ECON 481 – Data Science Computing for Economics

Background

Instructor: Lukas Hager (lghhager@uw.edu)

Grader: TBD

Class Times: Monday/Wednesday, 8:30AM-10:20AM

Office Hours: Tuesday, SAV 319F, Time TBD

Course Description

This course is designed to give you the technical foundations required to do rigorous computational analytical work in economics, either in an academic or professional setting. This includes learning how to gather data programmatically, use industry standard tools to analyze it, and build custom tools to analyze it as well. While it is not a theoretical course itself, it will reference econometric methods from other theoretical courses in discussing implementation. It is heavily based on the same course delivered by Drs. Phil Erickson and Konstantin Golyaev.

The main programming languages used in this course are Python and SQL. Prerequisites

- Calculus (MATH 126)
- Intermediate microeconomics (2.0 in ECON 300)
- Intro probability and statistics with calculus (STAT 311, MATH 390/STAT 390, or QSCI 381)
- Introductory econometrics (ECON 382 or ECON 482)
- MATH 208 recommended

Learning Objectives

By the end of this course, students will be able to:

- Understand basic `python` data structures, functions, and syntax
- Implement standard software quality control through versioning using `Git`
- Import economic data from various formats and transform them into usable data structures
- Optimize data manipulation using vector-centric `python` packages
- Create informative and professional-looking figures
- Fit econometric models to data using industry-standard packages, including `statsmodels` and `scikit-learn`.
- Write and test modules
- Query and manipulate data from `SQL` databases

Required Materials

There are no required materials for this course. However, I would recommend the following books as references:

- Python for Data Analysis (2nd Edition), Wes McKinney
- A Gentle Introduction to Effective Computing in Quantitative Research, Harry J. Paarsch and Konstantin Golyaev
- R for Data Science, Garrett Grolemund and Hadley Wickham

Not only should these books be helpful for this course, but they are excellent references for anybody doing research or work in quantitative fields. The first two both focus on Python, which will be the primary language used in this course. The third gives similar information, but for the `R` language. While we will not be using R in this course, it is commonly used as an alternative language to Python for econometrics and data science. This book should be a useful tool to help you apply the concepts from this course to R as well. We also suggest you get comfortable with Stack Overflow, which is the quintessential online Q&A forum for programmers, numerical or otherwise. The primary benefit of Stack Overflow is generally finding answers to your question when it was already asked by someone else.

Grading

Grades will be based on homework assignments (60%), final project proposal (10%), and a final project (30%). Assignments will be given on a roughly weekly basis and should be submitted by 11:59pm on their due date through Canvas. For each 24-hour period after the submission deadline, 25% of the total possible points will be taken from the assignment. Class notes, books, Stack Overflow, etc. can be used for these assignments. You may use ChatGPT *but you must be able to explain the code you submit (e.g. via comments)*. You may also work with other class members, although you must turn in your own assignments. In deciding how much to work with others on your assignments, remember, when you're interviewing at a tech company and are given a programming task as a part of the hiring process, you will have no one but yourself to rely on. Make sure you really learn your stuff.

The final project will be based on the semester's material. As a part of this project, you will be required to write a package in `python` as part of a group of no more than three, use that package to perform some type of economic analysis on an actual dataset, and present the analysis to the class. The code, data, and presentation should be included in a .zip archive and submitted electronically. You must also provide the link to a GitHub repository with commits by all authors. The due date for the project is the first day of finals week. A proposal for your project is due the last Friday of Week 7 and should be no more than two pages.

Schedule

The course will roughly follow this schedule of computation concepts and economics applications:

- Week 1: Introduction to `python`, programming basics, `git`
- Week 2: Vectorization, `numpy`, `scipy`, Likelihood Estimation
- Week 3: `pandas`, Visualization with `matplotlib` and `seaborn`
- Week 4: Analysis with `statsmodels` and `scikit-learn`
- Week 5: Web scraping
- Week 6: Testing, writing modules
- Week 7: Basic `R`, integrating `R` and `python`
- Week 8: Basic `SQL` and working with databases
- Week 9: More `SQL`, Project Presentations
- Week 10: Project Presentations

Additional University Policies

Religious accommodations

Washington state law requires that UW develop a policy for accommodation of student absences or significant hardship due to reasons of faith or conscience, or for organized religious activities. The UW's policy, including more information about how to request an accommodation, is available at Religious Accommodations Policy. Accommodations must be requested within the first two weeks of this course using the Religious Accommodations Request form.

Access and Accommodations

Your experience in this class is important to us. It is the policy and practice of the University of Washington to create inclusive and accessible learning environments consistent with federal and state law. If you have already established accommodations with Disability Resources for Students (DRS), please activate your accommodations via myDRS so we can discuss how they will be implemented in this course. If you have not yet established services through DRS, but have a temporary health condition or permanent disability that requires accommodations (conditions include but not limited to; mental health, attention-related, learning, vision, hearing, physical or health impacts), contact DRS directly to set up an Access Plan. DRS facilitates the interactive process that establishes reasonable accommodations. Contact DRS at uwdrs@uw.edu.