



LECTURE 2: INTRODUCTION TO R CONT

C91AR: Advanced Statistics using R

Dr Peter E McKenna

2025-01-17

LECTURE & COMPUTER LAB READING

- PsyTeachR Chapter on Programming Basics
- R for Data Science chapter on Workflow
 - will cover this in more detail in the lab and as we progress on the course

SESSION OBJECTIVES

- Show you how to set up a project in R
- Show you how to create subdirectories in your project
- Cover some basic principles of programming R
- What we don't finish today we can revisit in the computer lab
- **Feel free to discuss your code with your neighbour if you're stuck**

COMPUTER LAB OBJECTIVES (FOR THURSDAY)

- Pick up where to finish today in this session
- Focus on creating an RMarkdown document

USING PROJECTS

- In R, we create projects which become the directories for our work
- It's a bit like having a folder for each of your subjects/projects.
- Make sure you name your projects appropriately
- Unless you are adding a project to another related project, use the **New Directory** option and fill out the required details

SOME IMPORTANT RULES ABOUT DIRECTORIES IN R

- DO NOT create a folder called ‘R’, as this is the default title R gives to its own namespace, which will lead to conflicts and unusual behaviour
- Good practice is to create R projects in designated project folders, and I will show you how to do this
- Creating an **.RProj** creates a project directory, avoiding any potential confusion R may experience when switching between projects

DEMO: SETTING UP A PROJECT IN R

- Let's set up our project and create
- We'll create an R script today for the exercise, but we'll move over to RMarkdown files come the tutorial.
- Call your R script "intro_to_R.R"

CREATE SUB-DIRECTORIES

Relevance

Code

- Once you are in to your new project you need to create a set of useful folders
- To do so, type the following code lines one at a time into the Console and press enter
- We will be adding files to these folders throughout the course

DEMO: BASIC R PROGRAMMING

```
1 # Using operators  
2  
3 3 + 5  
4  
5 12/7  
6  
7 # Assigning variables  
8  
9 x <- 5  
10  
11 7 -> x
```

```
1 # some simple maths
2
3 x <- 23
4
5 y <- 67
6
7 area_rect <- x * y
```

BEFORE I CONTINUE, DO YOU HAVE ANY QUESTIONS?

```
1 # BMI
2
3 height <- 1.8
4
5 weight <- 80
6
7 bmi <-
8   (weight/(height^height)) # write over 2 lines to keep things tidy
9
10 bmi <-
11   (height/(weight^weight))
```

R FUNCTIONS

```
1 # Using functions
2
3 sqrt(4)
4
5 a <- 49
6
7 sqrt(a)
8
9 round(3.14159)
10
11 args(round) # shows you the format of the function input
12
13 round(3.14159, digits = 2)
```

VECTORS AND DATA TYPES

```
1 test_scores <-
2   c(3, 5, 2, 8, 7) # use concatenate to create a vector of (numeric) values
3
4 test_scores
5
6 students <-
7   c("Roland", "Sophie", "Max", "Teresa", "Mandy") # use concatenate to create a vector of strings
8
9 # length function
10 length(test_scores)
11
12 # class function
13 class(students)
```

GIVEN WHAT YOU HAVE LEARNED ABOVE, WHAT DO YOU THINK THE FOLLOWING CODE DOES?

```
1 students <-  
2   c(students, "Dennis")
```

FURTHER EXERCISES (TIME DEPENDENT)

```
1 # Mixed data types ----  
2  
3 test_answers <-  
4   c("a", 5, 8, TRUE, 3i)  
5  
6 class(test_answers)  
7  
8 str(test_answers)
```

```
1 # Sub-setting vectors ----  
2  
3 test_answers  
4  
5 test_answers[2]  
6  
7 test_answers[c(3, 2)]  
8  
9 test_answers[c(9, 23, 5)]
```

```
1 # Conditional sub-setting
2
3 test_scores <-
4   c(test_scores, 10, 2, 1, 3, 6)
5
6 test_scores > 3
7
8 test_scores[test_scores > 3]
9
10 test_scores[test_scores <= 2 | test_scores == 6]
11
12 students[students == "Max" | students == "Mandy"]
```

CLEANUP

```
1 # Clear data
2 rm(list = ls())  # Removes all objects from environment
3
4 # Clear packages
5 p_unload(all)  # Remove all contributed packages
6
7 # Clear plots
8 graphics.off()  # Clears plots, closes all graphics devices
9
10 # Clear console
11 cat("\014")  # Mimics ctrl+L
```

SUMMARY

- Set up an R project (.RProj file)
- Created relevant subdirectories for the course (for use later on)
- Covered some basics of R programming using an R script to give you some hands on experience programming in R

COMPUTER LAB

- Creating RMarkdown files
- Writing text and code in RMarkdown
- Writing tidy code (in more detail)

