

ECON 242: Data Analysis & Computing for Economics and Business

Fall 2026 — University of Richmond

ROBINS SCHOOL OF BUSINESS

Instructor:	Yumin Hong	Class Times:	MW 12:00pm – 1:15pm MW 1:30pm – 2:45pm
E-Mail:	yumin.hong@richmond.edu	Classroom:	BUS 225
Office:	Q273A Robins School of Business	Website:	Canvas
Office Hours:	Tu 4–5pm or by appointment		

Course Description: This course will teach students both the basics of programming and how to use these programming skills to solve economics/business problems that require extracting and analyzing data and automating tasks. Students completing this course will also be able to apply the knowledge learned in this course towards learning other programming languages and projects. By the end of this course, my goal is for you to be able to:

1. Understand how programming and quantitative methods are used to analyze economic/business data.
2. Approach data with greater confidence and use it to investigate economic/business questions.
3. Build foundational programming skills and know how to use available resources when you need help.

Catalog Course Description: Introduction to data analysis and programming in economics/business. Students will learn general-purpose programming techniques and how to use computing tools for data preparation and analysis, along with their application to economic/business problems.

Relevant RSB Economics Program Learning Objectives: This course also contributes to the following learning objectives of the undergraduate economics program:

- O1.1 Students will produce solutions to economic problems using appropriate quantitative and qualitative techniques.
- O1.2 Students will identify core issues, and evaluate and apply evidence in support of a coherent position or recommendation.
- O2.1 Students will make a professional and effective oral presentation.
- O3.1 Students will demonstrate an ability to work in teams and collaborate with others.

Learning Outcomes in Quantitative Data Literacy: Our learning outcomes, as described by the Web of Inquiry curriculum, are as follows:

- Students will represent information and formulate questions in forms amenable to quantitative analysis, recognizing multiple approaches to communicating quantitative information.
- Students will apply relevant quantitative approaches to solve problems and analyze data.
- Students will evaluate assumptions, limitations, biases in, and/or ethical implications of particular analytical frameworks and/or study designs.
- Students will formulate results, draw appropriate conclusions, and communicate findings using relevant quantitative evidence.

COURSE OUTLINE & TENTATIVE SCHEDULE

Part I: Introduction to Programming with Python (Weeks 1–6)

This block introduces basic programming concepts, syntax, and scripting logic.

- **Topics covered:**
 - Introduction to Jupyter Notebooks, Variables, and Basic I/O
 - Data Types: Numbers, Strings, and Date/Time formatting
 - Control Flow: Boolean Logic, if Statements, and Loops
 - Data Structures: Lists, Dictionaries, Tuples, and Sets
 - Functions, Modules

Part II: Data Analysis & Visualization (Weeks 7–12)

Focuses on using Pandas for data manipulation and visualization libraries for reporting.

- **Topics covered:**
 - Data Cleaning, Handling Missing Values, and Merging Datasets
 - Split-Apply-Combine Operations and Aggregations
 - Data Visualization: Matplotlib, Seaborn, and interactive Plotly
 - Spatial Data basics with Geopandas and introductory Statsmodel

Part III: Final Project (Weeks 13–14)

Students apply the full course toolkit to an end-to-end data analysis project, working in small groups.

- **Project:** Choose one among datasets provided by the instructor. Each group will pose their own question, then clean, analyze, and visualize the data to answer it.
- **Deliverables:** A reproducible Jupyter notebook and a short in-class presentation of key findings during the final weeks.

**Note: Topics are tentative and subject to adjustments throughout the semester.*

Grading Structure: You will be evaluated based on your performance in

Evaluation component		Points (%)
Lab Assignments	Completion-based (Lowest two scores dropped)	40 points (20%)
Quizzes	4 Quizzes (Lowest dropped); Dates TBA	30 points (15%)
Midterm 1	<i>September 30</i>	40 points (20%)
Midterm 2	Cumulative; <i>November 18</i>	60 points (30%)
Final Project	Group work	30 points (15%)
Total		200 points (100%)

- **Midterms:** In the event of a documented, unavoidable absence from one midterm, the missed exam will be assigned a score equal to 75% of your % score earned on the other midterm. You must complete at least one of the two midterms; missing both midterms is not permitted.
- **Quizzes:** There are no make-up quizzes. A missed quiz receives a zero, which may be dropped as your lowest quiz score.
- **Late submissions:** Late submissions are not accepted.
- **Attendance:** Regular attendance is expected. Even though there is no specific attendance requirement for the course, students who miss more than 7 classes (i.e. more than 25% of the classes) may fail the course.

Textbooks: No required textbook. All reading materials, code notebooks, and handouts will be provided on Canvas/in class.

Regrade Requests: If you believe an assignment, quiz or exam has been graded incorrectly, you may submit a written regrade request using the Regrade Request Form, available on Canvas, by the beginning of the next class after the exam is returned. A regrade request may result in the entire assignment/quiz/exam being reviewed, and the resulting score may increase, decrease, or remain unchanged. Regrade requests submitted after this deadline will not be considered.

Group Membership: I believe programming is learned faster with others than alone. Explaining a bug out loud to someone else is one of the fastest ways to find it, and explaining a concept you just learned is the fastest way to find out whether you actually learned it. Your group is a standing place to work through the labs, get stuck and unstuck together, and compare approaches. Lab assignments in this course are recommended group work, although they are submitted individually. I encourage you to plan, debug, and talk through every part of a lab together, but each of you writes and uploads your own notebook. If your group is stuck together, come talk to me.

- **Forming your group:** I will use information from the first-week survey to form groups of 2-4 students. I will do my best to accommodate preferences when possible, but cannot guarantee all requests.
- **Changing your group:** Groups are expected to remain together throughout the semester. However, if your group is not working well for you, please come talk to me. Group membership can be adjusted when needed, and you do not owe me a detailed explanation for requesting a change. I may also reorganize groups if necessary.
- **Final Project:** Your study group will be your default project team for Part III, but project groups may be adjusted if needed.

Classroom Climate: I begin from the premise that every student in this course is fully capable of engaging with and mastering the material, and my job is to meet you at least halfway in that process. Our classroom should be an inclusive space in which ideas, questions, and even misconceptions can be raised and discussed with respect. Most problems admit more than one route to a solution, and we are all intellectually richer when we remain open to approaches other than our own. Accordingly, all students should feel comfortable expressing thoughts and asking and answering questions in class. **Do not insult, belittle, or make fun of other students.** A student who does so will be dismissed from the class for the remainder of the term. I look forward to getting to know each of you, both as individuals and as a learning community.

Classroom Conduct & Devices: You may not use any recording device in class, including devices that capture audio, video, or photographs. All class materials are copyrighted to the instructor and may not be distributed without my permission. If you feel you need an exception to any of these rules, please come talk to me.

Names & Pronouns: Class rosters are provided to me with each student's legal name and sometimes do not list gender or pronouns. I will gladly honor your request to be addressed by a chosen name and/or the pronouns that correspond to your gender identity; please let me know early in the semester so I can update my records. I expect chosen names and pronouns to be respected at all times in our classroom. I also recognize that honest mistakes in addressing one another happen (including by me), so I encourage an environment of openness to correction and learning.

Academic Integrity, Honor, Trust & Use of AI: Academic integrity, honor and trust are fundamental to the UR's mission. Collaboration is permitted and encouraged, but every student must write and submit their own work. You're welcome to use generative AI to assist you in understanding and completing coding assignments,

but any written answers or reflections you submit should be in your own words. AI can be a useful tool, but I want to see what you have understood and learned from the course.

You should be prepared to discuss, explain, or be quizzed on any assignment completed outside of class. This course includes lab assignments and a final project involving empirical analysis. After submitting your work, you should be prepared to explain what each piece of your code does and how your analysis works. Our goal is to build the skill of using AI as a resource while still thinking independently.

During in-class quizzes and exams, you may not speak with any other student, use any communication device or notes, or leave the room. Any violation of this rule, regardless of the subject matter of the communication, will be considered academic misconduct and a violation of the [University Honor Code](#). Additionally at the beginning of the exam, I will ask you to turn off their phone and place it in a box at the front of the room, to be picked up once they have submitted the exam to me. You will not receive an exam until you have turned in your phone. Any deviation from these guidelines may result in failure of the class. I take the University's [Community Standards](#) very seriously and trust that all students have read and understood them.

Accessibility & Accommodations: The University is committed to providing an accessible and inclusive learning environment consistent with University policy and federal and state law. Please let me know if you experience any barriers to learning in this course so that I can work with you to ensure you have an equal opportunity to participate fully. If you have (or think you may have) a disability, there is a formal process at UR for registering it and requesting appropriate accommodations: students with learning disabilities, attention deficit disorders, and psychiatric disabilities can contact [Counseling and Psychological Services \(CAPS\)](#), and students with physical or medical disabilities can contact [Student Health Center](#). More information is available at <https://disability.richmond.edu/>. If you are already registered, please share your Disability Accommodation Notice (DAN) with me as early in the semester as possible so we can discuss your approved accommodations. If you have approved testing accommodations, please make any necessary [Testing Center](#) reservations as early as possible in the semester.

Mental Health Resources: College can be stressful, and your well-being matters more than any single course. If you are struggling, for any reason, and believe it is affecting your performance in this class, please reach out to me so I can connect you with whatever resources or support I can provide. You can also contact [Counseling and Psychological Services \(CAPS\)](#) directly.

Course Computing Environment: All programming in this course is done on the University's shared Jupyter server, known as **Spiderweb**. You do **not** need to install Python, Jupyter, or any packages on your own laptop: everything runs on the University's servers, and you interact with it through your web browser.

- **Access:** <https://spiderweb.richmond.edu:8000/> and log in with your UR NetID.
- **On campus:** works directly without extra steps.
- **Off campus:** requires the University VPN — see below.
- Your files persist between sessions. Nothing is lost when you close your browser or restart your laptop.

Off-Campus Access (VPN): Spiderweb can only be reached from the campus network. If you are on campus WiFi, no extra steps are needed. If you are working from off campus (home, traveling, etc.), you must connect to the University VPN first, every time, before Spiderweb will let you in.

- **Software:** Cisco Secure Client (also called AnyConnect). Install it once on your personal laptop (Windows, Mac, iOS, and Android are all supported).
- **Server address to enter during setup:** `cipher.richmond.edu`
- **Login:** your UR NetID and password, plus Duo two-factor approval (the same Duo app you already use for other UR logins).
- **Setup instructions:** [How to Install and Configure Cisco Secure Client \(AnyConnect\) VPN](#)
- **Background / what the VPN is for:** [What is the VPN?](#)

I recommend installing and testing the VPN before the first week of class if you plan to use it, not the night before an assignment is due. If you run into trouble, contact UR IS using the instructions on the pages above. I do not have the ability to reset your VPN access.