

# PLSC 476: Empirical Legal Studies

Christopher Zorn

August 28, 2025

## Two options:

- **Final Exam**

- “Take-home” format...
- December 11-17, 2025

- **Research Paper / Project**

- Empirical “research note” (or other)
- $\approx$  15-20 pages
- Due December 18, 2025

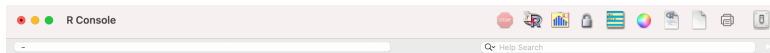
- Either is worth 400 points (40% of total grade)
- Indicate / change preference on/before **November 20, 2025**

## Options:

- Preferred: R + RStudio / Posit
- Also viable: [Stata](#), Python, Javascript, etc.
- Good, but obscure: Julia, Go, MATLAB
- Highly discouraged: SAS, SPSS/PSPP, Minitab, Statistica, etc.

## Options:

- Simplest: Install and run R + RStudio locally
  - **R** (v. 4.5.1, from [cran.r-project.org](https://cran.r-project.org)), and
  - **RStudio (Desktop)** (v. 2025.05.1+513, from <https://posit.co/download/rstudio-desktop/>)
- Almost as simple: [RStudio.cloud](https://posit.co/download/rstudio-cloud/)...
- Here's a “[Beginner's Guide](#)” to RStudio



R version 4.5.1 (2025-06-13) -- "Great Square Root"  
Copyright (C) 2025 The R Foundation for Statistical Computing  
Platform: aarch64-apple-darwin20

R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

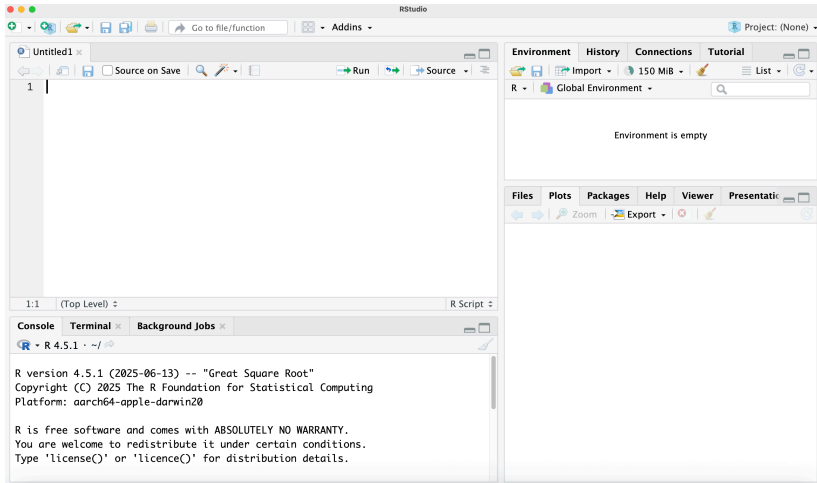
R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.

[R.app GUI 1.82 (8536) aarch64-apple-darwin20]

[History restored from /Users/cuz10/.Rapp.history]

> |



# RStudio (annotated)

This is the "Source" window.

- It's the place where you'll type the code that will then be sent to R.
- It's basically a text editor. You can open text files of any kind here if you want.
- Files that appear here end in (and should be saved with) the extension ".R" (as in "MyCode.R").

You'll spend most of your time working here.

Click here to save your source code. Save often!

Highlight text in the Source window, then click this button to "run" the code.

This is the "Environment" window. It is where you can find all the various "objects" that you create, grouped by object type (data frames, lists, graphs, etc.). Environment is empty.

There's also a "History" tab above; switching to that will show what has transpired in the Console window recently.

This is the "working directory." Anything you save will be saved here, unless you tell the program to save it somewhere else.

This is the "Console." When you run the code in the Source window, the results that aren't graphics appear here.

Type 'demo()' for some demos, 'help()' for on-line help, or 'help.start()' for an HTML browser interface to help. Type 'q()' to quit R.

This window shows various other things. Those things are tabbed above and include:

- Plots (graphs) that you have created
- Packages that are loaded
- Help results (obtained by typing "?XXX" in the Console window, e.g. "?table").

← → ↺ 🏠 🌐 posit.cloud/content/10771278 ☆ 📁 📄 👤 ⋮

Reminder: The ability to publish applications and documents to Posit Cloud has been **deprecated**. That functionality will be removed for all users by the end of July. ✕

**posit** Cloud

Spaces

- Your Workspace
- COVID-19 in PA  
Christopher Zorn
- Supreme Court Thesis  
Richard Mortons
- + New Space

Learn

- Guide
- What's New
- Recipes
- Cheatsheets

Help

- Current System Status
- Posit Community

Info

- Plans & Pricing
- Terms and Conditions

Your Workspace / PLSC 476 - Fall 2025

RAM ⚙️ ⋮ Christopher Zorn

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Add-ins

Untitled1 x

1

Run Source

Environment History Connections Tutorial

R 4.5.1

Import Dataset 97 MiB

R Global Environment

Environment is empty

Files Plots Packages Help Viewer Presentation

New Folder New File Upload Delete Rename More

Cloud > project

Name	Size	Modified
..		
.Rhistory	0 B	Aug 27, 2025, 8:37 PM
project.Rproj	205 B	Aug 27, 2025, 8:37 PM

Console Terminal Background jobs

R 4.5.1 - /cloud/project/

```
R version 4.5.1 (2025-06-13) -- "Great Square Root"
Copyright (C) 2025 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

Connected to your session in progress, last started 2025-Aug-28 00:37:18 UTC (2 minutes ago)
>
```



# Inside the Source Window

This:

```
> table(df$X)
```

... means “Type the phrase ‘table(df\$X)’ on the command line,” or – equivalently – “Type the phrase ‘table(df\$X)’ into your Source code, and then run it.”

# Inside the Source Window

More often, you'll see:

```
with(df, plot(Y~X,pch=19,col="red")) # draw a scatterplot
abline(h=0,lty=2)                    # add a horizontal line at zero
abline(v=0,lty=2)                    # add a vertical line at zero
text(df$X,df$Y,labels=df$names,pos=1) # add labels
```

... which means “Put this block of text into your Source code, and then run it.”

## Note:

- R / RStudio ignores line breaks and white space
- Anything to the right of a “#” is a comment

# R: First Things

## 1. The “working directory”

- Where R “looks for” and “keeps” all the things
- It defaults to some version of the “root” directory on the system
- You probably want to change it more-or-less immediately, to something more useful...

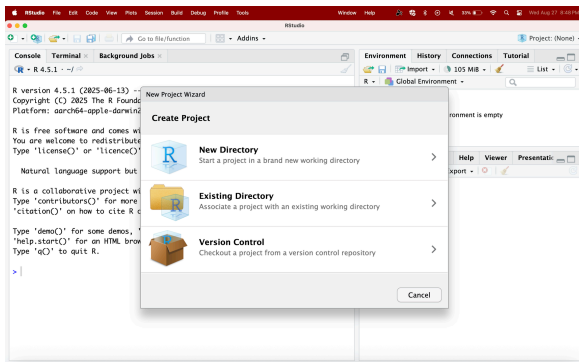
E.g.:

```
> setwd("~/Documents/PLSC476")      # OS-X / Linux  
> setwd("C:/My Documents/PLSC476") # Windows
```

# R: First Things

## 2. RStudio Projects

- Creates a file with the “.Rproj” suffix
- Allows you to keep source code, data, history, etc. in one “place”
- Can also do useful things like auto-load data frames, auto-save on exit, etc.
- Makes the project “portable” (→ enhances reproducibility)
- Some guidance [here](#)



## 3. Packages

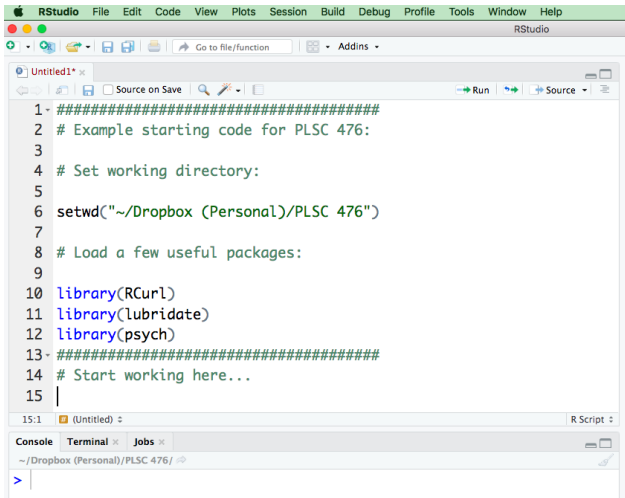
- Bundles of functions, commands, data, and documentation that do specific things (data management, statistics, graphics, etc.)
- To be used, packages first have to be *installed*, using the `install.packages()` command
- Once a package is installed, it needs to be *loaded* for use, using `library()`
- If you try to use a command from a package, and the package is not installed and loaded, you'll get an error (see below)

E.g.:

```
> qqp(SCOTUS$LiberalPct)
Error in qqp(df) : could not find function "qqp"
> install.packages("car")      # install
> library(car)                 # load for use
> qqp(SCOTUS$LiberalPct)      # <-- works
```

# A Source Code Header

Tip: Keep these things at the beginning of your source code,  
e.g.:

A screenshot of the RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Window, and Help. Below the menu is a toolbar with icons for saving, running, and other functions. The main editor window shows a script titled 'Untitled1\*' with the following R code:

```
1 #####
2 # Example starting code for PLSC 476:
3
4 # Set working directory:
5
6 setwd("~/Dropbox (Personal)/PLSC 476")
7
8 # Load a few useful packages:
9
10 library(RCurl)
11 library(lubridate)
12 library(psych)
13 #####
14 # Start working here...
15 |
```

The status bar at the bottom shows the file path as '~/Dropbox (Personal)/PLSC 476/' and the current line as 15:1. The console window is also visible at the bottom.

## 4. Objects

- R is *object-oriented*
- R objects are of different *classes*...
- ...which have different structures and properties
- See (e.g.) [here](#) for more details

E.g.:

```
> str(cars)
'data.frame': 50 obs. of  2 variables:
 $ speed: num  4 4 7 7 8 9 10 10 10 11 ...
 $ dist : num  2 10 4 22 16 10 18 26 34 17 ...
```

# Getting Data

1. You can enter data into R/RStudio by hand (but you'd rather not)
2. You can read data from a local file. The simplest way to do that is if the data is in standard `.csv` format:
  - values in each row separated by commas,
  - rows separated by line breaks, and
  - a single row with comma-separated variable names at the top

E.g.:

```
> DF <- read.csv("Data/SCOTUS-votes.csv")
```



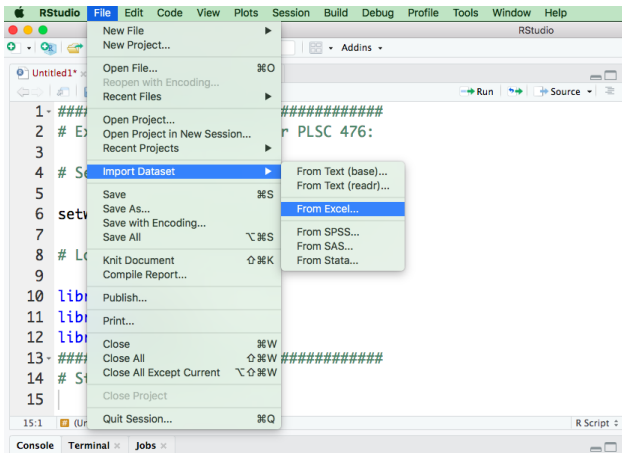
3. You can read a .csv (or, for that matter, a .xls, or other format file) directly from the web, via the file's URL, using various commands (read.csv, fread, etc.).

All of these commands do the same thing:

```
> foo <- read.csv("http://mqscores.wustl.edu/media/2022/court.csv")
> library(readr)
> foo <- read_csv("http://mqscores.wustl.edu/media/2022/court.csv")
> library(data.table)
> foo <- fread("http://mqscores.wustl.edu/media/2022/court.csv")
```

# Getting Data

4. In RStudio, you can read data interactively, via “File → Import Dataset”



## 5. You can import data dynamically via RESTful (and other) APIs...

- Simplest via the `httr` and `jsonlite` packages
- Usually requires knowledge of JSON-formatted data
- A basic tutorial is [here](#).

## Summary:

Data Formats and Packages	
Format	Useful Command(s) / Package(s)
Plain text (.txt, etc.)	<code>read.table</code> , <code>readr</code>
Comma-separated (.csv)	<code>read.csv</code>
Excel (.xls, etc.)	<code>readxl</code> , <code>xlsx</code>
SPSS (.sav, etc.)	<code>haven</code> , <code>foreign</code>
Stata (.dta)	<code>haven</code> , <code>foreign</code>
SAS (.ssd, etc.)	<code>haven</code> , <code>foreign</code>
JSON (.json)	<code>jsonlite</code>
Databases (.adb, .sql, etc.)	<code>DBI</code> , <code>odbc</code>

Examples...

(see `PLSC476-2025-Second-Day-Software.R`)

# Help For Learning R(Studio)

In rough order of preference:

- Quick-R (<http://www.statmethods.net/>)
- [An Introduction to R for Research](#)
- The “[Level-Zero](#)” [R Tutorial](#) (doesn't integrate RStudio, but is otherwise very good)
- [Statistics with R](#)
- The [Do It Yourself Introduction to R](#)
- Also be sure to consult the PLSC 476 “[Useful R Resources](#)” guide (on GitHub).

# How to Figure Things Out in R

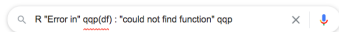
1. One leading question mark ("?foo") = "help."

```
> ?reshape
```

2. Two leading question marks ("??foo") = "search."

```
> ??reshape
```

3. Google is your friend:\*



4. [Stack Overflow](#) is really your friend...

5. [ChatGPT](#) (and other LLMs) is really your friend...

---

\*No, it isn't.