

Introducing R and RStudio

EES 4891/5891

Probability & Statistics for Geosciences

Jonathan Gilligan

Class #3: Tuesday, January 14 2025

Learning Goals

Learning Goals

1. Get RStudio configured for this course
2. Learn about working with R and RStudio
3. Import data from a text file into R
4. Use R to visualize global temperature data

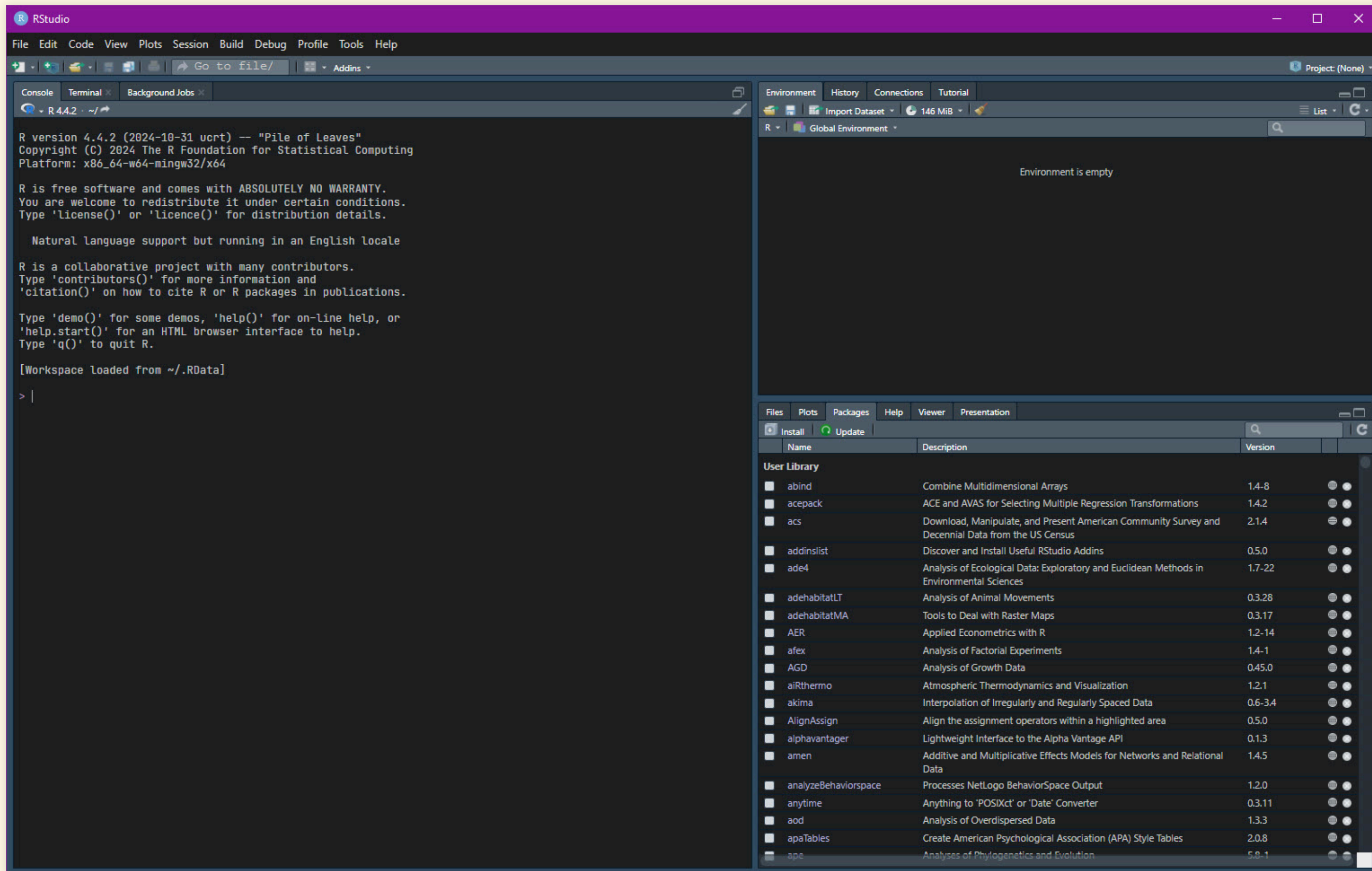
Announcement

Announcement

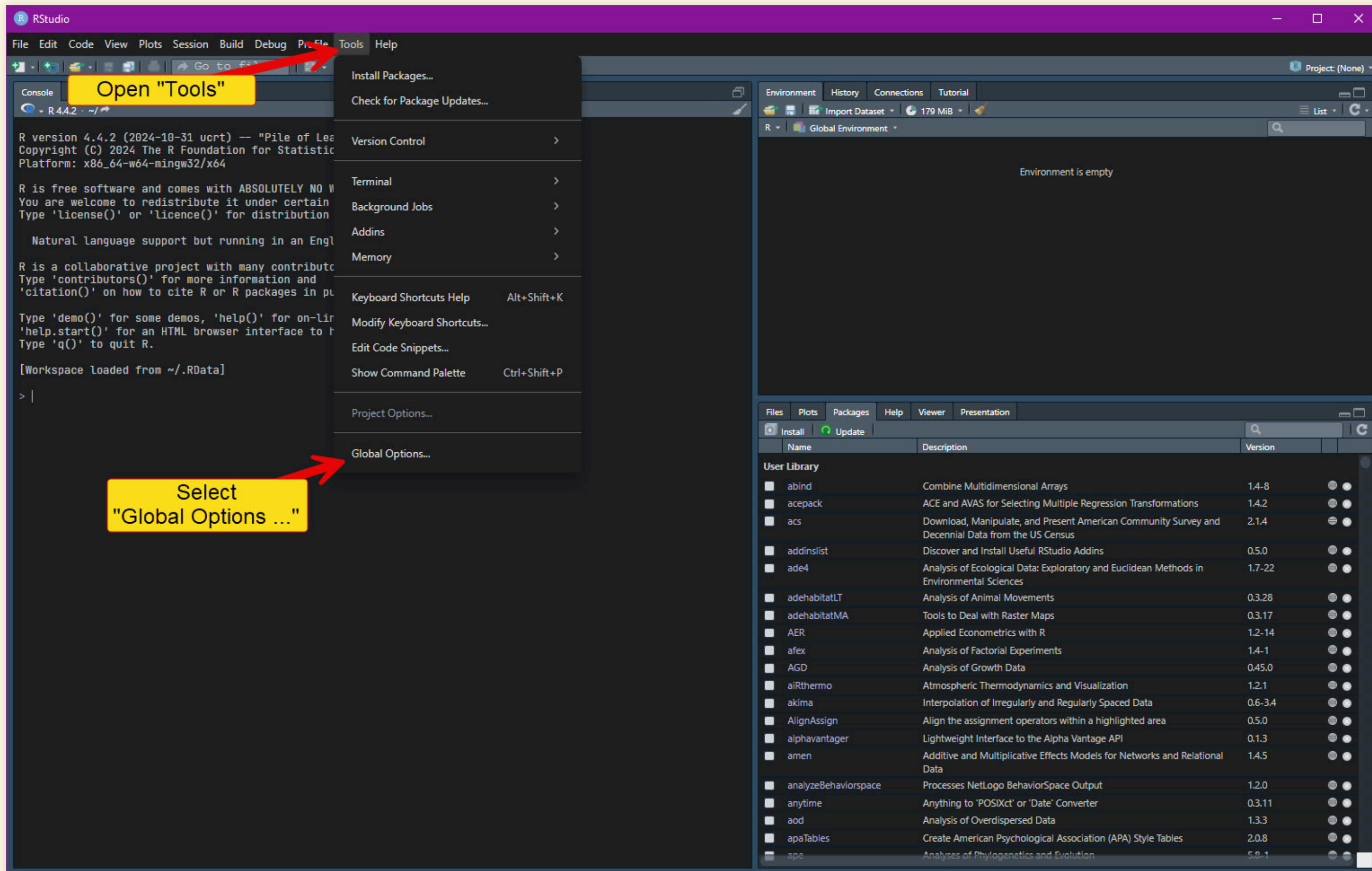
- Tomorrow (Wednesday) my office hour will be shortened: 1:30–3:00.

Getting Started with RStudio

Getting Started with RStudio

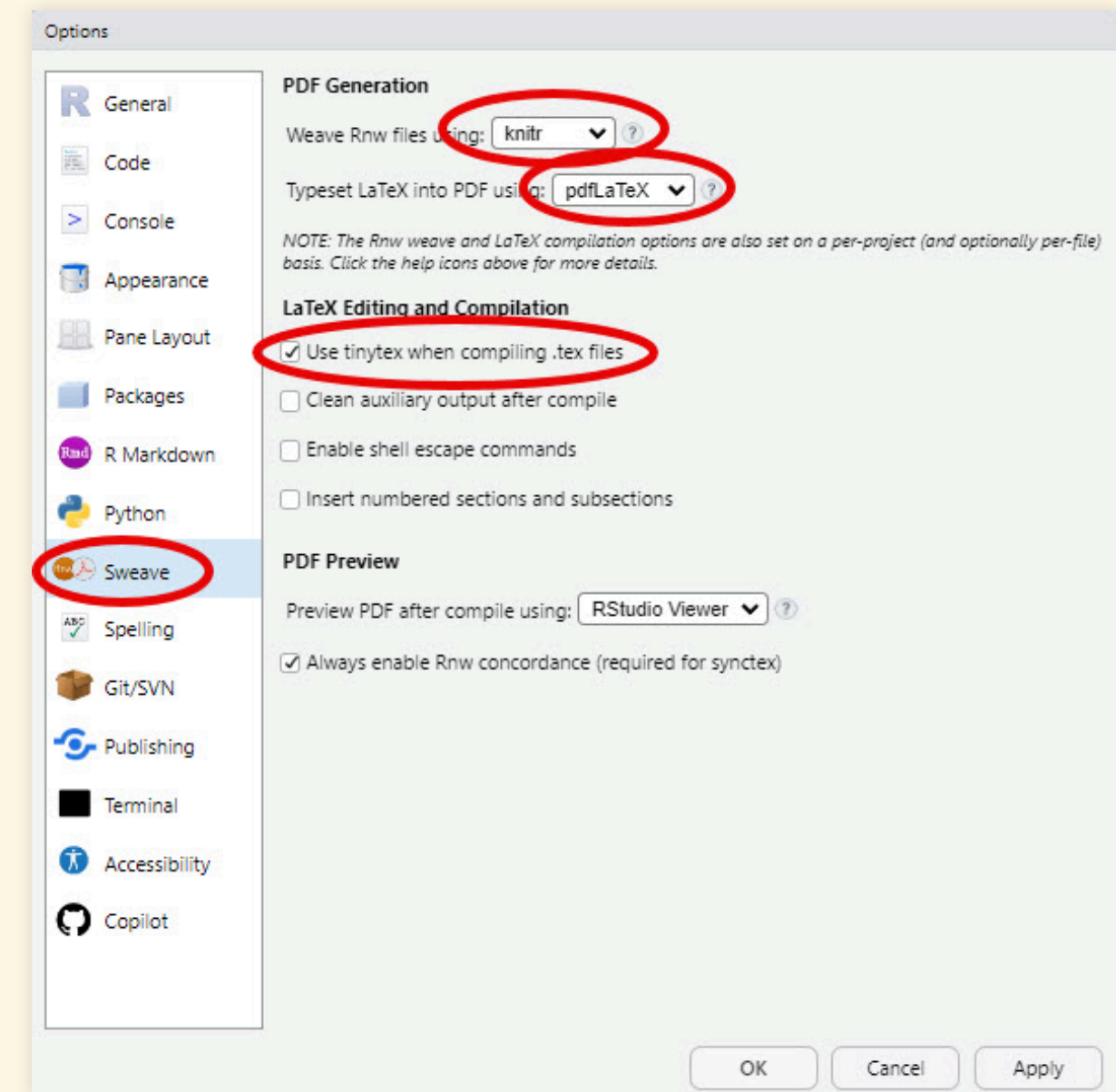
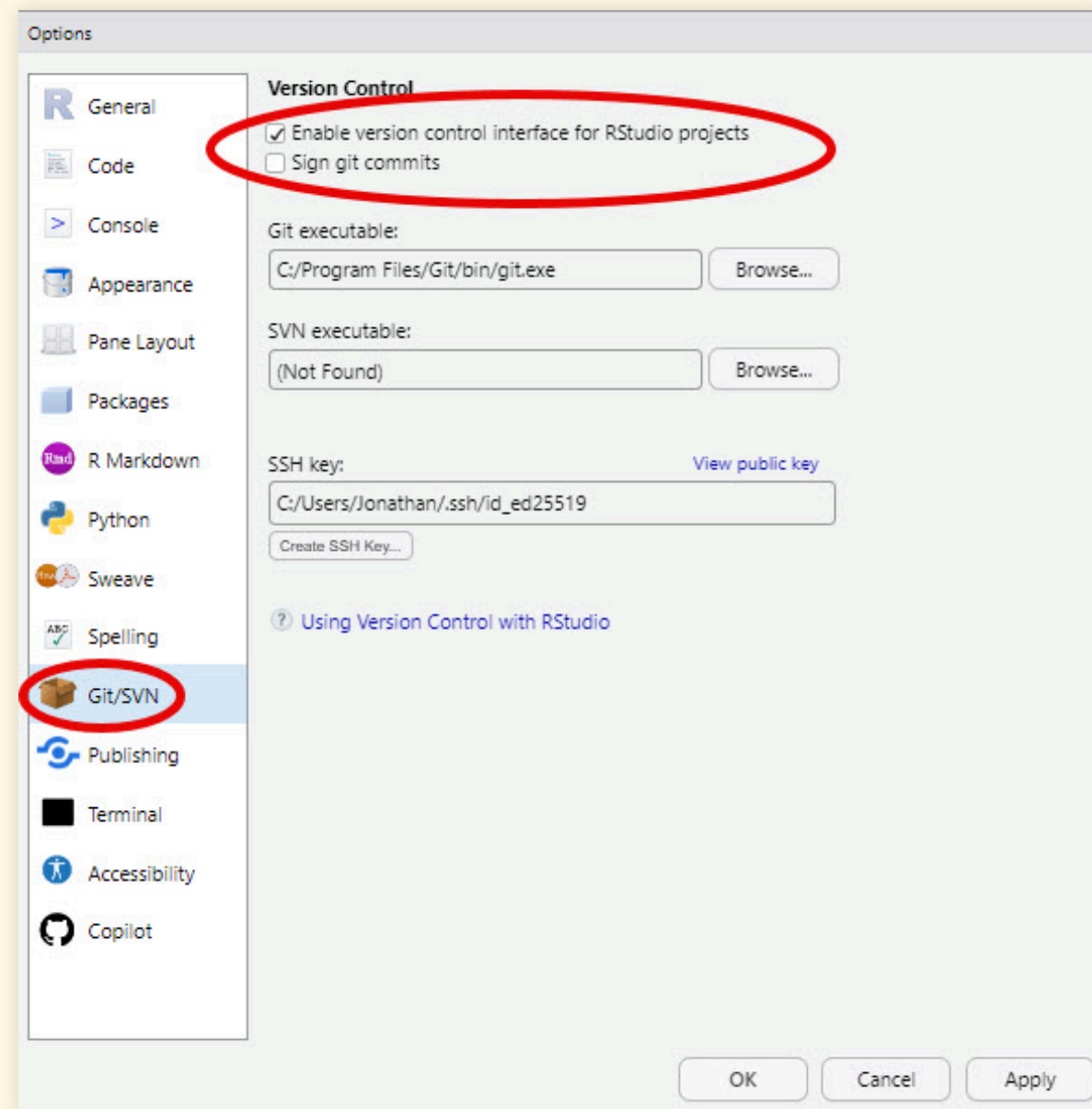


Set Global Options



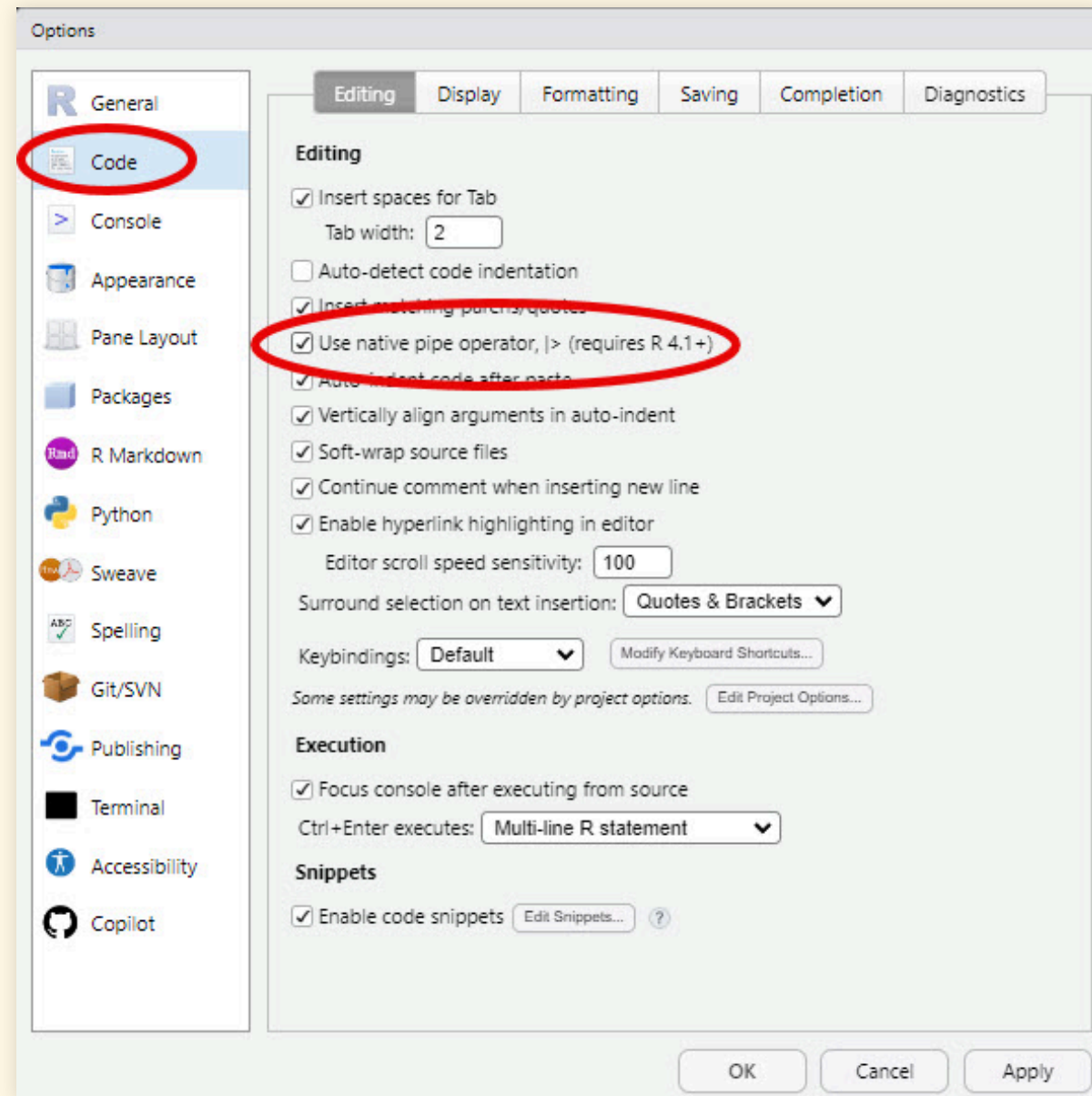
Set Global Options

- Configure `git` for revision control
- Configure `knitr` for generating reproducible research documents



Set Global Options

- Configure code formatting to use “native pipe”



R Packages

R Packages

- There are thousands of free software packages that extend the basic capabilities of R.
- Packages are hosted on a global network of repositories called the Comprehensive R Archive Network (CRAN)
- In this course, we will use a suite of packages called the `tidyverse`.
- You can install packages into R in two ways:
 - Interactively, using the “Packages” tab in RStudio
 - Automatically, using R scripts, with the `install.packages()` function

Installing Packages for This Course

The screenshot shows the RStudio interface. The console on the left displays the R version 4.4.2 (2024-10-31 ucrt) and the 'Pile of Leaves' project. The 'Packages' pane on the right shows a list of installed and available packages. Two red arrows point to the 'Install' button and the 'Packages' tab. The 'Install' button is labeled 'Click "Install"' and the 'Packages' tab is labeled 'Open "packages" pane'.

Environment is empty

Open "packages" pane

Click "Install"

Files Packages Help Viewer Presentation

Install Update

Name	Description	Version
abind	Combine Multidimensional Arrays	1.4-8
acepack	ACE and AVAS for Selecting Multiple Regression Transformations	1.4.2
acs	Download, Manipulate, and Present American Community Survey and Decennial Data from the US Census	2.1.4
addinslist	Discover and Install Useful RStudio Addins	0.5.0
ade4	Analysis of Ecological Data: Exploratory and Euclidean Methods in Environmental Sciences	1.7-22
adehabitatLT	Analysis of Animal Movements	0.3.28
adehabitatMA	Tools to Deal with Raster Maps	0.3.17
AER	Applied Econometrics with R	1.2-14
afex	Analysis of Factorial Experiments	1.4-1
AGD	Analysis of Growth Data	0.45.0
aiRthermo	Atmospheric Thermodynamics and Visualization	1.2.1
akima	Interpolation of Irregularly and Regularly Spaced Data	0.6-3.4
AlignAssign	Align the assignment operators within a highlighted area	0.5.0
alphavantage	Lightweight Interface to the Alpha Vantage API	0.1.3
amen	Additive and Multiplicative Effects Models for Networks and Relational Data	1.4.5
analyzeBehaviorspace	Processes NetLogo BehaviorSpace Output	1.2.0
anytime	Anything to 'POSIXct' or 'Date' Converter	0.3.11
aod	Analysis of Overdispersed Data	1.3.3
apaTables	Create American Psychological Association (APA) Style Tables	2.0.8
ape	Analyses of Phylogenetics and Evolution	5.8-1

Working with GitHub and GitHub Classroom

Working with GitHub and GitHub Classroom

- Following Reproducible Research practices, we will use tools for managing code and documents:
 - `git` is software on your computer that integrates with RStudio to manage files on your computer and keep track of the history of changes you make.
 - **GitHub** is a website that works with `git` to keep a copy of your local files and their history, in the cloud
 - **GitHub** makes it easy to collaborate and share your work
 - **GitHub** also provides safe, reliable storage in case something happens to your computer
 - **GitHub Classroom** is a service of **GitHub** for distributing and turning in computational assignments.

Connecting RStudio with GitHub

- The `credentials` package that you installed helps manage your GitHub login credentials so RStudio can get files from GitHub and save your work to GitHub

Type the following into the RStudio console:

```
library(credentials)  
git_credential_ask()
```

- This will open a browser window where you can log in to GitHub and it will prompt you to connect GitHub to RStudio.

Working with RStudio

Working with RStudio

- RStudio thinks about “projects”:
 - All files for a project are in a directory, and possibly sub-directories
 - RStudio creates a `.Rproj` file and an `.Rproj.user` directory to help it remember what you were doing, and re-open projects where you left off.
 - It also creates
 - `.RData` to keep track of temporary variables and data you were using
 - `.Rhistory` to keep track of the R commands you used
- RStudio can use `git` to manage projects
 - Create a project from a project on GitHub
 - Update the GitHub project with your local changes

Working with RStudio and GitHub

- RStudio can import projects from GitHub.
- We will import a project from GitHub Classroom
- Go to <https://classroom.github.com/a/SdzYvx3b>
 - You can find the link at the homework assignment for today on the course website <https://ees5891.jgilligan.org>



Accepting an Assignment

Join the classroom:

gilligan-ees-5891-classroom-2025

To join the GitHub Classroom for this course, please select yourself from the list below to associate your GitHub account with your school's identifier (i.e., your name, ID, or email).

Can't find your name? [Skip to the next step →](#)

Identifiers	
Blair, Gavin Keith	>
Claiborne, Lily	>
Geist, Brandt	>

gilligan-ees-5891-classroom-2025

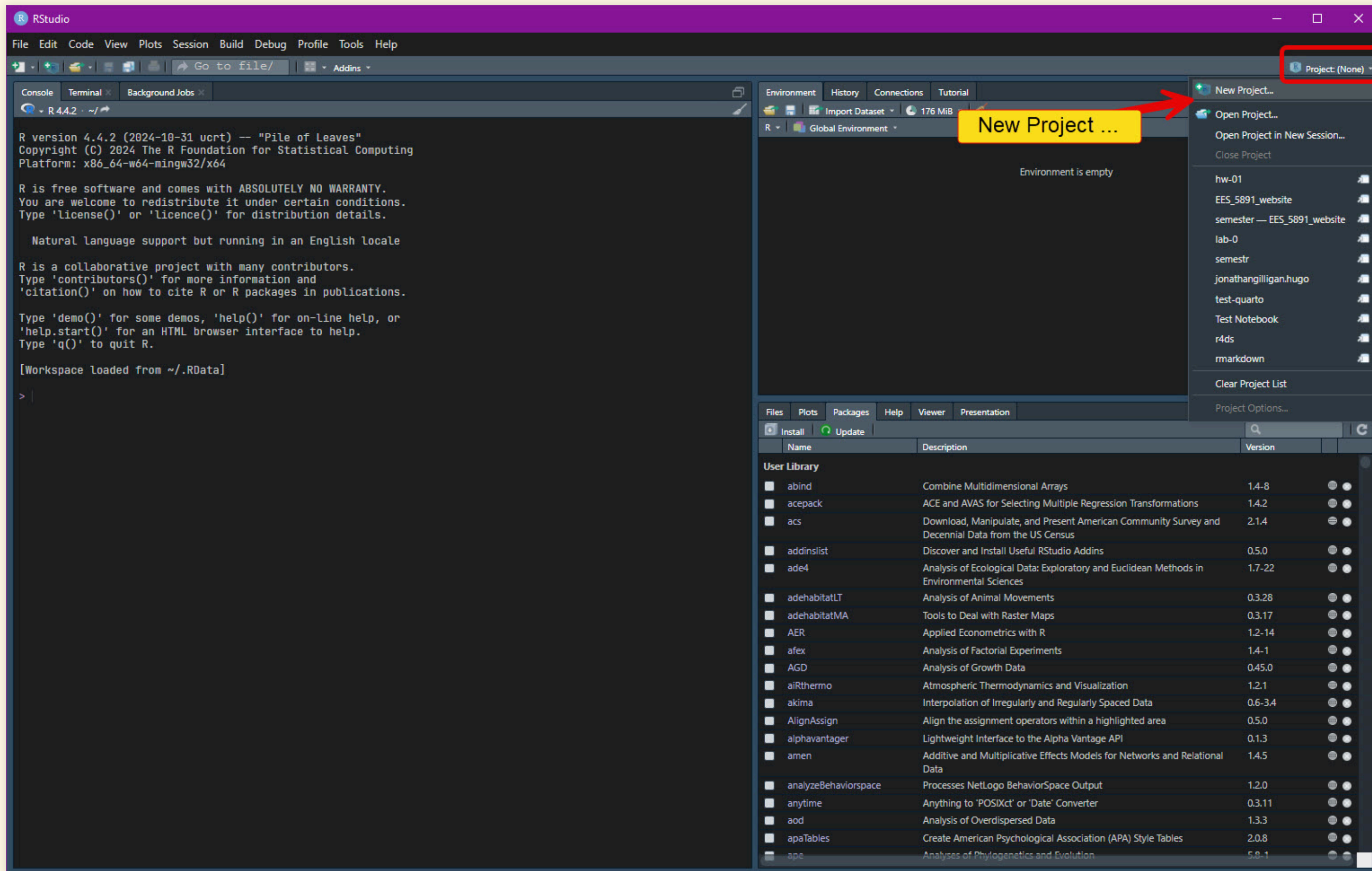
Accept the assignment —

Homework #1: Introduction to R and RStudio

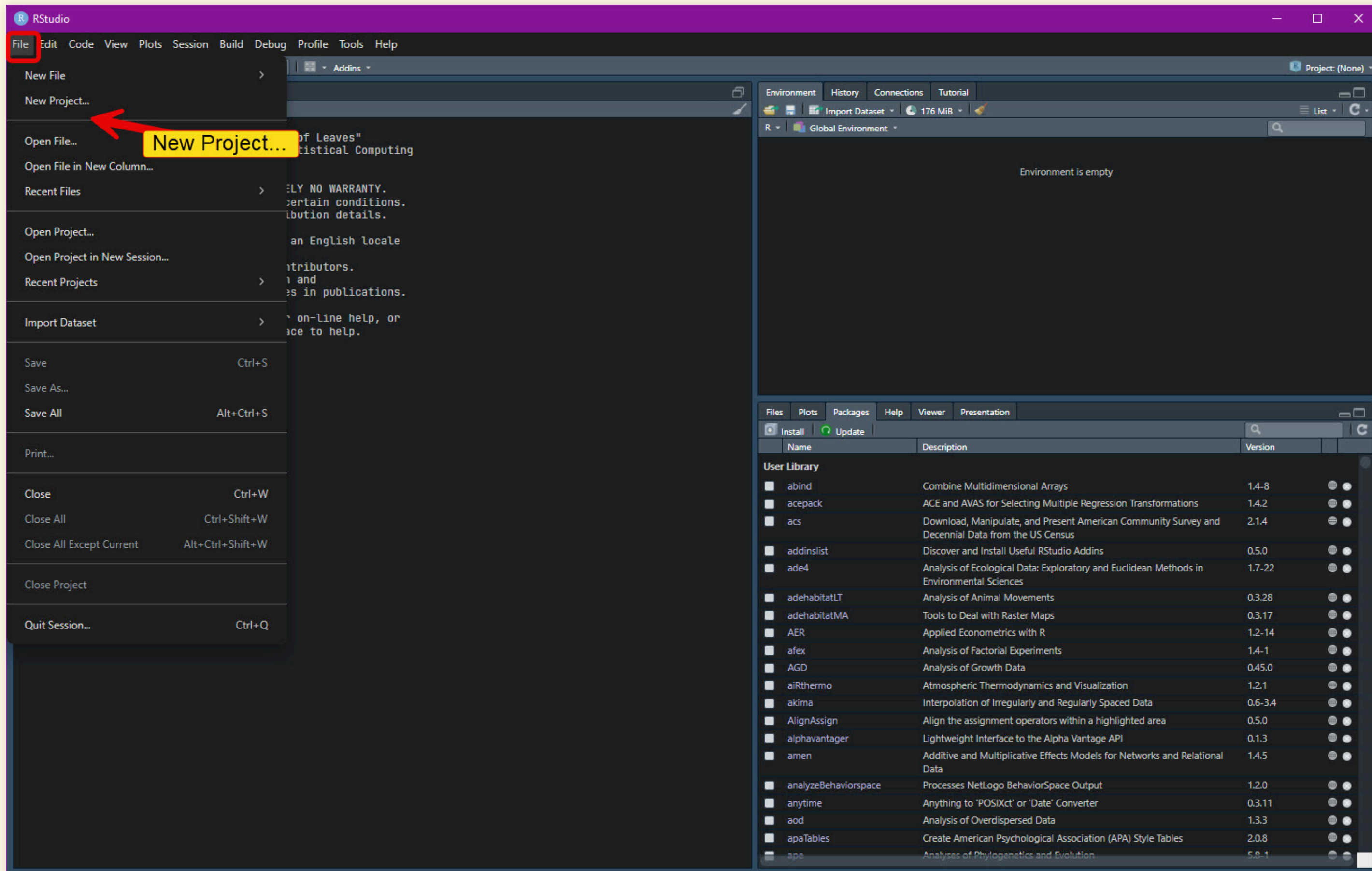
Once you accept this assignment, you will be granted access to the `hw-01-intro-r-studio-jonathan-g` repository in the [gilligan-ees-5891-2025](#) organization on GitHub.

Accept this assignment

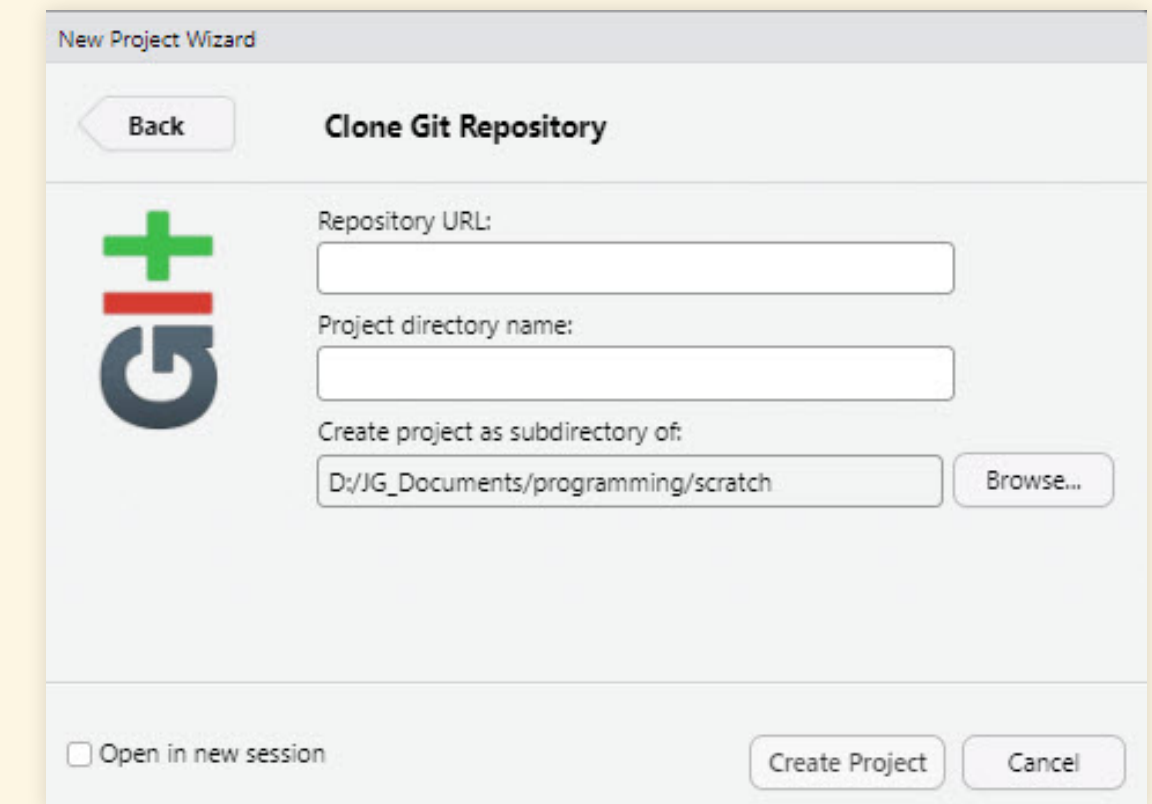
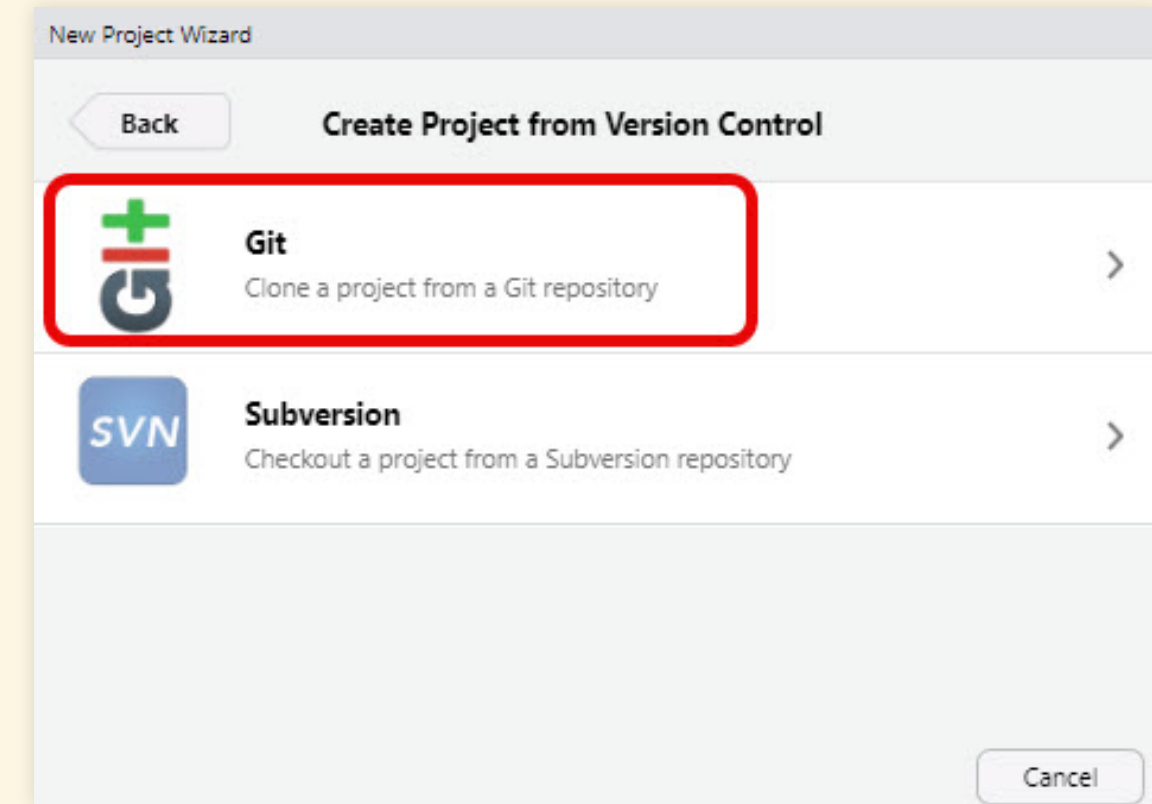
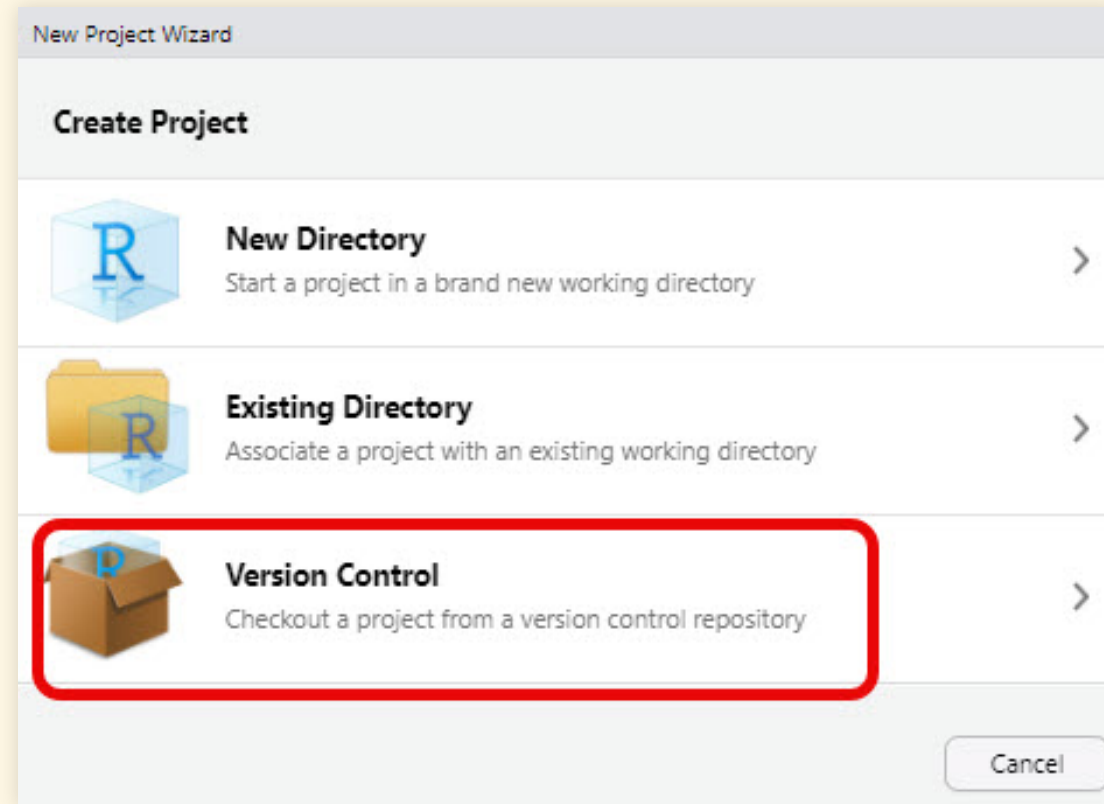
Creating an RStudio Project



Creating an RStudio Project



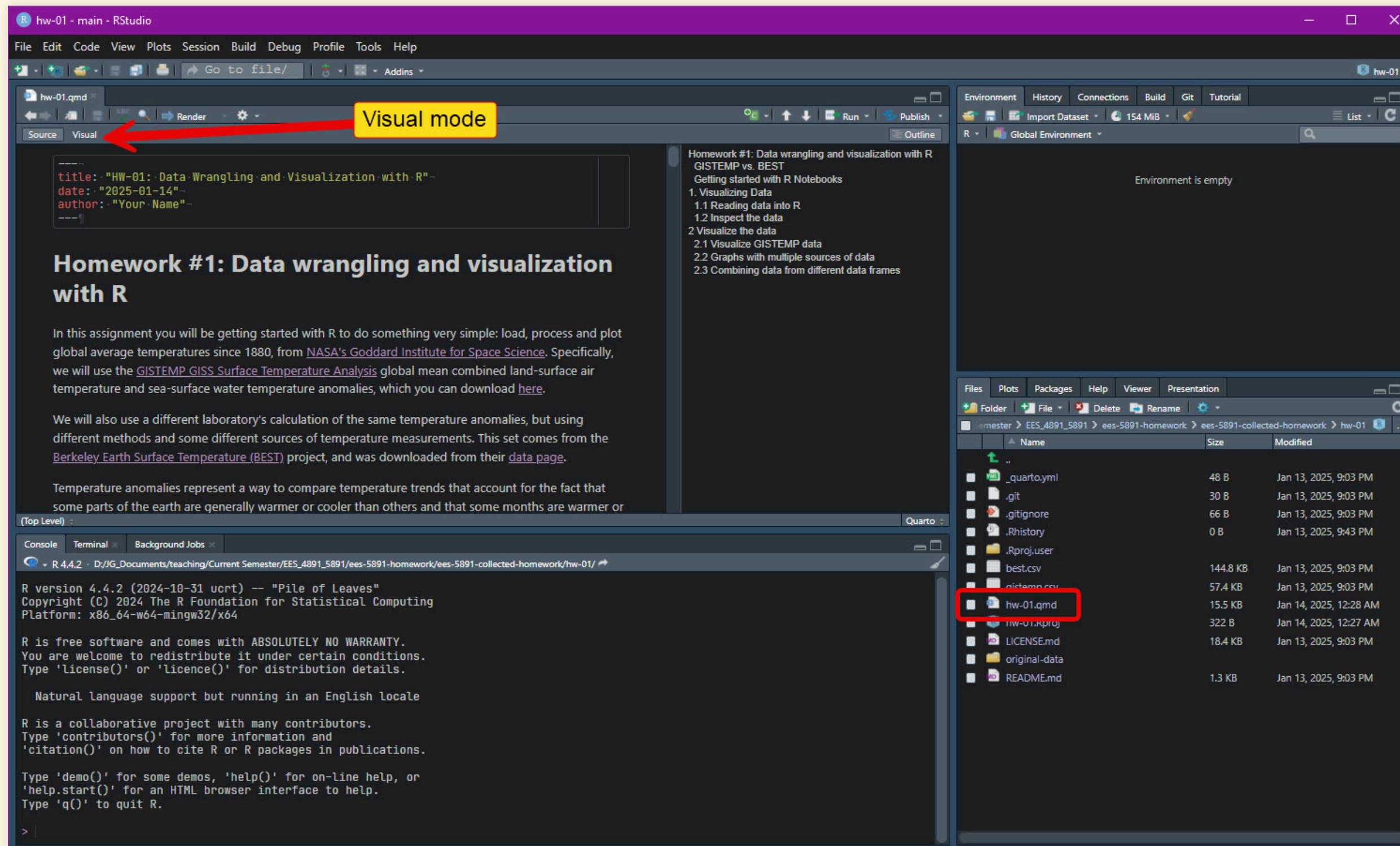
Creating an RStudio Project from GitHub



Working with Data in R

Working with Data in R

- Open the file `hw-01.qmd`
- Set the editor pane to “Visual” mode



Quarto Notebooks

- Notebooks mix text with R code

1.1 Reading data into R

Worked example Load the `gistemp.csv` file with the `read_csv()` function and assign it to a variable `gistemp`.

```
{r}-
gistemp = read_csv("gistemp.csv")
```

Question 1.1 Load `best.csv` and assign it to a variable `best`

```
{r}-

```

- They can also use Python and other languages

- Text and graphics output:

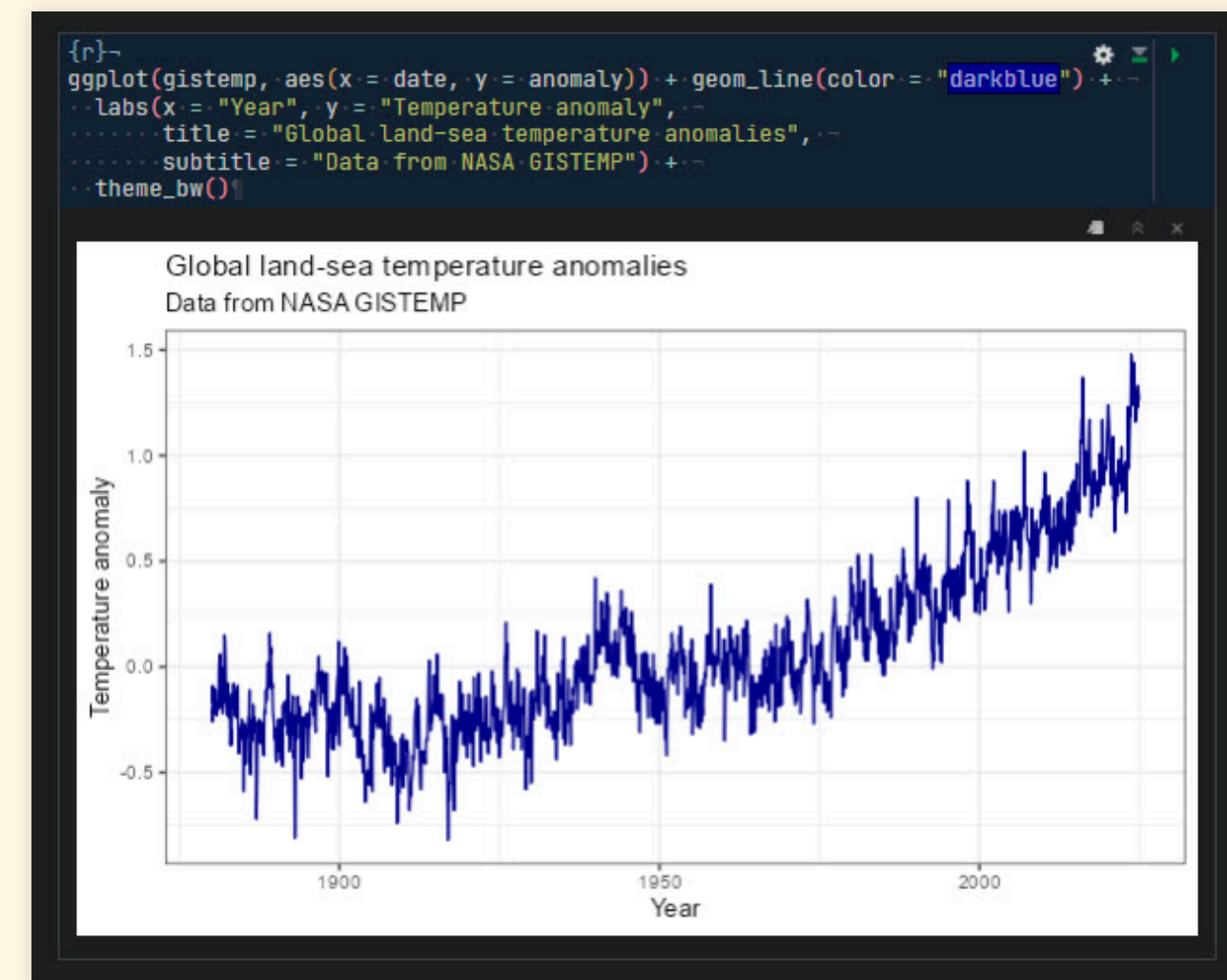
• Use the `head()` function to view the first few rows of a data frame or tibble: `head(gistemp)` will show the default of 6 rows, or you can show more or fewer rows: `head(gistemp, 10)`.

```
{r}-
head(gistemp, 10)
```

A tibble: 10 × 3

date <date>	anomaly <dbl>	annual_avg <dbl>
1880-01-15	-0.20	NA
1880-02-15	-0.26	NA
1880-03-15	-0.09	NA
1880-04-15	-0.17	NA
1880-05-15	-0.10	NA
1880-06-15	-0.22	NA
1880-07-15	-0.21	-0.1808333
1880-08-15	-0.11	-0.1808333
1880-09-15	-0.16	-0.1725000
1880-10-15	-0.23	-0.1633333

1-10 of 10 rows



Walk Through the Notebook

Grammar of Graphics

Grammar of Graphics

- The heart of data visualization in the Tidyverse is the `ggplot2` package.
 - `gg` from Leland Wilkinson, *The Grammar of Graphics*
- A graph has a grammar, which can be expressed in terms of 7 layers

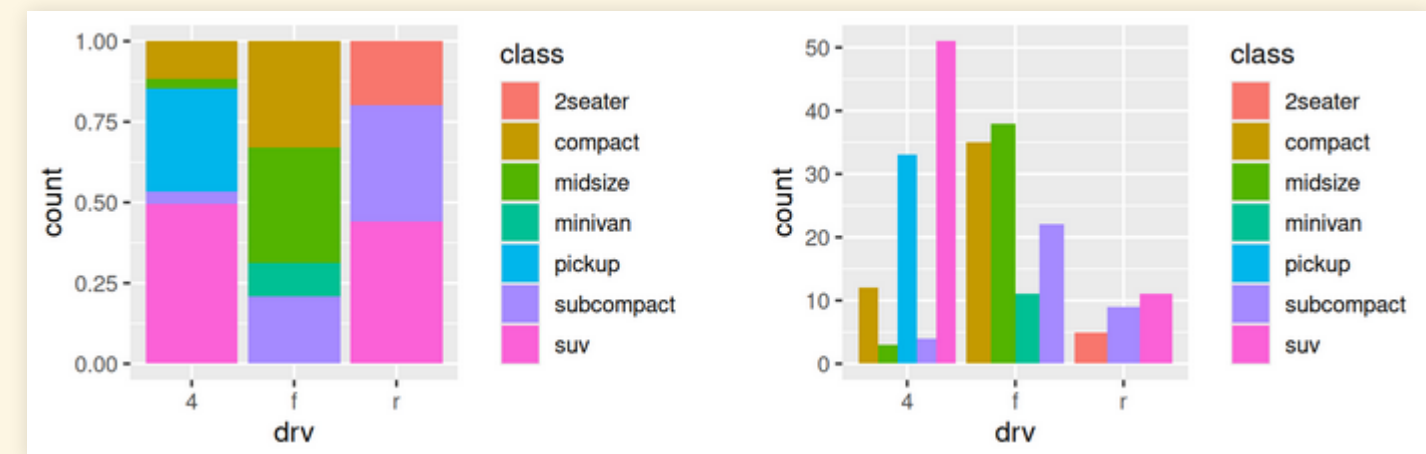
1. Data

2. Geometries: How to draw the data (Lines, points, etc.)

3. Aesthetics: How to mapping different dimensions of the data to visual attributes (x & y coordinates, colors, shape, etc.)

4. Statistical transformations: histograms, smoothing, box-plots, etc.

5. Position adjustments: e.g., side-by-side versus stacked bar charts



6. Coordinate systems: Cartesian vs. polar, pie charts, etc.

7. Facets: Breaking a graph into sub-plots.

Grammar of Graphics in R

- **Data** is represented as a `data.frame`, or a special kind of data frame called a `tibble` (for “Tidy Table”)

- A `data.frame` is like a spreadsheet.
- Each column represents a different variable (year, month, temperature, rainfall, humidity, etc.)
- Each row represents a different measurement of all the variables.
- Load the global temperature record:

```
library(tidyverse)
gistemp = read_csv("gistemp.csv")
```

- To start a graph, specify the data and aesthetics:

```
ggplot(gistemp, aes(x = date, y = anomaly))
```

- Next, we add geometries

```
ggplot(gistemp, aes(x = date, y = anomaly)) +  
  geom_line()
```

- Label coordinates

```
ggplot(gistemp, aes(x = date, y = anomaly)) +  
  geom_line() +  
  labs(x = "Year", y = "Temp. Anomaly",  
        title = "NASA GISS Global Temperatures")
```

