

# Introduction to Data Analysis in R

Caitlin C. Mothes, PhD and Katie Willi

2023-08-17



# Contents



# Chapter 1

## Welcome!

This site hosts the course curriculum for Colorado State University's **WR 696** course: *Introduction to Data Analysis in R*

### Set up Instructions

As your first step, please follow the instructions on the R Setup page to make sure you have all the necessary software installed on your computer before the class starts.

### Navigating this site

The table of contents on the left allows you to navigate to the lesson for each week of the course. Each lesson will walk you through the topic(s), analysis, etc. for that week. There are exercises at the end of the lesson that will be that week's homework assignment.

Homework will be submitted through Canvas and exercise questions (including any code, figures, etc.) must be submitted as a rendered R Markdown document in Word or HTML format. The Intro to R and R Markdown lesson will walk through how to create these R Markdown documents; you will use these R Markdowns as the basis for writing your code/answers to the exercises and then render them to either a Word or HTML report which is what you will submit through Canvas.

## Goals

The broad goal of this course is to introduce you to the skills necessary for a career that involves data science - namely manipulating data sets and performing statistical tests in R. The more specific goals of this course are to teach students to:

1. Navigate the RStudio interface and create R Markdown documents for reproducible reporting;
2. Utilize R packages and functions to manipulate and analyze data effectively and apply data wrangling techniques using the `{tidyverse}` framework;
3. Differentiate between various data types and structures within R; and
4. Explore comparative analyses, linear regression, and trend analysis techniques to reveal patterns in data.

## Approach and Expectations

This class is flipped, meaning all materials for the week ahead must be reviewed before class. To encourage this we will have weekly quizzes on pre-class content each Monday before we dive into the assignment.

So without lectures in class what do we do together? We code! This class has almost four hours of contact time per week, and we design lessons so that you should be able to finish your assignments in class. The flipped class allows for deeper discussion about the common pitfalls of coding and allows for collaborative work time with your fellow classmates.

Generally we will do a quick live-code review of concepts from the assignment and the pre-class materials, but more than 1.5 hours per day will be dedicated to you coding and working on the assignment in class.

As such, coming to class is a vital part of how you can be successful and we fully expect you to be there every day.

We also will actively encourage a collaborative coding environment where students help each other and discuss the best approach to solving various coding problems. We also hope that outside of class, you will use our Teams channel to discuss code issues!

We will always send announcements with assignments, web links, and other updates through Canvas. The course syllabus will also be posted on Canvas.

## Class Schedule

| Week | Data     | Content  |
|------|----------|--|
|      | (Monday) |  |
| 1    | 8/22     | Before class: Primers Basics and R Markdown; download R and RStudio<br>Introduction to R, RStudio, RMarkdown |
| 2    | 8/28     | Before class: Work with Data<br>Exploratory data analysis  |
| 3    | 9/4      | Before class: Visualize Data<br>Data visualization   |
| 4    | 9/11     | <b>No Class Monday 9/4!</b> Before class Wednesday: Tidy Your Data<br>T-tests, ANOVA                         |
| 5    | 9/18     | Before class: Iterate<br>Linear modelling pt. 1  |
| 6    | 9/25     | Before class: Write Functions<br>Linear modelling pt. 2  |
| 7    | 10/2     | Modelling assumptions, power   |
| 8    | 10/9     | PCA  |

## Additional introductory resources

If you are looking to learn even more outside of what this class offers, below are some great introductory R resources:

- Stat 158 - Vectors, data frames, installing R, etc...
- RStudio Materials - A series of videos, books, and more to get you started in R.

### 1.0.0.1 Tidyverse Introduction

- R for Data Science - Covers all of the basic intro material, from a tidyverse perspective. As discussed, this is one way to find solutions in R, it happens to be my preferred way, but there are lots of Base R ways that work just fine!
- Stat 159 - A CSU specific course for an intro to the tidyverse
- R Markdown - The primary book for learning more about R Markdown and all of its quirks.

- Cheatsheets - Short, clear documents that cover so much material from dplyr to shiny apps. Great for quick references. We find the `rmarkdown` and `ggplot2` ones particularly useful!



## Chapter 2

# Setup Instructions

This tutorial walks you through all the **required** setup steps for using R and RStudio in this course. Please use our class's Teams Channel to post any questions/problems that you encounter.

### 2.1 Install R and RStudio

**R** is an open source language and software environment for statistical analysis and graphics (plus so much more!). You must **first download the R software (for free) here: <https://www.r-project.org/>**.

- Click the download link based on your operating system (OS). Then, for Mac users, install the latest release based on your macOS. For Windows users, click the 'install R for the first time' link.

*Note: If you already have R installed, you **must have at least version 4.0.0. or greater**, but it is best to have the most recent version installed (4.3.1)*

**RStudio** is a (also free) R Integrated Development Environment (IDE) that provides a built-in editor and other advantages such as version control and project management. **Once you have the R software installed on your computer, you can install RStudio Desktop here: <https://posit.co/download/rstudio-desktop/>**.

- Under **Step 2** click the download RStudio Desktop button.

## 2.2 Package Installation

While the R software comes with many pre-loaded functions (referred to as ‘base R’ functions) to perform various operations in R, there are thousands of R packages that provide additional reusable R functions. In order to use these functions you need to first install the package to your local machine using the `install.packages()` function. Once a package is installed on your computer you don’t need to install it again (but you may have to update it). Anytime you want to use the package in a new R session you can load it with the `library()` function.

We will be working in RStudio for this entire course, so after you have installed both R and RStudio, **open a new session of RStudio**.

You will learn more about the ins and outs of RStudio in class, but for set up purposes you will just be running code in the Console. Normally you want to save the code you write, but since package installation is only needed once (unless you are working on a new machine or need to update any packages) you can execute this directly in the console.

Run the following three lines of code (one at a time) in the console. *You can click the copy button in the upper right corner when you hover over the code chunk, then paste that after the > in the Console. Spelling and capitalization is important!*

**Install the {tidyverse} package.** The Tidyverse is actually a collection of multiple R packages designed for data manipulation, exploration, and visualization that you are likely to use in every day data analysis. When you install the Tidyverse, it installs all of these packages, and you can later load all of them in your R session with `library(tidyverse)`. Since you are installing multiple packages here this may take a little while.

```
install.packages("tidyverse")
```

**Install the {palmerpenguins} package.** This is a data package that installs a couple of spreadsheets you can load and work with in R.

```
install.packages("palmerpenguins")
```

**Install the {rmarkdown} package.** Later on in the course you will be working in and rendering R Markdown files and reports. R Markdown is a notebook style interface integrating text and code, allowing you to create fully reproducible documents and render them to various elegantly formatted static or dynamic outputs.

You can learn more about R Markdown at their website, which has really informative lessons on their Getting Started page, and see the range of outputs you can create at their Gallery page.

```
install.packages("rmarkdown")
```

To see if you successfully installed all three packages, use the `library()` function to load the packages into your session. You should either see nothing printed to the console after running `library()`, or in the case of the `tidyverse` you may see some messages printed. As long as there are no **error** messages, you should be all set! Please use our class's Teams Channel for assistance if you get any error messages.

```
library(tidyverse)
library(palmerpenguins)
library(rmarkdown)
```

