Fostering Data Literacy through Paper Replication & Reproducible Workflows

Thanicha Ruangmas[a]*

[a]*The First-Year Innovation & Research Experience (FIRE), University of Maryland, College Park, USA*

*ruangmas@umd.edu

Fostering Data Literacy through Paper Replication & Reproducible Workflows

This paper presents a curriculum designed to cultivate data literacy among first-year undergraduates through the replication of a published economics paper and the application of reproducible research workflows. Unlike existing methods that necessitate prior econometrics knowledge, this curriculum is accessible to students without prerequisites, providing an early introduction to economics research. The curriculum integrates basic R programming instruction via an online platform, which enables students to replicate a notable economics study. By incorporating tools such as GitHub and Posit Cloud, the curriculum emphasizes the importance of reproducibility in research. The structure and implementation of the course are detailed, offering insights and practical guidance for educators aiming to achieve similar educational objectives.

Keywords: data literacy; economic education; reproducibility

Subject classification codes: A22, C81

1. Introduction

Given the growing demand for empirical skills like causal inference and research design (Angrist and Pischke 2017), data literacy has become a foundational skill that students should acquire early in their college education. Defined as the ability “to read, create, and communicate data as information” (Halliday 2019), data literacy encompasses not only analyzing data but also documenting thought processes and presenting outcomes effectively. Engaging undergraduate students in economics research and teaching reproducible workflows have separately become common practices for building students' data literacy (Hoyt and McGoldrick 2017; Marshall and Underwood 2020). This paper outlines a curriculum for training first-year undergraduates in economics research by replicating a published paper and simultaneously teaching the reproducible research workflow.

Research reproducibility and research replicability have different definitions. Reproducibility refers to the ability to regenerate the exact same results from an original study. A reproducible research workflow showcases all the steps and thought processes behind each step in a study so that the results can be reproduced. In contrast, replicability refers to the ability to generate the same results from different data or methodologies to ensure the robustness of an original study. Table 1 provides definitions of terms frequently used in the context of this article.

This paper responds to a call for such a curriculum in multiple ways. First, the curriculum described in this paper does not rely on prerequisites. Although many papers have documented methods that train undergraduate students in economics research, they require prior econometrics training (Klein 2013; Henderson 2018; Marshall and Underwood 2020; Gitter 2021). There are no guidelines on how to teach undergraduate economics students to conduct research without prerequisites, enabling them to engage in economics research early in their studies.

Second, this paper provides a curriculum for paper replication and reproducible workflow for first-year graduates. The only documentation on training upper-level students to replicate economics papers while employing the reproducible research workflow is by Vilhuber et al. (2022). Similar to why first-year undergraduates are not typically taught economics research methods, there are challenges in teaching students to replicate research papers, including the need for prerequisite training (Vilhuber et al. 2022; Bauer et al. 2023), the demand for time and patience from instructors (Hoffler 2013), and a lack of well-established practices and instructor training (Horton et al. 2022).

As first-year students do not have prior training in R programming and econometrics, the curriculum utilizes an online platform to teach them basic R programming. Subsequently, students apply their newly acquired skills in assignments that guide them to replicate an American Economic Review paper. To instill a habit of ensuring reproducibility, students learn to generate a reproducible workflow using Posit Cloud and Git simultaneously.

This paper is organized as follows: Section 2 describes the course in which the curriculum is taught. Section 3 covers the tools used to teach paper replication. Section 4 discusses the tools used to teach the practice of a reproducible workflow. Section 5 explains how these tools are taught to students in two learning environments. Finally, Section 6 discusses the lessons learned. I hope this paper can serve as a guideline for instructors who wish to achieve similar learning outcomes and overcome the same difficulties.

Table 1: Definitions of Acronyms and Common Terms

Term	Definition
Data literacy	The ability to read, create, and communicate data as information
Replicability	The ability to generate the same results from different data or methodologies to ensure the robustness of an original study
Reproducibility	The ability to regenerate the exact same results from an original study using the same data and code
Git	A version control software system
GitHub	A web platform that allows Git to be run in the cloud
repository	A storage space on GitHub that allows all files and versions from a project to be stored together
ghclass	An R package that allows instructors to distribute assignments to students (Rundel et al. 2020)
Posit Cloud	A cloud-based version of RStudio that allows RStudio to be run on a web browser
R Markdown	A document format that ends with .md which allows the user to create documents that combine code, text, and output, all in a single file
Quarto	An updated version of R Markdown with a .qmd extension which allows for multiple programming languages and more output options
GitHub-Flavored Markdown (GFM)	A document that displays code, text, and output on a GitHub platform

Term	Definition
clone	A Git action that allows the user to make a copy of a Git repository on a local computer or a cloud-based environment such as Posit Cloud
commit and push	A Git action that allows the user to save the current version of their work and update the cloud-based repository with the current local version

2. Course Description

The curriculum described in this paper has been used to teach data literacy skills to students as part of a research course called FIRE Semester 2 in Sustainability Analytics. This is the second course in a three-course sequence as part of the University of Maryland’s First-Year Innovation & Research Experience (FIRE) program, where students are introduced and trained in empirical environmental economics research in order to conduct research in the following semester.

Note that this course is not intended to replace an introductory econometrics course but to allow students from various majors to develop proficiency in R programming to clean and combine multiple data sources into a balanced panel, create tables and figures, and test hypotheses. Students are not expected to understand the underlying statistical theory but are taught to run and interpret ordinary least squares and difference-in-differences models. Table 2 outlines the tools and activities used to teach data literacy from Week 3 to Week 10 of the course.

Table 2: Curriculum for Teaching Paper Replication and Reproducible Workflow in FIRE Semester 2

Week	DataCamp Assignments	Classroom Discussions	Lab Activities
3	Course: Intro to R ● Chapter: Intro to basics	● Introduce the RStudio interface in Posit Cloud	

Week	DataCamp Assignments	Classroom Discussions	Lab Activities
		- *Discuss the importance of reproducibility**	
4	Course: Intro to R - Chapters: Vectors, Matrices, Factors, Dataframes	- Introduce packages	- *Demonstrate how to clone assignments, make changes, save, commit, and push into GitHub**
5	Course: Introduction to the Tidyverse - Chapters: Data wrangling, Data visualization	- Introduce the Quarto interface - Applying the filter function to remove rows with NA values - Using the mutate and ifelse functions together to create conditional variables	- Work on Paper Replication Assignment 1: Understanding the Data
6	Course: Introduction to the Tidyverse - Chapters: Grouping and summarizing, Types of visualizations	- Discuss how to use group_by and summarize functions to find total and average emissions across seasons	- Work on Paper Replication Assignment 2: Make Figure 1
7	Course: Joining data with data.table in R - Chapters: Joining multiple data.tables	- Discuss the differences between cross-sectional, time series, and panel datasets	- Work on Paper Replication Assignment 3: Make Table 1
8		- Discuss Paper Replication Assignment 4: Make Appendix Figure 2A	- Work on Paper Replication Assignment 4: Make Appendix Figure 2A
9	Course: Modeling with data in the tidyverse - Chapters: Introduction to modeling, Modeling with basic regression,	- Discuss how to use regressions to predict vs. to explain	

Week	DataCamp Assignments	Classroom Discussions	Lab Activities
	Modeling with multiple regression		
10		• Discuss the role of dummy variables and fixed effects • Demonstrate how to run and interpret difference-in-differences models	• Work on Paper Replication Assignment 5: Make Table 2

*Course content for teaching the reproducible research workflow is in italics.

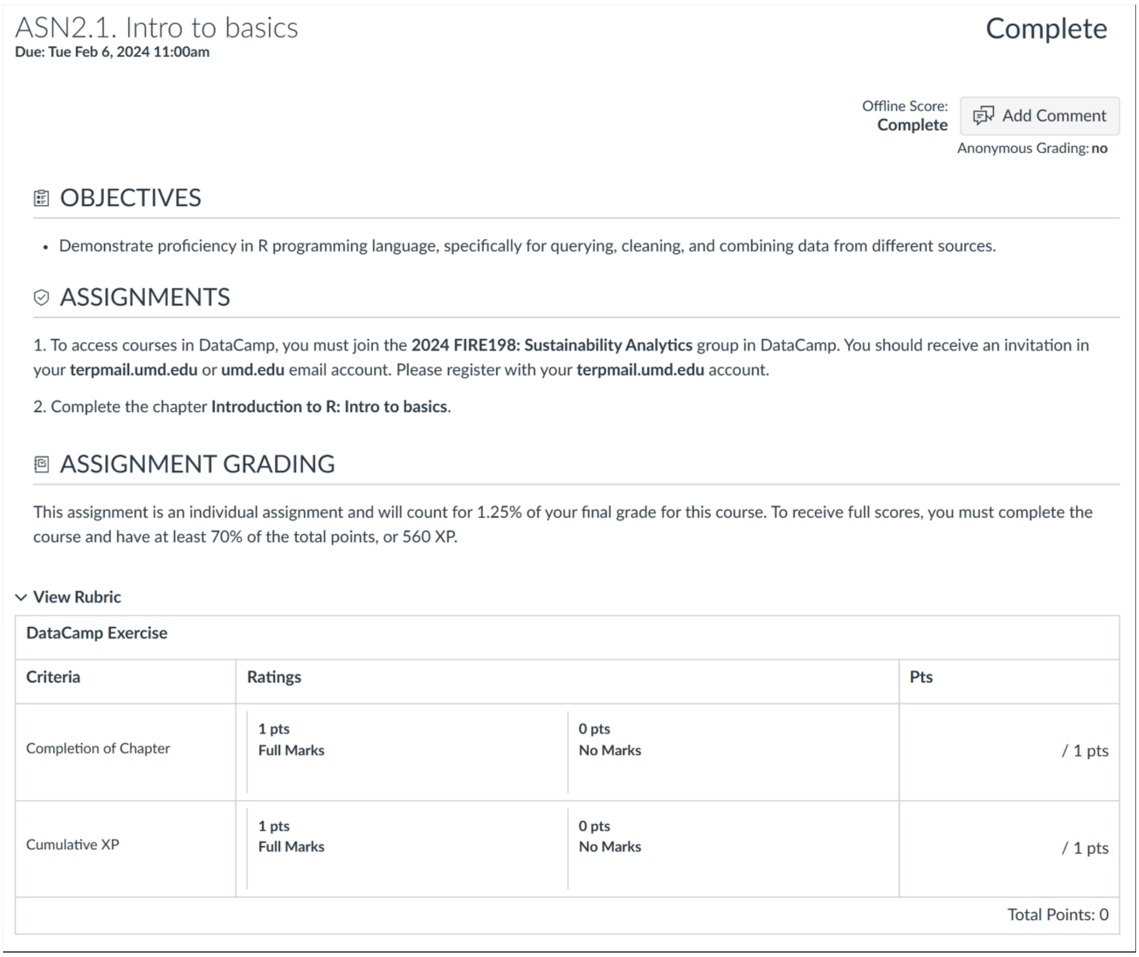
3. Tools for Teaching Paper Replication

This curriculum outsources teaching basic programming syntaxes to an online platform called DataCamp. Afterward, the Paper Replication Assignments allow students to follow the steps of a successful research project. This section discusses how these two tools are used in detail.

3.1. DataCamp

The online instructional platform of choice is DataCamp.com, which has a DataCamp for Universities program that allows instructors to give students access to any programming course in their platform at no cost. The instructor will also be able to see each student's completion status for each course assigned. At the beginning of the semester, the instructor must submit a request to create a classroom. Once the request has been approved, the instructor can invite students as Members of the group. Afterward, the instructor can change each student's DataLab status to Classroom and assign courses or chapters as assignments with deadlines. I recommend that the instructor take this step after students have accepted their invitation; otherwise, the instructor will not be able to see if the student has completed the assignment.

With DataCamp for Universities, the instructor cannot see detailed reports of students' progress. The instructor will only be able to see whether the assignment is in progress or completed by each student before or after the deadline. Additionally, the instructor will be able to see each student's total experience points (XP) from the last 30 days, 90 days, 365 days, or all time. Students gain full XPs within each lesson when they watch a video or complete an exercise correctly. Students lose some points if they take hints or ask for the answer. Each week, students are graded based on their completion status of the week's DataCamp assignments and their cumulative XP for the last 30 or 60 days. Cumulation XP has to be taken into account when grading as it prevents students from asking for the answers through every exercise. An example of instructions for a DataCamp assignment and its rubric on a learning management platform is shown in Figure 1. Table 2 outlines which DataCamp courses and chapters are assigned to students each week.



ASN2.1. Intro to basics

Due: Tue Feb 6, 2024 11:00am

Complete

Offline Score: **Complete**

Anonymous Grading: **no**

OBJECTIVES

- Demonstrate proficiency in R programming language, specifically for querying, cleaning, and combining data from different sources.

ASSIGNMENTS

1. To access courses in DataCamp, you must join the **2024 FIRE198: Sustainability Analytics** group in DataCamp. You should receive an invitation in your **terpmail.umd.edu** or **umd.edu** email account. Please register with your **terpmail.umd.edu** account.

2. Complete the chapter **Introduction to R: Intro to basics**.

ASSIGNMENT GRADING

This assignment is an individual assignment and will count for 1.25% of your final grade for this course. To receive full scores, you must complete the course and have at least 70% of the total points, or 560 XP.

DataCamp Exercise

Criteria	**Ratings**		**Pts**
Completion of Chapter	**1 pts** **Full Marks**	**0 pts** **No Marks**	/ 1 pts
Cumulative XP	**1 pts** **Full Marks**	**0 pts** **No Marks**	/ 1 pts
			Total Points: 0

Figure 1: DataCamp Assignment Instructions for Students

3.2. Paper Replication Assignments

Although educators have reviewed their experiences of teaching upper-level economics students to replicate published studies with training beforehand (Hoffler 2013; Vilhuber et al. 2022), I am teaching students the skills of R programming and basic causal inference through replicating a published study. Each paper replication assignment aims to replicate a specific figure or table in Deschenes et al. (2017) by using R programming functions taught in an assigned DataCamp course the week before. Figure 2 compares the four results in Deschenes et al. (2017) and the end result from each paper replication assignment. Through this process, students understand the relevance and practicality of each R programming function. Students learn to interpret results by generating and reading about them in the original paper. By creating step-by-step exercises for students to follow and using Cognitive Apprenticeship methods, I can avoid creating frustration among students towards the papers' authors when they cannot replicate, as documented in Hoffler (2013). All Paper Replication Assignments can be found on this website. Cognitive Apprenticeship methods are discussed in section 5.2. of this paper.

While other instructors have allowed upper-level students to choose the paper that they would like to replicate (Hoffler 2013), I have chosen "*Defensive Investments and the Demand for Air Quality*" by Deschenes et al. (2017) for students to replicate for four main reasons. First, this seminal environmental economics paper fits FIRE Sustainability Analytics' research focus. Second, part of this paper utilizes data from the Environmental Protection Agency's Clean Air Markets Program Data (EPA CAMD), which is publicly available. The students are empowered by seeing how their newly acquired data-cleaning skills can be used to transform a publicly available dataset to the format used to generate the same conclusions in a renowned study. Third, the *American Economic Review* also requires the authors to share the cleaned data and code from the study. This allowed me, the instructor, to compare the data set we cleaned from EPA CAMD with the authors' data. With the regression code available, students can run the same regressions as the paper, ensuring their results are as close to the published paper as possible. Lastly, the paper uses a triple differences estimator derived from a difference-in-differences estimator, one of the easiest causal inference models applicable to secondary data to understand.

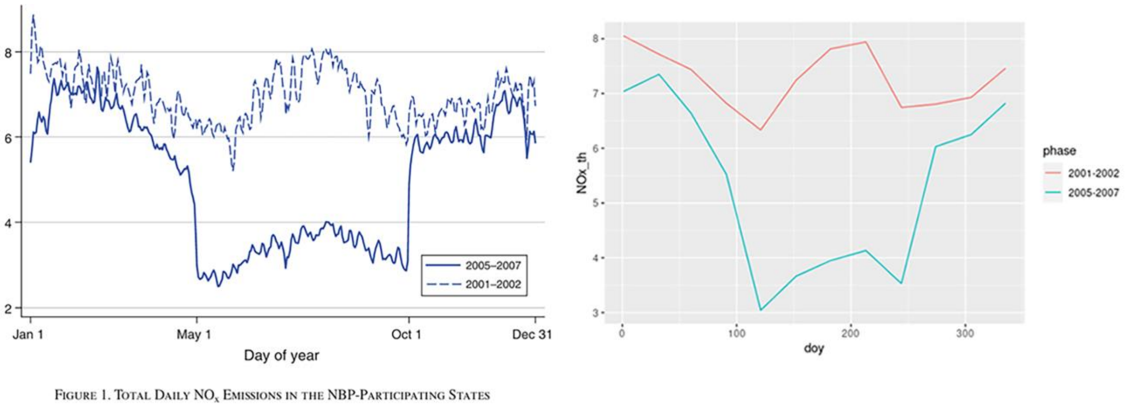


FIGURE 1. TOTAL DAILY NO_x EMISSIONS IN THE NBP-PARTICIPATING STATES

Figure 2A: A Comparison of Figure 1 in Deschenes et al. (2017)[1] on the Left and the Paper Replication Assignment on the Right

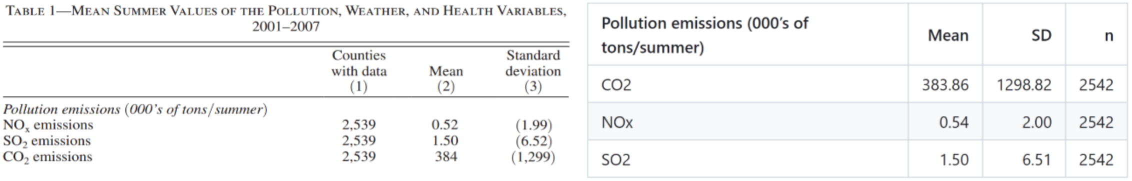
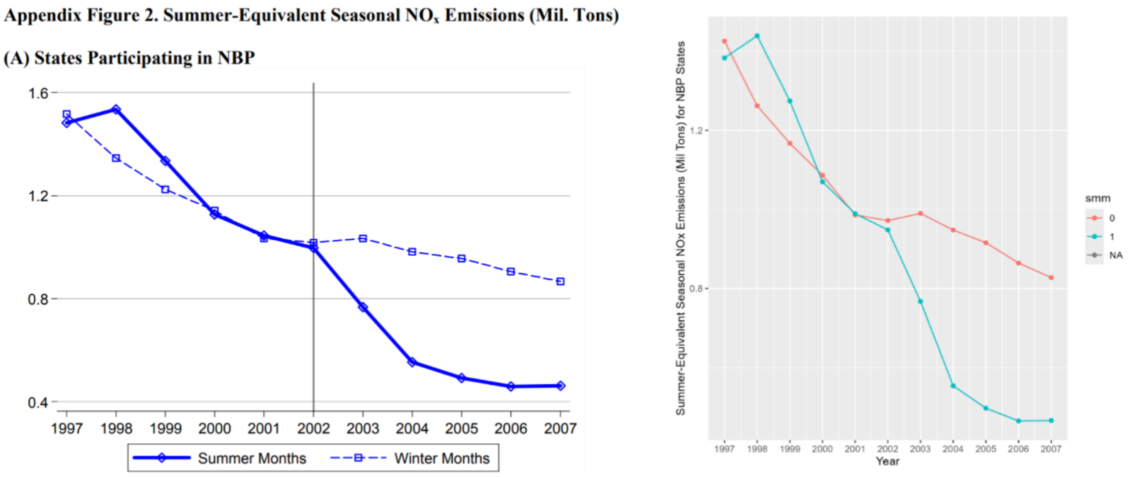


TABLE 1—MEAN SUMMER VALUES OF THE POLLUTION, WEATHER, AND HEALTH VARIABLES, 2001–2007

	Counties with data (1)	Mean (2)	Standard deviation (3)
Pollution emissions (000's of tons/summer)			
NO_x emissions	2,539	0.52	(1.99)
SO_2 emissions	2,539	1.50	(6.52)
CO_2 emissions	2,539	384	(1,299)

Pollution emissions (000's of tons/summer)	Mean	SD	n
CO2	383.86	1298.82	2542
NOx	0.54	2.00	2542
SO2	1.50	6.51	2542

Figure 2B: A Comparison of Table 1 from Deschenes et al. (2017)[1] on the Left and the Paper Replication Assignment on the Right



Appendix Figure 2. Summer-Equivalent Seasonal NO_x Emissions (Mil. Tons)

(A) States Participating in NBP

[1] [Copyright American Economic Association; reproduced with permission of the *American Economic Review*].

Figure 2C: A Comparison of Appendix Figure 2A from Deschenes et al. (2017)[1] on the Left and the Paper Replication Assignment on the Right

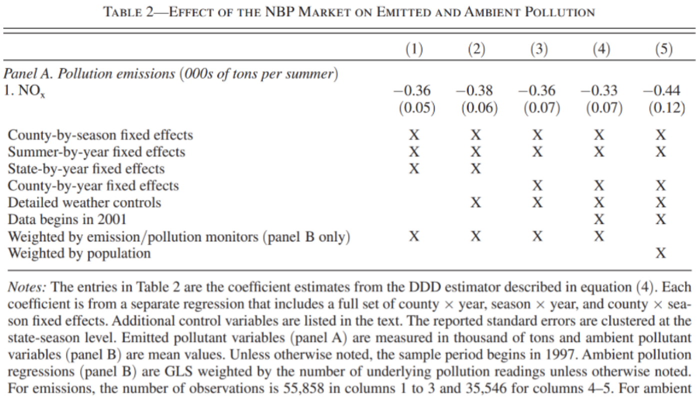
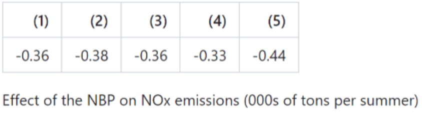


Table 2—Effect of the NBP Market on Emitted and Ambient Pollution

	(1)	(2)	(3)	(4)	(5)
Panel A. Pollution emissions (000s of tons per summer)					
1. NO_x	−0.36 (0.05)	−0.38 (0.06)	−0.36 (0.07)	−0.33 (0.07)	−0.44 (0.12)
County-by-season fixed effects	X	X	X	X	X
Summer-by-year fixed effects	X	X	X	X	X
State-by-year fixed effects	X	X			
County-by-year fixed effects			X	X	X
Detailed weather controls		X	X	X	X
Data begins in 2001				X	X
Weighted by emission/pollution monitors (panel B only)	X	X	X	X	
Weighted by population					X

Notes: The entries in Table 2 are the coefficient estimates from the DDD estimator described in equation (4). Each coefficient is from a separate regression that includes a full set of county × year, season × year, and county × season fixed effects. Additional control variables are listed in the text. The reported standard errors are clustered at the state-season level. Emitted pollutant variables (panel A) are measured in thousand of tons and ambient pollutant variables (panel B) are mean values. Unless otherwise noted, the sample period begins in 1997. Ambient pollution regressions (panel B) are GLS weighted by the number of underlying pollution readings unless otherwise noted. For emissions, the number of observations is 55,858 in columns 1 to 3 and 35,546 for columns 4–5. For ambient



(1)	(2)	(3)	(4)	(5)
-0.36	-0.38	-0.36	-0.33	-0.44

Effect of the NBP on NOx emissions (000s of tons per summer)

Figure 2D: A Comparison of Table 2 from Deschenes et al. (2017)[1] on the Left and the Paper Replication Assignment on the Right

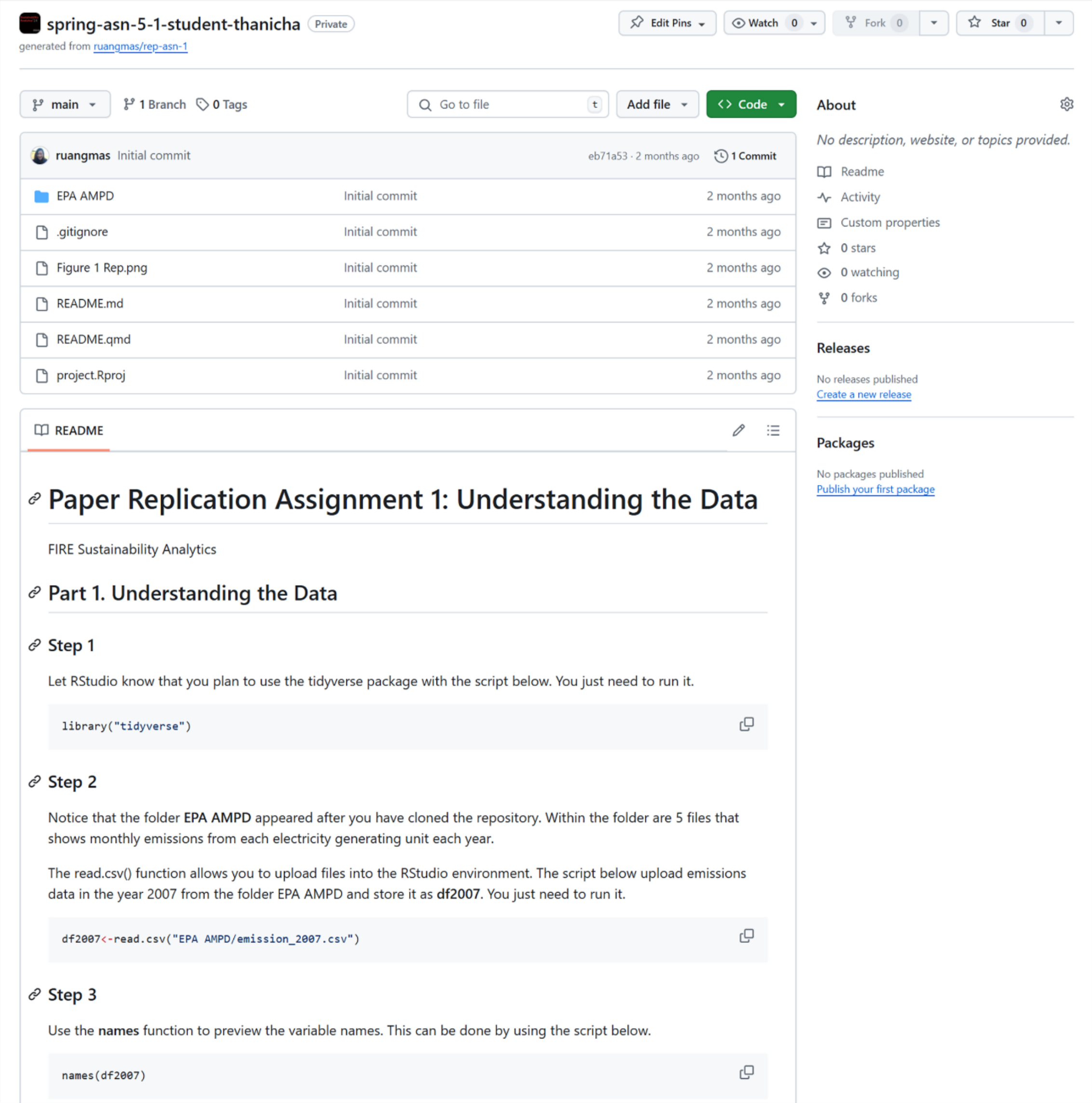
4. Tools for Teaching the Reproducible Workflow

In computational sciences, reproducible research begins with making the data and code accessible (Peng 2011). It extends to ensuring that the entire workflow and results can be consistently regenerated over time, from the development phase through publication and many years afterward (Peikert and Brandmaier 2021). The curriculum outlined in this paper supports dynamic report generation, enabling statistical results to be directly generated from the code and data (Peikert and Brandmaier 2021). It also emphasizes version control, allowing each iteration of work to be recorded as snapshots in time (Peikert and Brandmaier 2021). This section reviews the tools for implementing a reproducible workflow first described by Centinkaya and Rundel (2018) and explains how they are being applied to this course.

4.1. Git

Git is a powerful tool to ensure reproducibility as it is a version control system that allows changes to be tracked over time (Beckman et al. 2021). At the same time, it is an essential skill in the tech sector (Beckman et al. 2021) and a very marketable skill that can distinguish an economics student from their peers. When the Git process is taught from the inception of a student’s data computation training, students can get into the habit of creating a reproducible workflow.

To integrate the Git process into the course, each paper replication assignment is distributed to each student via a GitHub repository. To comply with the Family Educational Rights and Privacy Act (FERPA), each repository is created per assignment per student, as recommended by Centinkaya and Rundel (2018). This allows students not to see other students’ answers once they have completed the assignment. Creating copies of GitHub repositories for students can easily be automated using the ghclass package (Rundel et al. 2020). Figure 3 shows an example of a GitHub repository created per assignment per student. The repository includes a template of the assignment in README.qmd and the required data sets to complete the assignment. Figure 4 shows assignment instructions given to each student via the course’s learning management system.



EPA AMPD	Initial commit	2 months ago
.gitignore	Initial commit	2 months ago
Figure 1 Rep.png	Initial commit	2 months ago
README.md	Initial commit	2 months ago
README.qmd	Initial commit	2 months ago
project.Rproj	Initial commit	2 months ago

No description, website, or topics provided.

Paper Replication Assignment 1: Understanding the Data

FIRE Sustainability Analytics

Part 1. Understanding the Data

Step 1

Let RStudio know that you plan to use the tidyverse package with the script below. You just need to run it.

```
library("tidyverse")
```

Step 2

Notice that the folder **EPA AMPD** appeared after you have cloned the repository. Within the folder are 5 files that shows monthly emissions from each electricity generating unit each year.

The read.csv() function allows you to upload files into the RStudio environment. The script below upload emissions data in the year 2007 from the folder EPA AMPD and store it as **df2007**. You just need to run it.

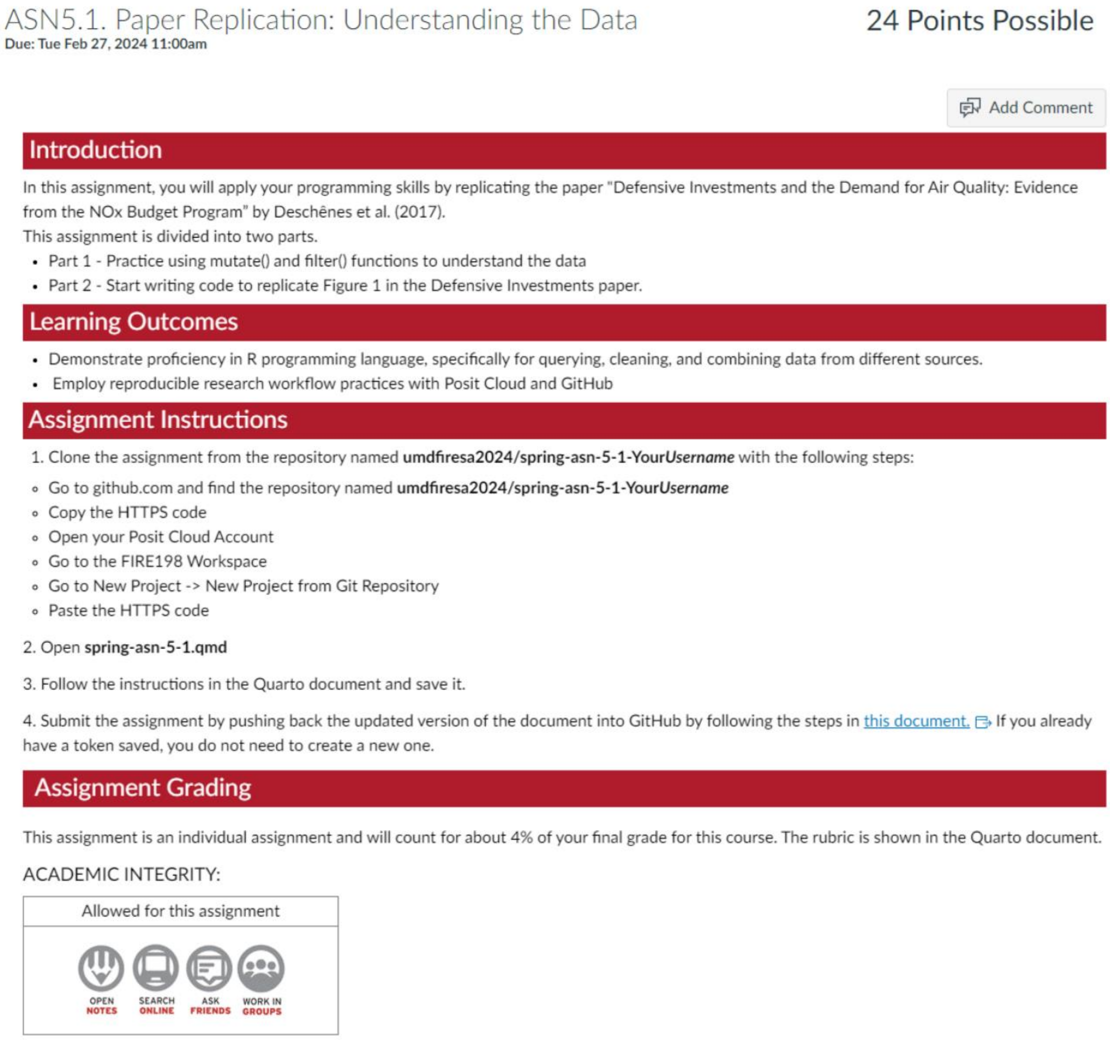
```
df2007<-read.csv("EPA AMPD/emission_2007.csv")
```

Step 3

Use the **names** function to preview the variable names. This can be done by using the script below.

```
names(df2007)
```

Figure 3: Example of a Paper Replication Assignment in a Student-Specific GitHub Repository



ASN5.1. Paper Replication: Understanding the Data

Due: Tue Feb 27, 2024 11:00am

24 Points Possible

Introduction

In this assignment, you will apply your programming skills by replicating the paper "Defensive Investments and the Demand for Air Quality: Evidence from the NOx Budget Program" by Deschênes et al. (2017).

This assignment is divided into two parts.

- Part 1 - Practice using mutate() and filter() functions to understand the data
- Part 2 - Start writing code to replicate Figure 1 in the Defensive Investments paper.

Learning Outcomes

- Demonstrate proficiency in R programming language, specifically for querying, cleaning, and combining data from different sources.
- Employ reproducible research workflow practices with Posit Cloud and GitHub

Assignment Instructions

1. Clone the assignment from the repository named **umdfiresa2024/spring-asn-5-1-Your*Username*** with the following steps:

- Go to github.com and find the repository named **umdfiresa2024/spring-asn-5-1-Your*Username***
- Copy the HTTPS code
- Open your Posit Cloud Account
- Go to the FIRE198 Workspace
- Go to New Project -> New Project from Git Repository
- Paste the HTTPS code

2. Open **spring-asn-5-1.qmd**

3. Follow the instructions in the Quarto document and save it.

4. Submit the assignment by pushing back the updated version of the document into GitHub by following the steps in this document. If you already have a token saved, you do not need to create a new one.

Assignment Grading

This assignment is an individual assignment and will count for about 4% of your final grade for this course. The rubric is shown in the Quarto document.

ACADEMIC INTEGRITY:

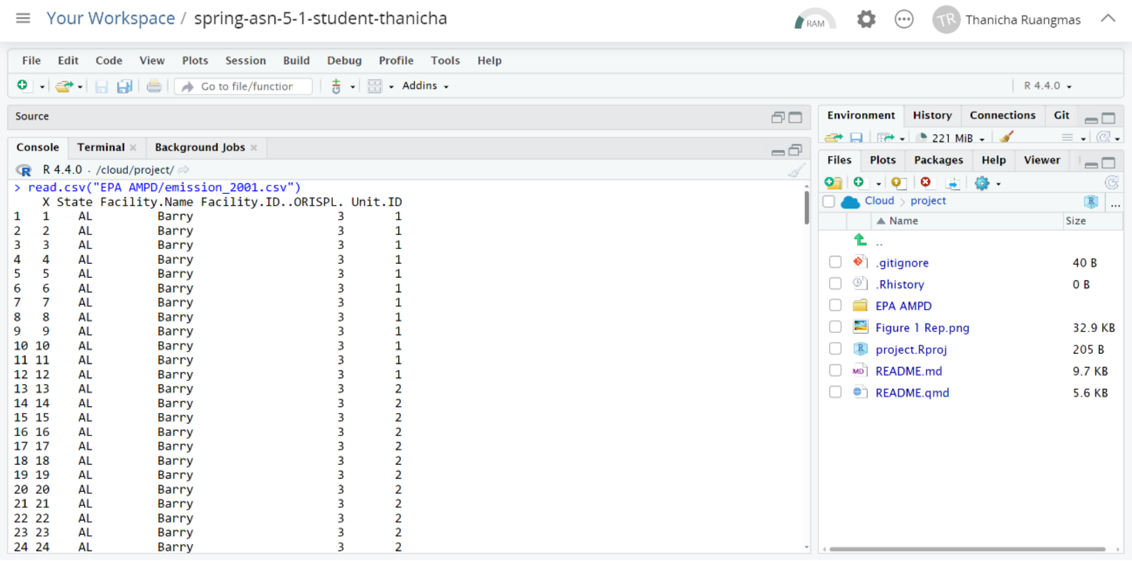
Allowed for this assignment

Figure 4: Example of a Canvas Assignment that Refers Students to Their GitHub Repository

4.2. Posit Cloud

Students use the Posit Cloud computing environment to apply their skills to replicate results from Deschenes et al. (2017). The use of Posit Cloud, the web-based version of RStudio recommended by Centinkaya and Rundel (2018), has allowed students to learn the RStudio integrated development environment (IDE) without having to install multiple programs into their computer or owning a powerful personal computer. Posit Cloud also overcomes installation problems due to different operating systems and versions of RStudio.

Figure 5 shows a screenshot of RStudio within the web-based Posit Cloud environment after an assignment from a GitHub repository has been cloned. Posit Cloud and newer versions of RStudio already have Git embedded in their interface, on the top left tab in the figure, so students can use Git without learning another programming language. Students can refer to a data set by only referring to the folder name within the GitHub repository and the file name and not having to set up a working directory or identify the folder location within their computer.



```
> read.csv("EPA AMPD/emission_2001.csv")
   X State Facility.Name Facility.ID..ORISPL. Unit.ID
1  1    AL         Barry                    3       1
2  2    AL         Barry                    3       1
3  3    AL         Barry                    3       1
4  4    AL         Barry                    3       1
5  5    AL         Barry                    3       1
6  6    AL         Barry                    3       1
7  7    AL         Barry                    3       1
8  8    AL         Barry                    3       1
9  9    AL         Barry                    3       1
10 10   AL         Barry                    3       1
11 11   AL         Barry                    3       1
12 12   AL         Barry                    3       1
13 13   AL         Barry                    3       2
14 14   AL         Barry                    3       2
15 15   AL         Barry                    3       2
16 16   AL         Barry                    3       2
17 17   AL         Barry                    3       2
18 18   AL         Barry                    3       2
19 19   AL         Barry                    3       2
20 20   AL         Barry                    3       2
21 21   AL         Barry                    3       2
22 22   AL         Barry                    3       2
23 23   AL         Barry                    3       2
24 24   AL         Barry                    3       2
```

▲ Name	Size
..	
.gitignore	40 B
.Rhistory	0 B
EPA AMPD	
Figure 1 Rep.png	32.9 KB
project.Rproj	205 B
README.md	9.7 KB
README.qmd	5.6 KB

Figure 5: Example of a Posit Cloud Interface with a Cloned Assignment Template

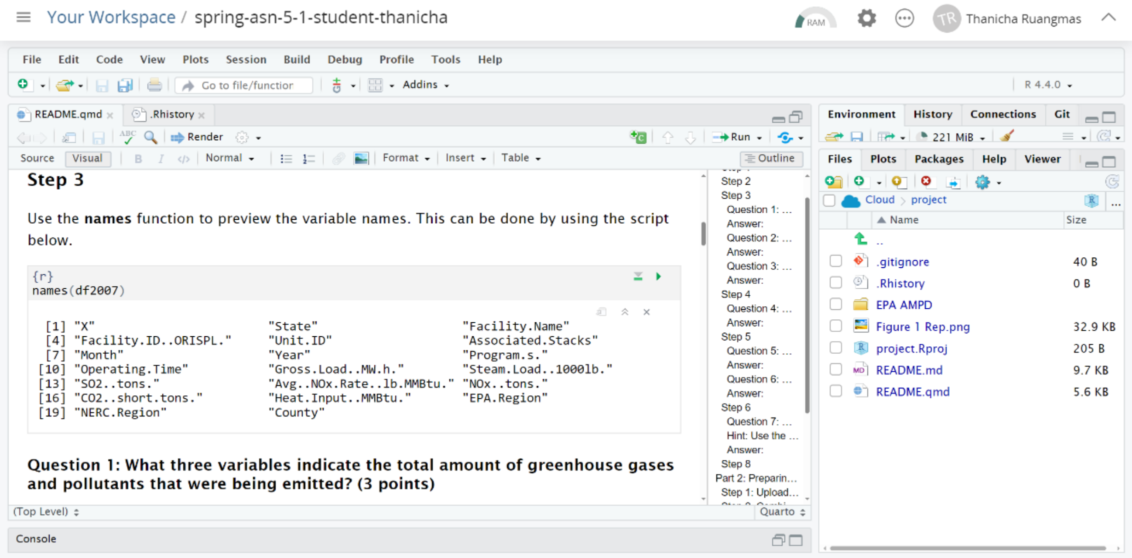
Although Posit Cloud can sometimes be clunky, I confirm a conclusion from Centinkaya and Rundel (2018) that it is a great tool to get students started. Once students use Posit Cloud more frequently, they install R and RStudio on their computers and seamlessly move their work to their computers.

One of the most critical skills taught in this course is cleaning and combining raw data ready to be modeled. Instead of working with cleaned panel data from the replication package made available by the authors, the course provides monthly data from each electricity-generating unit queried from EPA CAMPD. Because Deschenes et al. (2017) use eleven years of daily data from every electric-generating unit in the United States, I have decided to start with less detailed data that will fit into GitHub's file size limit of 100 MB and the free tier of Posit Cloud's RAM of 1 GB. I have also removed some variables from the EPA CAMPD data set that are not required to replicate the results from Deschenes et al. (2017) to satisfy the computational limitations.

4.3. Quarto

Within Posit Cloud, students write their code, generate output, and communicate their results in Quarto, a dynamic document format that allows code, output, and text description to be reported together. Quarto is an updated version of R Markdown, whose benefits have been discussed by Centinkaya and Rundel (2018) and Dvorak et al. (2019). By allowing students to analyze their code, instructors can emphasize the thought process and analysis of outputs, not just getting the correct output, which can often be done with artificial intelligence (Bean 2023).

Quarto has a more user-friendly graphical user interface than R Markdown, similar to Google Docs with embedded code chunks. This allows students to work with a graphical user interface that they are familiar with while only learning one programming language, which is R. An example of an assignment in Quarto format is shown in Figure 6. An assignment template like the one shown in Figure 6 is provided in most assignments. Students can open the template by cloning the assignment from GitHub and clicking on the README.qmd file on the bottom left tab in Figure 5.



Step 3

Use the **names** function to preview the variable names. This can be done by using the script below.

```
{r}
names(df2007)
```

```
 [1] "X"                    "State"                     "Facility.Name"
 [4] "Facility.ID..ORISPL." "Unit.ID"                   "Associated.Stacks"
 [7] "Month"                "Year"                      "Program.s."
[10] "Operating.Time"       "Gross.Load..MW.h."         "Steam.Load..1000lb."
[13] "SO2..tons."           "Avg..NOx.Rate..lb.MMBtu."  "NOx..tons."
[16] "CO2..short.tons."     "Heat.Input..MMBtu."        "EPA.Region"
[19] "NERC.Region"          "County"
```

Question 1: What three variables indicate the total amount of greenhouse gases and pollutants that were being emitted? (3 points)

Name	Size
..	
.gitignore	40 B
.Rhistory	0 B
EPA AMPD	
Figure 1 Rep.png	32.9 KB
project.Rproj	205 B
README.md	9.7 KB
README.qmd	5.6 KB

Figure 6: Paper Replication Assignment Template in Quarto

In addition, Quarto allows the user to generate many output formats, such as HTML, PDF, Word, or GitHub-Flavored Markdown (GFM). Each paper replication assignment instruction in this curriculum is in a GFM format. Students can see the assignment questions from the beginning on the webpage of their GitHub repository. To complete the assignment, students clone the GitHub repository with the questions into their Posit Cloud account and answer each question. When students are done, they render the Quarto document so that an .md file is created. Students then stage, commit, and push the .md and .qmd file to replace the original GitHub repository with a version of the document with their answers. This allows the original webpage to regenerate the webpage with their answers. Graders can view each student's completed work by going to each student's assignment repository webpage, not having to re-render their R Markdown documents, as done in Dvorak et al. (2019) and Bean (2023).

5. Teaching Environments

After learning programming syntaxes from DataCamp, students learn in two additional teaching environments that allow them to learn programming skills and solve a previously answered research question. The first mode consists of 50-minute classroom sessions led by me, the instructor. The second mode is a computing lab where students learn to use the Posit Cloud and GitHub graphical user interface and complete Paper Replication Assignments.

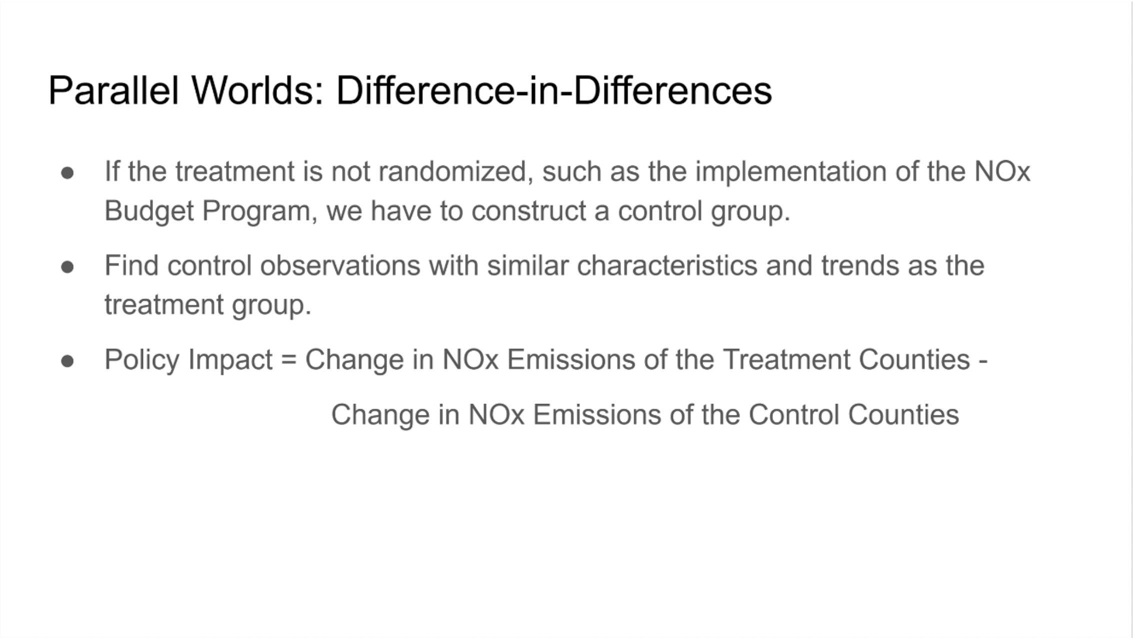
5.1. Classroom Lectures

Classroom sessions serve as a bridge between DataCamp and Paper Replication Assignments. While DataCamp teaches students the R programming language, it does not specify where to apply it. The classroom sessions also cover the Posit Cloud, RStudio, and GitHub graphical user interfaces.

Students are also introduced to the contents of Deschenes et al. (2017) and the associated dataset. Additionally, the classroom covers data cleaning skills required to clean raw data, such as the EPA CAMD, which are not included in the DataCamp courses. These skills include addressing NA values before summarizing data and creating conditional variables with the mutate and ifelse functions. The classroom also emphasizes concepts students may need further help understanding from DataCamp courses, such as finding averages and total values for each group, using a regression for prediction versus causal inference, and running and interpreting OLS regressions.

Once students have mastered basic data cleaning, the class covers data modeling concepts not taught in DataCamp. This includes different structures of data: cross-sectional, time-series, and panel. Students learn the practical use of fixed effects, described as dummy variables for each group. Table 1 outlines the weekly classroom discussion content in chronological order.

In Week 10, basic causal inference models are taught without requiring prior econometrics training, except for interpreting OLS regression models. Students start by learning about randomized control trials, building on their high school science experiments. The difference-in-differences model is then introduced to estimate treatment effects when experiments are not feasible. The model in Deschenes et al. (2017) is used as an example of adapting these concepts to real-world data. Figure 7 shows a lecture slide where the difference-in-differences concept is simplified in the context of the Deschenes et al. (2017) paper.



Parallel Worlds: Difference-in-Differences

- If the treatment is not randomized, such as the implementation of the NOx Budget Program, we have to construct a control group.
- Find control observations with similar characteristics and trends as the treatment group.
- Policy Impact = Change in NOx Emissions of the Treatment Counties - Change in NOx Emissions of the Control Counties

Figure 7: Simplification of Difference-in-Differences Taught in Class

5.2. Cognitive Apprenticeship in the Lab

Lab participation is essential for skill building for three reasons. First, the course only includes 50-minute weekly classroom sessions and has no exams. Second, the course avoids requiring students to learn additional programming languages like Git, so many point-and-click steps are necessary to create a reproducible workflow. Lastly, students must understand specific details in Deschenes et al. (2017) to grasp the logic behind certain data cleaning steps.

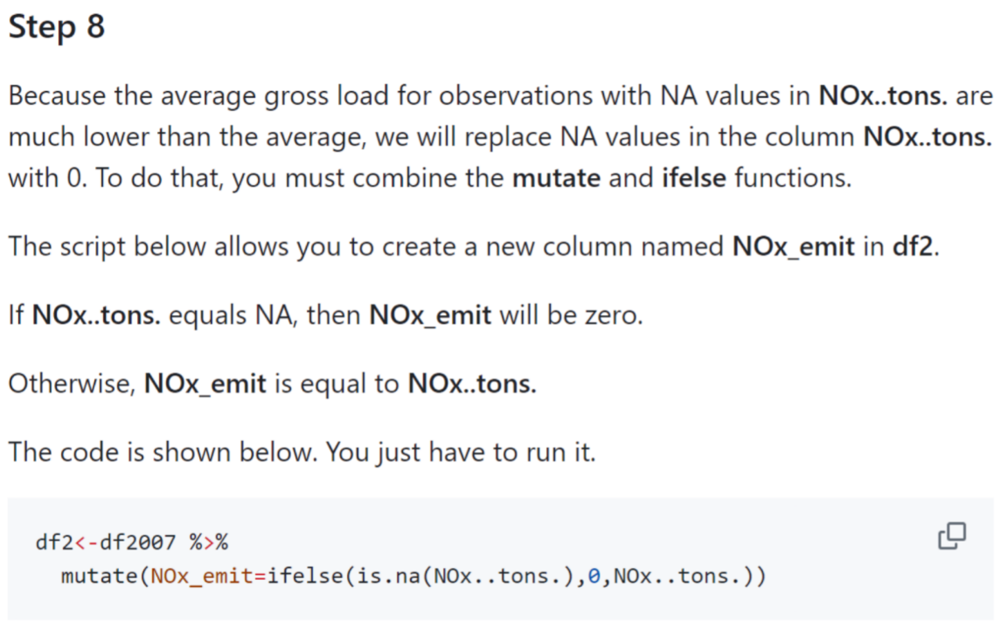
In the lab, students are supervised by peer research mentors—students who have completed the course sequence and been selected to return as mentors. Peer research mentors receive 2-course credits a semester for spending about 8 hours per week mentoring new students and continuing their research projects. At the beginning of the semester, students and peer research mentors complete a survey to indicate their weekly availability. About five students are matched with one peer research mentor and meet once a week in the FIRE Sustainability Analytics lab to work on the Paper Replication Assignments.

Cognitive Apprenticeship methods, which involve demonstrating and explaining the reasoning behind each step to students (Collins 1989), are used to train students. Peer research mentors employ five Cognitive Apprenticeship methods—modeling, articulation, scaffolding, coaching, and reflection (Shah 2023). Before training students each week, peer research mentors meet to discuss which portion of the Paper Replication Assignment requires their explanation and which Cognitive Apprenticeship method to use. Each peer research mentor then leads one lab session for about an hour. Afterward, students work independently but are encouraged to return to the lab if they have questions. The rest of this section provides examples of how Cognitive Apprenticeship methods are utilized to teach reproducible workflows and create conditional variables.

5.2.1. Modeling

This method refers to the instructor showing how to complete a task so students can observe the process (Shah 2023). Modeling is used to teach students version control for the first time in Week 4 of the course. A peer research mentor demonstrates each step using the GitHub and RStudio graphical user interfaces, allowing the students to follow along. As this is all mechanical, students can refer to this webpage for step-by-step instructions in later weeks.

A commonly used data-cleaning task that new programmers have difficulty learning is creating a conditional variable, which involves using the ifelse and mutate functions together. Modeling, or live coding, is the first step after Week 5's lecture to teach students how to do it. Once students reach Step 8 in Assignment 1, peer research mentors will review the reasoning behind conditional variables and demonstrate how to create one. This step in the assignment is shown in Figure 8.



Step 8

Because the average gross load for observations with NA values in **NOx..tons.** are much lower than the average, we will replace NA values in the column **NOx..tons.** with 0. To do that, you must combine the **mutate** and **ifelse** functions.

The script below allows you to create a new column named **NOx_emit** in **df2**.

If **NOx..tons.** equals NA, then **NOx_emit** will be zero.

Otherwise, **NOx_emit** is equal to **NOx..tons.**

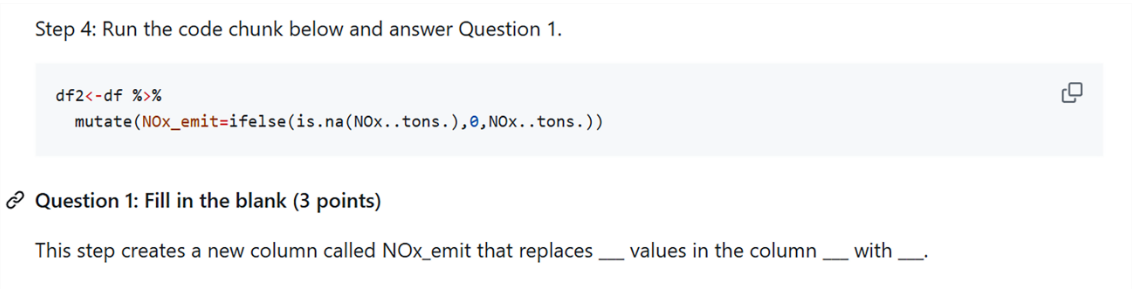
The code is shown below. You just have to run it.

```
df2<-df2007 %>%
  mutate(NOx_emit=ifelse(is.na(NOx..tons.),0,NOx..tons.))
```

Figure 8: Instructions on how to create conditional variables in Paper Replication Assignment 1

5.2.2. Articulation

In Week 6, peer research mentors check the students' understanding of conditional variables by letting them explain the logic behind conditional variable creation. This part of the assignment is shown in Figure 9, where students are tasked with explaining the script's function without having to write it themselves.



Step 4: Run the code chunk below and answer Question 1.

```
df2<-df %>%
  mutate(NOx_emit=ifelse(is.na(NOx..tons.),0,NOx..tons.))
```

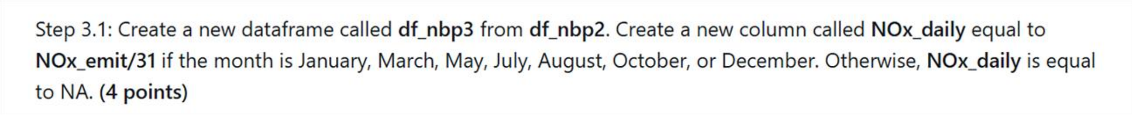
Question 1: Fill in the blank (3 points)

This step creates a new column called NOx_emit that replaces __ values in the column __ with __.

Figure 9: Instructions on how to create conditional variables in Paper Replication Assignment 2 Part 1

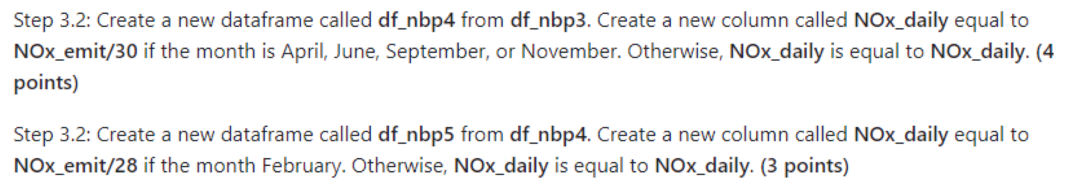
5.2.3. Scaffolding

In the same assignment, scaffolding, where assistance from the instructor is slowly removed, is used. Initially, the peer research mentor uses modeling again to live code the problem in Step 3.1. shown in Figure 10. The next step, shown in Figure 11 students are challenged to modify the script created in the previous step. The peer research mentors then check if each student is doing it correctly.



Step 3.1: Create a new dataframe called **df_nbp3** from **df_nbp2**. Create a new column called **NOx_daily** equal to **NOx_emit/31** if the month is January, March, May, July, August, October, or December. Otherwise, **NOx_daily** is equal to NA. **(4 points)**

Figure 10: Instructions on how to create conditional variables in Paper Replication Assignment 2 Part 3



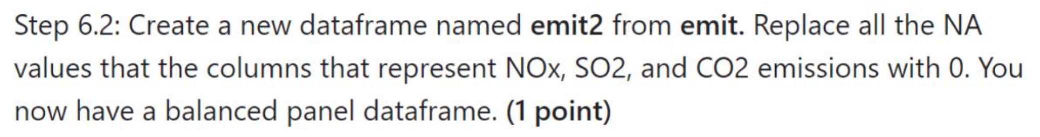
Step 3.2: Create a new dataframe called **df_nbp4** from **df_nbp3**. Create a new column called **NOx_daily** equal to **NOx_emit/30** if the month is April, June, September, or November. Otherwise, **NOx_daily** is equal to **NOx_daily**. **(4 points)**

Step 3.2: Create a new dataframe called **df_nbp5** from **df_nbp4**. Create a new column called **NOx_daily** equal to **NOx_emit/28** if the month February. Otherwise, **NOx_daily** is equal to **NOx_daily**. **(3 points)**

Figure 11: Instructions on how to create conditional variables in Paper Replication Assignment 2 Part 3

5.2.4. Coaching

In the following assignment in Week 7, students must write the script for creating conditional variables by themselves, with minimal explanation provided as shown in Figure 12. The peer research mentor then gives feedback, or coaches, the students if their answers are not correct.



Step 6.2: Create a new dataframe named **emit2** from **emit.** Replace all the NA values that the columns that represent NOx, SO2, and CO2 emissions with 0. You now have a balanced panel dataframe. **(1 point)**

Figure 12: Instructions on how to create conditional variables in Paper Replication Assignment 3

5.2.5. Reflection

Reflection refers to the instructor allowing students to solve a problem themselves and compare their methods to others afterward (Shah 2023). Although the course has no exam, students are assessed on their data-cleaning ability through complete Paper Replication Assignment 4: Make Appendix Figure 2A. In this assignment, students have up to two weeks to create Figure 2A from the Appendix section of Deschenes et al. (2017). Peer research mentors are not allowed to guide them on the assignment, but they can refer to previous assignments to complete each step. Afterward, students can show their final table and plot work to peer research mentors and articulate their reasoning behind each step. Peer research mentors will assess if the table they created is correct, if the figure they created is close enough to the figure in the paper, and if students can explain their work. If students do not get it correctly, peer research mentors can coach and let them try again.

6. Lessons Learned

6.1. Teaching data literacy can be done without prerequisites by guiding students to replicate a published paper

Although other institutions have allowed students to replicate papers in upper-level courses (Hoffler 2013; Vilhuber et al. 2022), paper replication is a valuable introductory pedagogical tool, offering students practical experience in applying programming and research skills to real-world scenarios. However, effective teaching requires a structured curriculum to equip students with the necessary skills to navigate the complexities of empirical research.

In a class survey conducted at the end of FIRE Semester 2 in 2023, students were asked which class activities helped them learn to use R programming to analyze data. Students were given the option of responding with "To a great extent," "Somewhat," "Very little," or "Not at all." The results are shown in Figure 13 below. As shown in the figure, about 75 percent of students responded that working on Paper Replication Assignments taught them to a great extent.

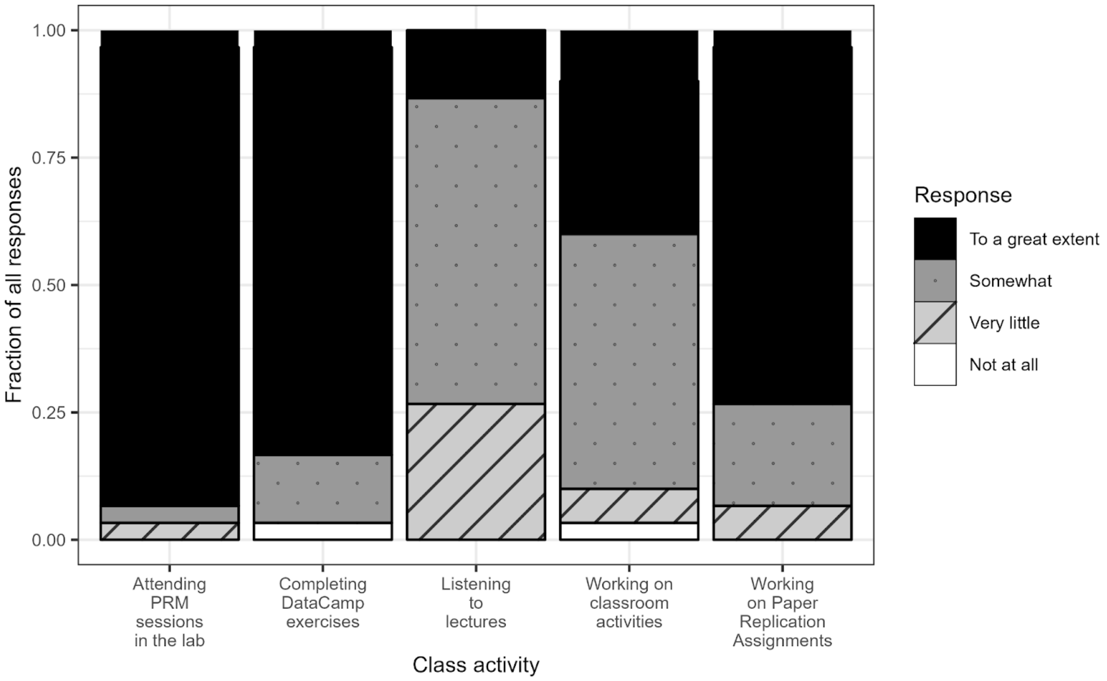


Figure 13: Class activities that helped students learn to use R programming to analyze data

6.2. Hands-on training is essential to teaching data literacy

One requirement noted in previous studies is the high demand for hands-on training, which translates to small classes (Bean 2023) or a high teaching assistant-to-student ratio (Janz 2021; Vilhuber et al. 2022). Using a high-impact pedagogical tool such as Cognitive Apprenticeship immensely reduces students' frustration when learning these skills. As shown in Figure 13, among all the class activities in the course, attending peer research mentor sessions in the lab has helped students learn R programming for data analytics the most. To reduce the demand for time, instructors can outsource teaching introductory skills to online platforms, such as DataCamp. As shown in Figure 13, DataCamp exercises are the second most helpful skill for learning programming with R for data analytics.

6.3. Reproducible workflow practices should be taught at the inception of data analytics training

While reproducibility is the gold standard for scientific research (Janz 2016), only a fraction of economics researchers share their data and code (Miguel 2021). Although some journals have policies on sharing data and code, there is still a lack of transparency in cleaning and merging raw data (Hoffler 2013). To increase the practice of reproducible research, instructors should incorporate reproducible research workflow practices from the inception of their training. The practice encourages students to organize their work, communicate their thought processes, and showcase their skills (Dvorak 2019). By setting this standard for students, educators can potentially create a “trickle-up” effect that can influence the profession (Ball and Medeiros 2012). Despite its potential benefits, Underwood et al. (2023) reported that only about 14 percent of liberal arts colleges teach reproducible methods in introductory statistics courses for economics majors.

This course confirms findings from Centinkaya and Rundel (2018) that reproducible workflow practices should be embedded in data analytics training from the beginning. With point-and-click tools such as Quarto and Posit Cloud with Git embedded, students do not have to learn an additional programming language to learn the workflow itself.

6.4. The curriculum to teach reproducible research workflow needs to evolve with industry standards continuously

As replication studies for undergraduates are being increasingly taught, I hope this paper provides a straightforward, well-established practice that instructors of other institutions can emulate. Ball and Medeiros (2012) first proposed a method of teaching reproducible workflow but noted that the system should be revised over time. This paper serves as an updated method for the same purpose. Like Ball and Medeiros (2012), I do not propose that this is the ideal system, but it should be revised over time.

7. Conclusion

Despite challenges such as instructor training and student readiness, the benefits of integrating paper replication and reproducible workflows to build data literacy are evident. A smaller institution can still offer the same curriculum with one class per week, additional one-hour hands-on sessions led by teaching assistants or peer mentors, and one to two office hours to provide additional assistance from the instructor before assignments are due. By providing a roadmap for educators and sharing insights from implementing this framework at a course at the University of Maryland, this paper aims to inspire similar initiatives and contribute to the broader adoption of reproducible research in economics education.

Acknowledgments

The author would like to thank Lars J. Olson and attendees at CTREE and ConnectUR for helpful comments on this article.

Declaration of Interest Statement

The authors report there are no competing interests to declare

References

Angrist, J. D., & Pischke, J. S. (2017). Undergraduate econometrics instruction: through our classes, darkly. *Journal of Economic Perspectives*, *31*(2), 125-144.

Ball, R., & Medeiros, N. (2012). Teaching integrity in empirical research: A protocol for documenting data management and analysis. *The Journal of Economic Education*, *43*(2), 182-189.

Bauer, G., Breznau, N., Gereke, J., Höffler, J. H., Janz, N., Rahal, R. M., ... & Soiné, H. (2023). Teaching constructive replications in the behavioral and social sciences using quantitative data. *Teaching of Psychology*, 00986283231219503.

Bean, B. (2023). Teaching Reproducibility to First Year College Students. *Journal on Empowering Teaching Excellence, Fall 2023*.

Beckman, M. D., Çetinkaya-Rundel, M., Horton, N. J., Rundel, C. W., Sullivan, A. J., & Tackett, M. (2021). Implementing version control with Git and GitHub as a learning objective in statistics and data science courses. *Journal of Statistics and Data Science Education*, *29*(sup1), S132-S144.

Çetinkaya-Rundel, M., & Rundel, C. (2018). Infrastructure and tools for teaching computing throughout the statistical curriculum. *The American Statistician*, *72*(1), 58-65.

Collins, A., Brown, J. S., & Newman, S. E. (1989). Cognitive apprenticeship: Teaching the crafts of reading, writing, and mathematics. In *Knowing, learning, and instruction* (pp. 453-494). Routledge.

Deschenes, O., Greenstone, M., & Shapiro, J. S. (2017). Defensive investments and the demand for air quality: Evidence from the NOx budget program. *American Economic Review*, *107*(10), 2958-2989.

Dvorak, T., Halliday, S. D., O'Hara, M., & Swoboda, A. (2019). Efficient empiricism: Streamlining teaching, research, and learning in empirical courses. *The Journal of Economic Education*, *50*(3), 242-257.

Gitter, S. (2021). A Guide for Student-led Undergraduate Research in Empirical Micro-Economics. *Journal of Economics Teaching*, 5(3), 83-115.

Halliday, S. D. (2019). Data literacy in economic development. *The Journal of Economic Education*, *50*(3), 284-298.

Henderson, A. (2018). Leveraging the power of experiential learning to achieve higher-order proficiencies. *The Journal of Economic Education*, 49(1), 59-71.

Höffler, J. H. (2014, March). Teaching replication in quantitative empirical economics. In *European Economic Association & Econometric Society 2014 Parallel Meetings (Toulouse), August* (Vol. 28).

Horton, N. J., Alexander, R., Parker, M., Piekut, A., & Rundel, C. (2022). The growing importance of reproducibility and responsible workflow in the data science and statistics curriculum. *Journal of Statistics and Data Science Education*, *30*(3) 207-208.

Hoyt, G. M., & McGoldrick, K. (2017). Promoting undergraduate research in economics. *American Economic Review*, *107*(5), 655-659.

Janz, N. (2016). Bringing the gold standard into the classroom: replication in university teaching. *International Studies Perspectives*, *17*(4), 392-407.

Klein, C. C. (2013). Econometrics as a capstone course in economics. The Journal of Economic Education, 44(3), 268-276.

Marshall, E. C., & Underwood, A. (2019). Writing in the discipline and reproducible methods: A process-oriented approach to teaching empirical undergraduate economics research. *The Journal of Economic Education*, *50*(1), 17-32.

Miguel, E. (2021). Evidence on research transparency in economics. *Journal of Economic Perspectives*, *35*(3), 193-214.

Peng, R. D. (2011). Reproducible research in computational science. *Science*, *334*(6060), 1226-1227.

Peikert, A., & Brandmaier, A. M. (2021). A reproducible data analysis workflow with R Markdown, Git, Make, and Docker. *Quantitative and Computational Methods in Behavioral Sciences*, 1-27.

Ruangmas, T., & Olson, L. J. (2024), FIRE Sustainability Analytics: An Innovative Approach to Engaging Undergraduate Students in Economics Research. *forthcoming in Applied Economics Teaching Resources*

Rundel,C., Çetinkaya-Rundel, M., & Anders, T. (2020), "ghclass: Tools for Managing Classes With GitHub," available at *http://github.com/rundel/ghclass*

Shah, A. (2023, August). Improving Students' Programming Processes using Cognitive Apprenticeship Methods. In *Proceedings of the 2023 ACM Conference on International Computing Education Research-Volume 2* (pp. 102-106).

Underwood, A., Sichel, A., & Marshall, E. C. (2023). Teaching Reproducible Methods in Economics at Liberal Arts Colleges: A Survey. *Journal of Statistics and Data Science Education*, 1-7.

Vilhuber, L., Son, H. H., Welch, M., Wasser, D. N., & Darisse, M. (2022). Teaching for large-scale Reproducibility Verification. *Journal of Statistics and Data Science Education*, *30*(3), 274-281.