Applied Economics Analysis Syllabus

2025-01-01

Course

Econ 2020: Applied Economics Analysis

Meeting Times & Location

Monday and Wednesday, 10:30 am - 11:50 am

PSTC Seminar Room

Instructor

Matthew DeHaven

Teaching Assistant

Ruchi Mahadeshwar

TA Office Hours:

Thursdays, 11:00 am - 1:00 pm

Fones Alley Seminar Room, Zoom

Course Site

https://matthewdehaven.com/course-applied-economics-analysis-2025/

Course Description

This course prepares students to conduct independent research by providing the necessary skills in programming and project organization. Topics covered will include version control, integrated development environments (IDEs), programming basics, package environments, functional programming, data visualization, data science, and more. Material will be presented using the programming language R, with some time spent on introductions to Python, Julia, and some basics of HTML. By the end of the course students should feel comfortable in the major programming languages used in economic research and in producing their own work as a replicable, sustainable project.

Learning Goals

- Able to replicate published papers in multiple programming languages
- Write clean, documented, reproducible code
- Apply software tools and best practices to economic research projects

Schedule

The schedule is subject to change as the course progresses.

Click on the icons for links to the lecture slides, in class coding exmaples, class feedback surveys, and assignments.

Table 1: Class and Assignment Schedule

#	Date	Topic	Lecture Slides	Coding Exam-ples	Assignments Due	Class Feedback
1	1/22	Intro, Git, & GitHub				
2	1/27	Visual Studio Code			PS1	
3	1/29	GitHub Projects & Branches				
4	2/03	Base R			PS2	
5	2/05	GitHub Copilot				
6	2/10	R Data Wrangling: `tidyverse`			PS3	
7	2/12	R Data Wrangling: `data.table`				
–	2/17	*No Class*			PS4	
8	2/19	Replication 1 Presentations			Replication 1	
9	2/24	*Guest:* Data Librarian				
10	2/26	R Databases and APIs			Proposal	
11	3/03	R Data Visualization: `ggplot2`			PS5	
12	3/05	R Regressions				
13	3/10	R Functional Programming			PS6	
14	3/12	R Writing Packages				
15	3/17	Websites, HTML, CSS			PS7	
16	3/19	Dynamic Documents with `Quarto`				
–	3/24	*No Class*				
–	3/26	*No Class*				
17	3/31	Crash Course: Python			PS8	
18	4/02	Crash Course: Julia				
19	4/07	TBD			PS9	
20	4/09	TBD				
21	4/14	TBD				
22	4/16	Final Presentations			Final Project	
23	4/21	Final Presentations				

#	Date	Topic	Lecture Slides	Coding Exam-ples	Assignments Due	Class Feedback
24	4/23	Final Presentations			Replication 2	

We will not use the final exam slot given by the registrar. Please use the time to study for your other finals!

Class Feedback

Each lecture will have an accompanying survey for students to fill out (see the schedule). The survey will ask about comprehension of topics covered in the lecture and will have open ended space for questions. These assignments are graded for completion only and are meant as a way to judge if any material needs to be covered again in more detail.

Assignments

Problem Sets

Problem sets will be assigned roughly once per week of material. This will end up close to 10 problem sets for the semester, possibly adjusted if the schedule changes.

Problem sets will begin with examples similar to those seen in class, but will then ask students to extend to a new application, method, or package. This may require some trial-and-error or research online, which is the goal. Problem sets will then end by asking students to apply the new material to their final project.

Class Projects

A few assignments will be worked on throughout the semester.

Replication 1

Students will be asked to replicate a published economics paper of their choice. The goal is to find a paper with some "replication files" which can be downloaded. Students will inspect the documentation, attempt to run the files, check the output, and see if they can find the data sources.

Final Project

The final project asks the student to take skills learned in the class and apply them to a research project. Some datasets will be provided to work with, or students can choose to use their own. Students are expected to perform some data cleaning, analysis, and charting. The last few classes will be set aside for students to present their final projects.

Replication 2

Students will be assigned another student's final project to replicate. This will mirror the first replication assignment, with the goal being to understand the documentation, execute the code, and validate the output. Half of the replication grade will come from completing the replication assignment, half will come from the student's project succesfully being replicated.

Grading

Each assignment will be graded out of 100 points.

Table 2: Assignment Weights

Assignment	Weight
Class Feedback	20%
Problem Sets	30%
Replication 1	10%
Final Project	30%
Replication 2	10%

Letter grades will be given according according to the following rubric:

Table 3: Letter Grade Calculation

Letter Grade	Numeric Grade
A	[80, 100]
B	[60, 79)
NP	[0, 59)

Credit Hours

You are expected to spend 180 hours on this course. You will spend approximately 35 hours on the lectures, 3 hours on each class readings and review (75 hours), 5 hours on each of the 14 assignments (70 hours).

Accessibility and Accommodations Statement

Brown University is committed to full inclusion of all students. Please inform me early in the term if you may require accommodations or modification of any of course procedures. You may speak with me after class, during office hours, or by appointment. If you need accommodations around online learning or in classroom accommodations, please be sure to reach out to Student Accessibility Services (SAS) for their assistance (sas@brown.edu, 401-863-9588). Undergraduates in need of short-term academic advice or support can contact an academic dean in the College by emailing college@brown.edu. Graduate students may contact one of the deans in the Graduate School by emailing graduate_school@brown.edu.

Acknowledgements

Material in this course comes from many locations, but I am especially thankful to the lecturers for this course before me, Michael Neubauer and Shunsuke Tsuda for sharing their materials.

I also want to acknowledge the excellent lecture slides by Grant McDermott for teaching a course on R at the University of Oregon.