Homework guidelines

Economics 7103

In this course, I have designed your homework assignments to teach the software skills and estimation methods necessary to execute and present cutting-edge empirical research in environmental and energy economics. A modern empirical research paper typically involves generating a testable hypothesis from a theory, introducing a context in which the hypothesis can be tested, and ultimately testing the hypothesis and discussing the results. Data are the pieces of information that characterize human behavior and environmental outcomes within the study context. Researchers use statistical techniques performed using computer software to analyze the data, conduct inference, and ultimately draw conclusions. Ultimately, the researcher then communicates the lessons learned from the data using a mix of graphs, tables, and prose to make an argument and draw conclusions about the hypothesis.

Writing an empirical paper is difficult. As a researcher, you will need to handle large quantities of data, make sense of it, write reproducible and accurate code to perform statistical analysis, and present the output clearly and attractively. Datasets are getting larger, statistical techniques are becoming more sophisticated, and the standards for doing good empirical research are increasing. Now, many journals require you to submit your data and code for review with the paper, and some of the best journals will actually run your code to make sure everything checks out.

These homework assignments will provide you examples of how I code my empirical projects and provide an opportunity for you to practice implementing the empirical techniques that are now the industry standard. My hope is that this will give you a head start on your research and help you to avoid the many costly mistakes that I made. It must also be said that my code may not be the most efficient way to do something, so feel free to do something differently than what I do.

Software

Required software:

1. **Python**, available for free from either of two sources:

    1. Directly from Python.org: https://www.python.org/ which requires a more technical setup process.
    2. Or from Anaconda, which might be easier: https://www.anaconda.com/download.

    In either case, I recommend following this guide to download and install Python and get it ready to use in VS Code (see item 2): https://code.visualstudio.com/docs/python/python-tutorial.

2. You can use any text editor to write Python code, but I recommend using **Microsoft VS Code**. VS Code is an editor called an IDE that contains tools to highlight code snippets intuitively, autocomplete lines of code, and an easy way to integrate AI into your coding. VS Code is available for free at https://code.visualstudio.com/.

    - Alternatively, you can use the Jet Brains PyCharm app that comes built into Anaconda that you can get for free with an educational license. This is a bit more of a fancy IDE with some cool tutorials and shortcuts. The main drawback is that I don't think it works well with Stata.
    - Before AI, I liked to use Spyder because it reminded me of Stata. This is still a viable option, but there just is no support for easy AI integration in Spyder.

3. Any free LaTeX editor. I recommend using Georgia Tech's institutional subscription to **Overleaf** available at https://www.overleaf.com/edu/gatech because it makes collaboration on projects easy later on in your career. Alternatively, you can edit TeX files in VS Code, which also has advantages.

4. A **GitHub account with the student developer pack** https://education.github.com/pack and the **GitHub desktop app** available at https://desktop.github.com/.

5. **Stata 18**, which you can download from IAC for your computer here: https://forms.iac.gatech.edu/?page_id=57. Alternatively, you can get access to Stata on the IAC VLab https://mycloud.gatech.edu/Citrix/GTMyCloudWeb/.

Students will learn applied econometric methods using Stata and Python and will learn how to automatically output tables and figures to LaTeX for presentation. You will implement version control with GitHub.

I teach Stata because it is the academic standard for economics and has peer-reviewed packages, and Python because (alongside R) it is the industry standard for economics and is an object-oriented programming language. Because Python is trending upward faster than R, often runs faster than R, has good web-scraping capabilities, can handle large data, and is considered more intuitive than R, we will focus on Python as an object-oriented programming language.

Why not MS Word? Word works well and can be faster than writing with LaTeX (and handles TeX syntax); however, the academic economics industry standard is LaTeX. In addition, you cannot automatically update tables, numbers, and figures with Word nearly as easily as with LaTeX. This course will teach you how LaTeX can automate a huge portion of the time you will spend formatting your research output.

Turning in an assignment

All coding assignments are due at midnight on the date specified. Late work is accepted, but the penalty for turning in an assignment late is 10 percentage points off the final grade per day late (and is accepted up to three days late). In this course, group work and discussion are allowed on all assignments.

A typical homework assignment will come with a simulated dataset and a series of tasks and questions related to the data. In addition, I have uploaded sample code to the course's GitHub repository to help get you started. My code may help you get an idea of what the right way to complete the task is. If you cannot figure out how to code something, the quickest way to get help is to use Google.

When completing an assignment, you will use LaTeX to typeset your answers and graphical output. In addition, you will keep a Python script or Stata do file that stores your code so that you can reproduce your statistical output. All of your code and output should be stored in a GitHub repository that you eventually turn in to me by sharing it with me and with the TA.

When turning in an assignment, share your GitHub repository with the following documents in it:

1. Your *.pdf output file containing your answers to the homework questions.

2. The *.tex files containing your LaTeX code that you use to generate the PDF.

3. Any *.pdf, *.eps, *.jpg, or *.png image files you include in your answer document (*.pdf or *.eps preferred as these are high-resolution file formats).

4. Any *.py Python scripts.

5. Any *.do Stata do files.

In theory, I should be able to fully replicate your homework submission from start to finish using your code.

Resources

To get started with GitHub, Python, Stata, and LaTeX, there are a bunch of free resources that I suggest you take advantage of.

GitHub:

- Installing and configuring GitHub desktop app: https://docs.github.com/en/free-pro-team@latest/desktop/installing-and-configuring-github-desktop
- Getting started with GitHub: https://docs.github.com/en/get-started
- Using GitHub with Overleaf: https://www.overleaf.com/learn/how-to/Using_Git_and_GitHub.
- Troubleshooting: https://www.google.com/.

VS Code:

- Setup guide: https://code.visualstudio.com/docs/setup/setup-overview
- Set up GitHub Copilot (AI programmer) in VS Code: https://code.visualstudio.com/docs/copilot/overview
- Set up VS Code to be able to highlight and automatically run Stata code: https://gdeiana.github.io/economics/stata-VSCode/

Python:

- Using Python in VS Code: https://code.visualstudio.com/docs/python/python-tutorial
- PACE course on intro to Python on the PACE computing clusters: https://pace.gatech.edu/python-101.
- A list of fifteen intro to Python courses: https://medium.com/swlh/5-free-python-courses-for-beginners-to-learn-online-e1ca90687caf.
- Troubleshooting: https://www.google.com/.

Stata:

- Stata video tutorials: https://www.stata.com/links/video-tutorials/.
- Typing `help <command>` into the Stata console will give you access to a very helpful document for the command.
- Statalist forums with community questions and answers: https://statalist.org/forums/.
- Troubleshooting: https://www.google.com/.

LaTeX:

- A list of LaTeX tutorials: https://www.overleaf.com/learn/latex/Tutorials.
- Troubleshooting: https://www.google.com/.