# Getting started with R

FOR 128: Lab 1

Solutions

2024-08-29

#### Welcome

Welcome to Lab 1! Today, we'll get started with a few data science tools (R, RStudio, and Quarto), learn to use R as a calculator, and get a glimpse into graphics with R.

#### Learning objectives

- Make sure R and RStudio are set up correctly.
- Set up folders on your computer for saving your work.
- Begin working with Quarto (.qmd) documents.
- Learn to save and render your work with Quarto.
- Start using R as a calculator.
- Gain familiarity with graphics in R

# Deliverables (i.e., what to put in the lab drop box)

Upload your rendered PDF (lab\_01.pdf) and Quarto (lab\_01.qmd) document to the lab drop box. Make sure the Quarto document properly renders to PDF.

# Collaborator(s)

List	any	collab	orators	you	worked	with	in t	the s	space	below	

# Part 1: Setting up your workspace

#### Exercise 1

Create a folder on your computer where you'll store all your work for this course. We do not suggest putting this on your Desktop. A better place would be your Documents folder or your home directory. Make sure to give the folder an informative name like for128 or for128\_f24 or something else. Put this file (lab\_01.qmd) in your new folder.

Now, run the following code chunk (press the green arrow on the far right of the code chunk). The output should include this file. What do you think the list.files() function does in this case? For all lab exercises in this course, please answer between the rows of dashed lines.

#### list.files()

```
[1] "datasets" "FEF_trees.csv"
[3] "lab_01_solutions.qmd" "lab_01_solutions.rmarkdown"
[5] "lab_01.qmd" "lab_02.qmd"
[7] "lab_03_example.R" "lab_03.html"
[9] "lab_03.qmd" "lab_04.qmd"
[11] "lab_05.qmd"
```

Shows the files in the current directory.

### Exercise 2

Go to D2L and download the datasets.zip file. This file contains datasets needed to work with our textbook. Use your computer to "unzip" the file (a folder called "datasets" should appear). Move this folder into the folder you just created in Exercise 1. For now, we'll leave this directory alone, but we will use it for future labs.

Now, run the following code chunk. What do you think specifying the path argument in the list.files() function did?

```
list.files()
```

```
[1] "datasets" "FEF_trees.csv"
[3] "lab_01_solutions.qmd" "lab_01_solutions.rmarkdown"
[5] "lab_01.qmd" "lab_02.qmd"
[7] "lab_03_example.R" "lab_03.html"
[9] "lab_03.qmd" "lab_04.qmd"
[11] "lab_05.qmd"
```

#### list.files(path = "datasets/")

```
[1] "BoundaryExamples"
                                   "city_block_survey.csv"
 [3] "doubleSamplingData.csv"
                                   "dplyrJoinAllometric"
 [5] "FACE"
                                   "FACE_aspen_core_growth.csv"
 [7] "FEF_trees_missing.csv"
                                   "FEF_trees.csv"
 [9] "FEF_trees.tsv"
                                   "FEF_trees.txt"
                                   "HF"
[11] "FVS_NE_coefficients.csv"
[13] "Honer_coefficients.csv"
                                   "Jenkins_coefficients.csv"
[15] "loblolly_trees.csv"
                                   "Loblolly.csv"
[17] "mn_trees_subset.csv"
                                   "mn trees.csv"
[19] "mnf-plot-55.csv"
                                   "neighborhood survey.csv"
                                   "PEF"
[21] "PA"
[23] "plotSamplingExample"
                                   "RDS-2013-0015"
[25] "RDS-2016-0016"
                                   "samplingWithCovs.csv"
[27] "tidyPlotSummaryExample"
                                   "tidyPlotSummaryExample.zip"
[29] "TODO"
                                   "TODO~"
[31] "tree_list_2_plots.csv"
                                   "tree_list_2_plots.csv~"
[33] "trees_1_tidyverse.csv"
                                   "twin-lakes.csv"
[35] "two_stands_wo_ht_vol.csv"
                                   "USFS_species_codes.csv"
[37] "WeatherKLAN2014.csv"
                                   "WeatherKLAN2014Full.csv"
```

Now, the files in the data subdirectory are listed.

#### Exercise 3

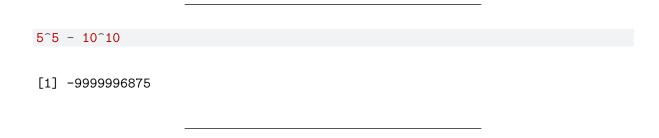
Render this document by clicking "Render" towards the top of the RStudio window. What does it look like? Does the output of your code from Exercises 1 and 2 look the same? If it doesn't, why do you think that is the case?

xercise just with words, so there is no n	eed for a code chunk for
emporary file appear. The student need it appears and attempt to explain why	_
itor	-
in the following code chunk.	-
ollowing line	
in the following code chunk.	-
i	mporary file appear. The student need t appears and attempt to explain why i tor  tor  in the following code chunk.

#### Exercise 6

Do Exercise 2.3 from IFDAR. Note, you'll have to make your own code chunk.

Hint: on Windows and Linux you can create a new code chunk with control + alt + I and on macOS you can make a new code chunk with command + alt + I. Otherwise, you can create a new code chunk by clicking the code dropdown menu in RStudio and then clicking Insert Chunk.



#### Exercise 7

Do Exercise 2.4 from IFDAR. Note, you'll have to make your own code chunk.

```
(5 + 5) / (3 * 4)
[1] 0.8333333
```

# Part 3: A glimpse into graphics

#### Exercise 8

## Part (a)

Run the following code to install the ggplot2 package. Note the eval: false setting. This means this code will not run when you render the document (we don't want to install ggplot2 every time we render the document, rather we'd like to install it just once when we run the chunk by hand). Do not change the eval setting for this code chunk.

```
install.packages("ggplot2")
```

If everything has gone correctly, the following chunk should return TRUE:

```
"ggplot2" %in% installed.packages()
```

[1] TRUE

## Part (b)

Use the rnorm() function to generate 100 random numbers with a mean of 0 and a sd of 1. Save these numbers to an R object called draws.

```
draws <- rnorm(n = 100, mean = 0, sd = 1)
```

# Part (c)

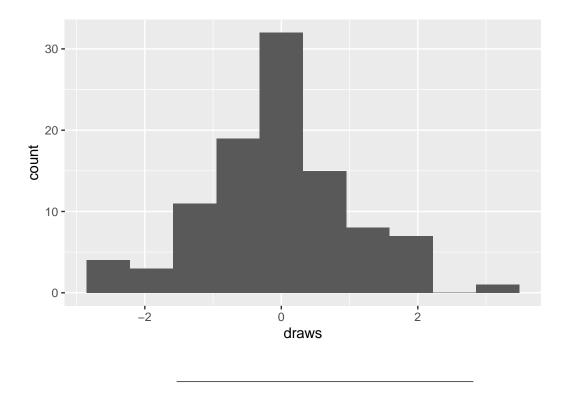
Once you have completed parts (a) and (b), set the eval option from false to true in the following chunk. Then run the following code and comment on what you see. What happens if you change the number of bins from 10 to 20? to 50? to 100? to 1?

Note: You are not expected to understand the following code at this stage of the course, but it is a good glimpse into what we'll do as the course progresses.

```
library(ggplot2)

dat <- data.frame(draws = draws)

ggplot(data = dat, mapping = aes(x = draws)) +
    geom_histogram(bins = 10)</pre>
```



The students should notice that the number of bins increases as bins increases.

# Wrap up

Congratulations! You've made it to the end of Lab 1. Make sure to render your final document and submit both the .pdf and .qmd file to D2L.