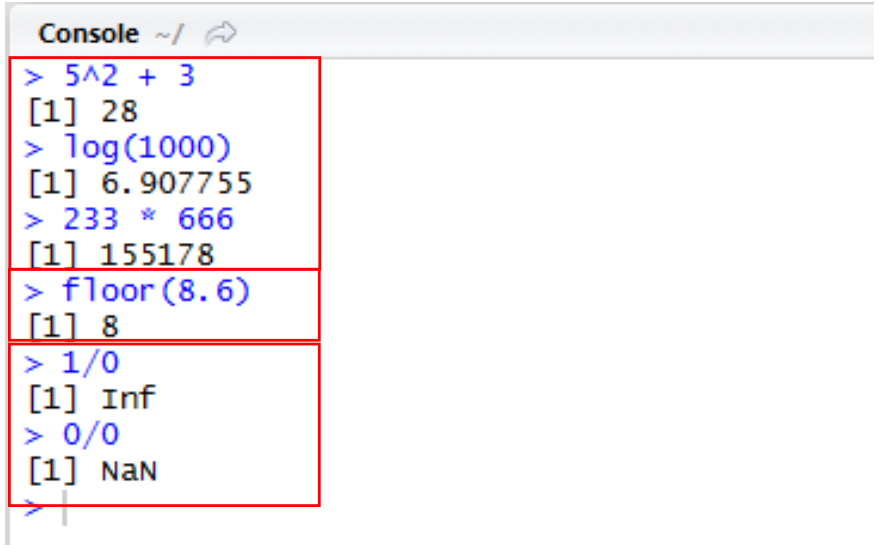
- Relational and logical Operators

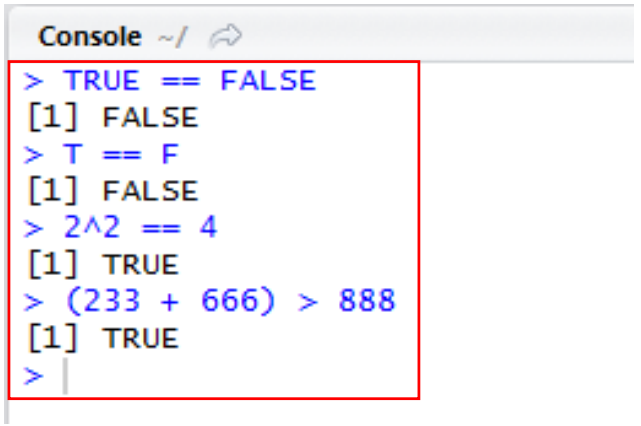
Operator	Description
>	Greater than
>=	Greater than or equal to
<	Less than
<=	Less than or equal to
==	Exactly equal to
!=	Not equal to

2.2 PART II

Expressions & Calculation

- R can be used as a calculator





- Go through Datacamp tutorial:
  - https://campus.datacamp.com/courses/free-introduction-to-r/chapter-1-intro-to-basics-1?ex=2
  - Read the section in Chapter 1 Intro to Basics:
    `> Arithmetic with R`

2.3 PART II Variables

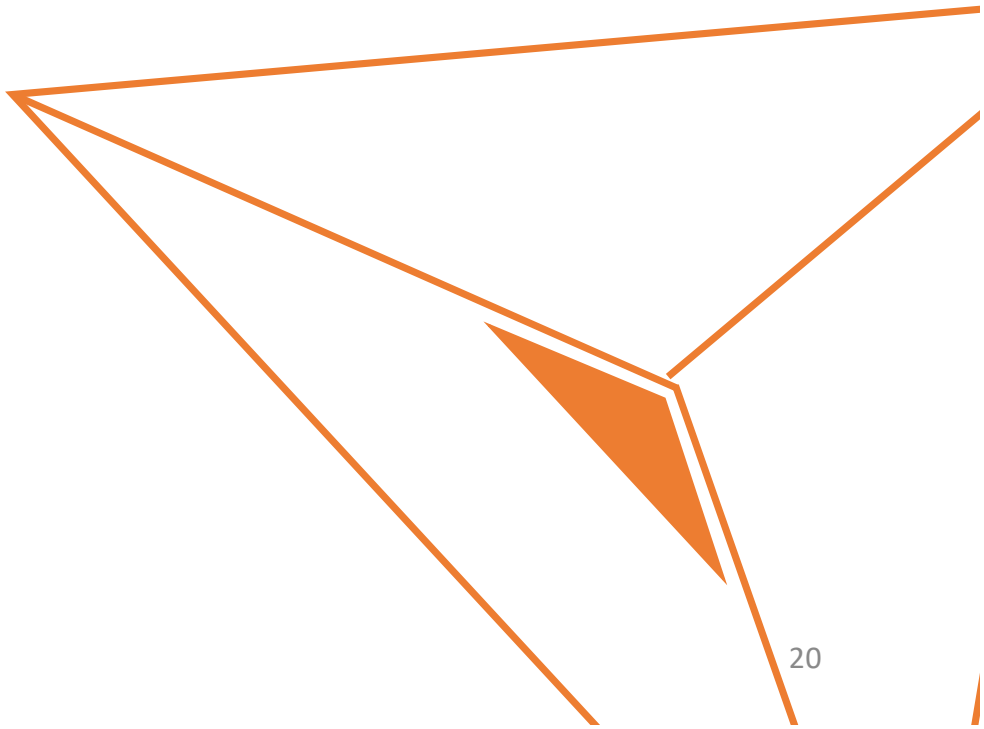
- Similar to other programing languages, you can create variables
  - R is case sensitive
  - Typical variable name rules applies, e.g., should start with an alphabet, etc.
  - Use <- or = for variable assignment

```
> x <- 0.3 * 2
> y <- 0.8
> x < y
[1] TRUE
```

- Go through Datacamp tutorial:
  - https://campus.datacamp.com/courses/free-introduction-to-r/chapter-1-intro-to-basics-1?ex=3
  - Read the sections in Chapter 1 Intro to Basics:
    - > Variable Assignment
    - > Variable Assginment (2)
    - > Variable Assignment (3)
    - > Apples and Oranges

Part III

Data Structures



PART III

3.1 Data Types

- Numeric

```
> x <- 10.5  # <- or = is used for variable assignment
> class(x)   # check the class of object
[1] "numeric"
```

- Integer

```
> y <- as.integer(3) #  coerce an object to a given class
> y <- 3L            # explicitly specify an integer
> is.integer(y)      # is y an integer?
[1] TRUE
```

- Logical

```
> z = x > y  # assign the value of the expression to variable
> z
[1] TRUE
```

3.2 Data Types (Cont.)

- Complex

```
> w = 1 + 2i
> w
[1] 1+2i
>class(w)
[1] "complex"
```

- Character

```
> y <- as.character(3)
> class(y)
character
> fname = "Harry"; lname = "Potter"
> paste(fname, lname)  # Concatenate two char values
[1] "Harry Potter"
```

3.3 PART I

RStudio Cont.

- Go through Datacamp tutorial:
  - https://campus.datacamp.com/courses/free-introduction-to-r/chapter-1-intro-to-basics-1?ex=7
  - Read the sections in Chapter 1 Intro to Basics:

    `> Basic data types in R`

    `> What's that data type?`

Read Page 27-29 of ***Business Analytics Using R – A Practical Approach***

- Data Structure in R

	Linear	Rectangular
All Same Type	Vector	Matrix
Mixed	List	Data Frame

- Vector: a sequence of data elements of the same basic type

```
> a <- c(1,2,3)              # c is a concatenate function
> a                          # numeric vector
[1] 1 2 3
> a[1]                       # access vector value by index
[1] 1
> b <- c("one", "two", "three") # character vector
> b
[1] "one" "two" "three"
> c <- c(TRUE, FALSE, TRUE)     # logical vector
> c
[1] TRUE FALSE TRUE
> x <- c (10, T, F)          # auto coercion will be performed
> x
[1] 10 1 0                   # class is “numeric”
```

3.6 PART III

Vector Cont.

- Go through Datacamp tutorial:
  - https://www.datacamp.com/courses/free-introduction-to-r
  - Complete all sections in Chapter 2 Vectors:

    `>From “Create a vector” to “Advanced selection”`

Read Page 27 of ***Business Analytics Using R – A Practical Approach***

PART III

3.7 Matrix

- Matrix: a collection of data elements arranged in a two-dimensional rectangular layout

```
> A <- matrix(data=c(1,2,3,4,5,6), nrow=2, ncol=3)
> A
     [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
> B <- matrix(data=c(1,2,3,4,5,6), nrow=2, ncol=3, byrow = TRUE)
> B
     [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
> B[2,3]              # element at 2nd row, 3rd column
> B[2, ]              # the 2nd row
> A[ ,c(1,3)]         # the 1st and 3rd columns
```

3.8 PART III

Matrix Cont.

- Go through Datacamp tutorial:
  - https://www.datacamp.com/courses/free-introduction-to-r
  - Complete all sections in Chapter 3 Matrices:

```
> From “What’s a matrix” to “A little arithmetic with
  matrices”
```

Read Page 30-31 of ***Business Analytics Using R – A Practical Approach***

- Vectors can be combined into matrices

```
> v1 <- c(1,2,3,4)
> v2 <- c(5,6,7,8)
> cbind(v1, v2)          # combine the columns
  v1 v2
[1,]  1  5
[2,]  2  6
[3,]  3  7
[4,]  4  8
> rbind(v1, v2) # combine the rows
   [,1] [,2] [,3] [,4]
v1    1    2    3    4
v2    5    6    7    8
```

3.10 PART III List

- List: a generic vector containing other objects

```
> list1 <- list(2.33, 6L, TRUE, "Hello")
> list1
[[1]]
[1] 2.33
[[2]]
[1] 6
[[3]]
[1] TRUE
[[4]]
[1] "Hello"
> a <- c(1,2,3)
> b <- c(T, F)
> list2 <- list(first=a, second=b) # assign names to list members
> list2[1]                        # reference by numeric index
> list2$first                     # reference by name
[1] 1 2 3
```

3.11 PART III List Cont.

- Go through Datacamp tutorial on RStudio
  - https://www.datacamp.com/courses/free-introduction-to-r
  - Complete all sections in Chapter 6 Lists:
    > From "Lists, why would you need them" to "Adding more movie information to the list"

Read Page 34 of ***Business Analytics Using R – A Practical Approach***

PART III

3.12 Data Frame

- Data frame: a list of vectors of equal length
  - Used for storing data tables

```
> d <- c(1, 2, 3, 4)
> e <- c("red", "yellow", "blue", NA)
> f <- c(T, T, T, F)
> mydata <- data.frame(d, e, f)      # create data frame
> names(mydata) <- c("ID", "Color", "Passed")    # set names
> mydata
  ID  Color Passed
1  1    red   TRUE
2  2 yellow   TRUE
3  3   blue   TRUE
4  4   <NA>  FALSE
> mydata$ID
[1] 1 2 3 4
```

3.13 PART III
Data Frame Cont.

- Go through Datacamp tutorial on RStudio
  - https://www.datacamp.com/courses/free-introduction-to-r
  - Read all the sections in Chapter 5 Data Frames:
    > From "What's a data frame" to Sorting your data frame

Read Page 32-33 of ***Business Analytics Using R – A Practical Approach***

3.14 Data Frame (Cont.)

- Example: build-in data frame


```
> mtcars
> mtcars[1,2]                  # reference by numeric index
> mtcars["Mazda RX4", "cyl"] # reference by row & col names
> nrow(mtcars)                 # number of data rows
> ncol(mtcars)
> head(mtcars)                 # preview
```

  - Retrieve data frame column vector

```
> mtcars[[9]]; mtcars[,9]; mtcars[["am"]]; mtcars$am
```

  - Retrieve data frame column slice

```
> mtcars[9]; mtcars["am"]; mtcars[c("mpg", "hp")]
```

- Example: build-in data frame (cont.)
  - Descriptive statistics

```
> min(mtcars$mpg)
> max(mtcars$mpg)
> mean(mtcars$mpg)
> median(mtcars$mpg)
> quantile(mtcars$mpg)          # quartile
> quantile(mtcars$mpg, c(0.05, 0.1, 0.9, 0.95))
> IQR(mtcars$mpg)               # inter-quartile range
```

PART III

3.17 Factor & Array

- Factor: a vector object used to specify categorical variables that have a fixed number of levels

```
> gender <- factor(c("Female", "Male", "Female", "Male", "Female"))
> levels(gender)                # levels of a factor
[1] "Female" "Male"
> nlevels(gender)               # number of levels of a factor
[1] 2
```

- Array: a multiply subscripted collection of data elements
  - Similar to matrices, but can have more than two dimensions

```
> z <- 1:24
> dim(z) <- c(2, 4, 3)     # dimensions 2×4×3
> z      # a 3d array consisting of 24 numbers in a sequence 1:24
```

Go through Datacamp tutorial:

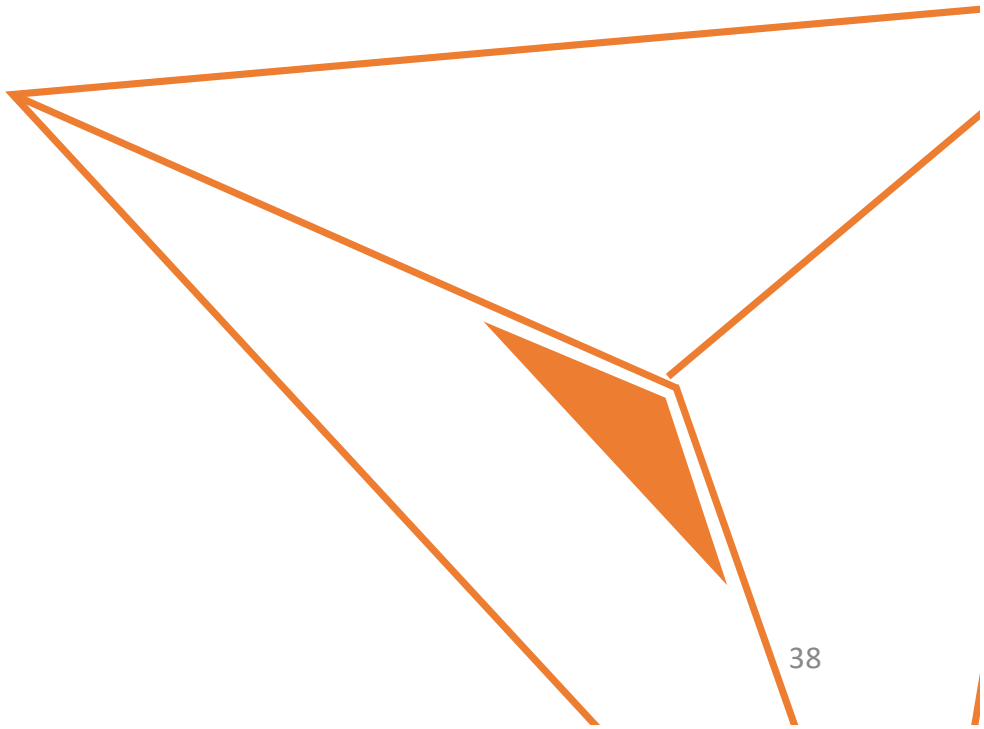
- https://www.datacamp.com/courses/free-introduction-to-r
- Complete all sections in Chapter 4 Factors:

> From "Lists, why would you need them" to "Adding more movie information to the list"

Read Page 35 of ***Business Analytics Using R – A Practical Approach***

Part IV

Read/Write Data



- Read text file with data separated by tabs/space

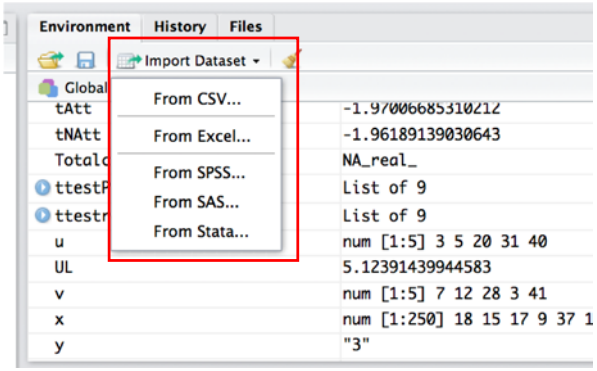
```
> mydata <- read.table("infile.txt", header=TRUE, sep="\t")
```

- Read csv file with data separated by comma

```
> mydata <- read.csv("infile.csv", header=TRUE, sep=",")
```

- Read data from the web

```
> mydata <-read.table
  ("https://archive.ics.uci.edu/ml/machine-learning-
  databases/iris/iris.data", sep=",",
  col.names=c("sepal_len", "sepal_width", "petal_len",
  "petal_width", "species"))
```



- Read data from Excel file
  - Packages: e.g. `readxl`
  - Example: `readxl` Package

```
> library("readxl")
> mydata <- read_excel("infile.xlsx", sheet="sheet_name",
  col_names = TRUE)
```

4.3 PART IV
Write Data

- Export to a tab delimited text file

```
>write.table(mydata, file="outfile.txt", sep="\t")
```

- Export to a comma separated csv file

```
>write.csv(mydata, file="outfile.csv", row.names=TRUE,
 col.names=TRUE)
```

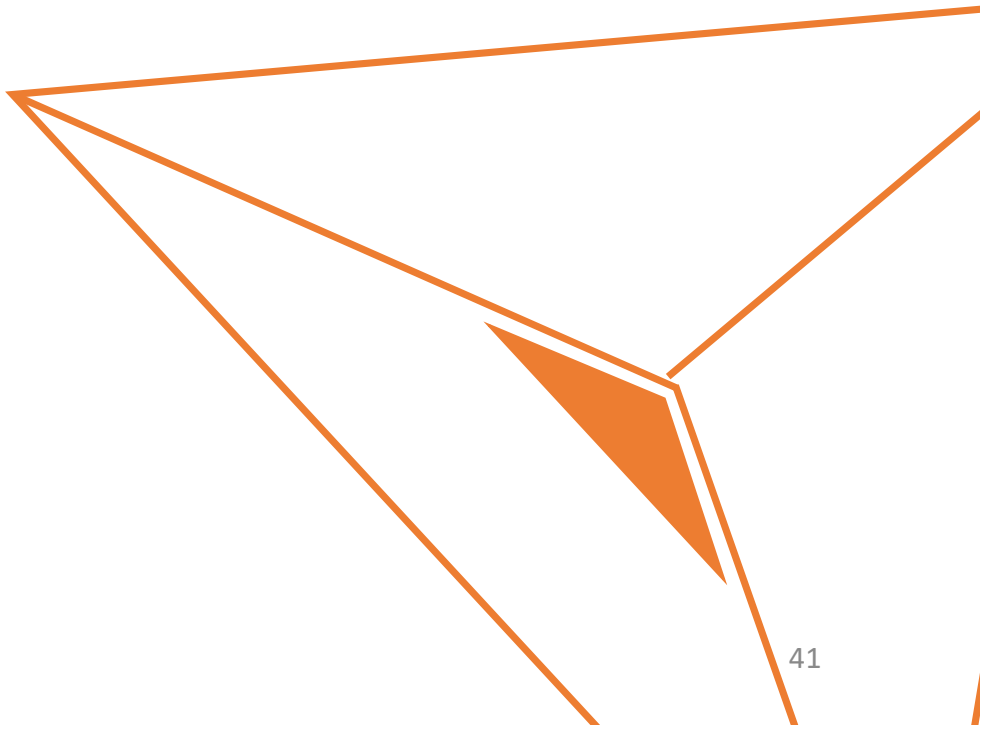
- Export to an Excel spreadsheet

```
>library("xlsx")
>write.xlsx(mydata, "outfile.xlsx", sheetName =
 "Sheet1", col.names = TRUE, row.names = TRUE)
```

Refer to Chapter 4.b.iv on Reading and Importing Excel Files:
https://www.datacamp.com/community/tutorials/r-tutorial-read-excel-into-r

Part V

Graphics

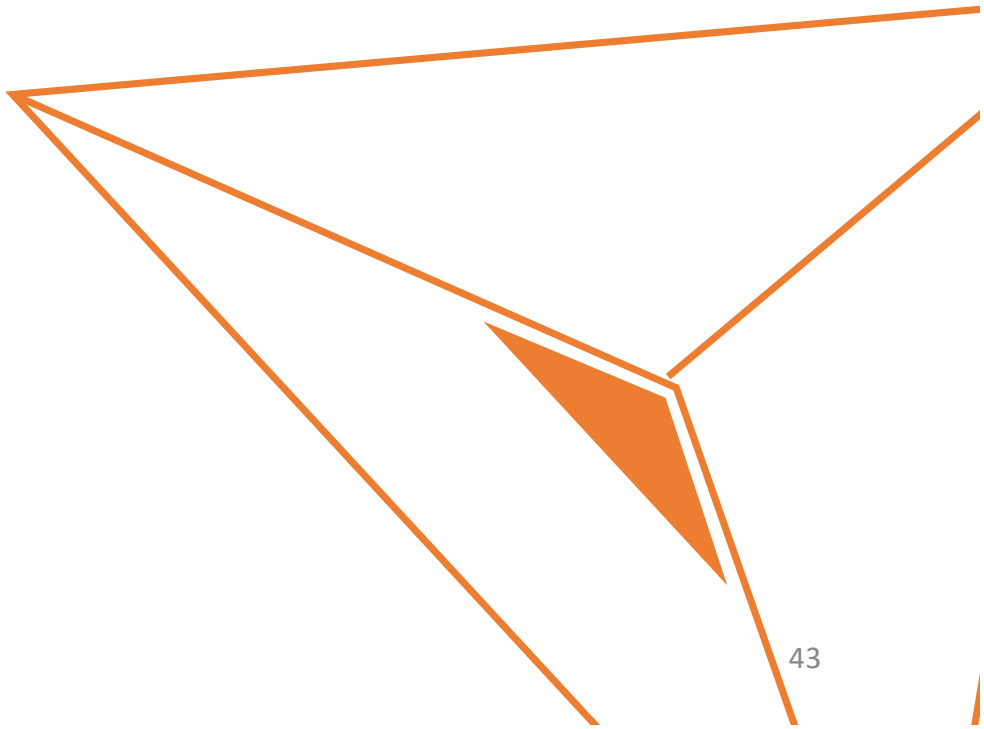


PART V

5.1 Plot

Chart Type	R Functions
Pie Chart	pie(x, labels, radius, main, col, clockwise)
Bar Chart	barplot(H, xlab, ylab, main, names.arg, col)
Box Chart	boxplot(x, data, notch, varwidth, names, main)
Histogram	hist(v,main,xlab,xlim,ylim,breaks,col,border)
Line Graph	plot(v,type,col,xlab,ylab)
Scatterplots	plot(x, y, main, xlab, ylab, xlim, ylim, axes)

References



References

- *The R Manuals*. Available online https://cran.r-project.org/manuals.html
- *An Introduction to R* by Venables, W. N., Smith, D. M., & the R Development Core Team. (2004). Available online: https://cran.r-project.org/doc/manuals/R-intro.pdf
- *The R Book* by Crawley, M. J. (2012).John Wiley & Sons. Available online: http://onlinelibrary.wiley.com.libproxy1.nus.edu.sg/book/10.1002/9781118448908
- *Business Analytics Using R – A Practical Approach* (2017). Dr. Umesh R. Hodeghatta and Umesha Nayak. Available online: https://link-springer-com.libproxy1.nus.edu.sg/book/10.1007%2F978-1-4842-2514-1


THANK YOU