

# Syllabus AGECE 652 – Spring 2024

---

*This syllabus is subject to revisions. You will be notified of any changes. Please monitor your Purdue email and Brightspace for updates. This version was created on Jan. X, 2024.*

## Course information

---

**AGECE 652 - Application of Quantitative Analysis: Mathematical Programming**

**CRN:** 10357

**Meeting days and times:** Tue/Thur 4:30 - 5:45 PM, Krannert Building (KRAN) G007

**Instructional Modality:** Face-to-face

**Course credit hours:** 3.0 credits

**Course website:** Brightspace (<https://purdue.brightspace.com>)

**Prerequisites:** This course is designed to integrate with other components of the Ag Econ PhD core and, therefore, it requires students have mastered a series of technical skills. The prerequisites for this course are:

- ECON 615 (or an equivalent graduate-level course in mathematics for economics). You should have a thorough understanding of matrix algebra, differential calculus, and analytical optimization methods used in economic modeling.
- ECON 606 and 607 (or an equivalent graduate-level course in microeconomic theory). You should have mastery over classic graduate-level microeconomic models (e.g., consumer and producer behavior, market equilibria, imperfect competition, decision-making under uncertainty, and game theory concepts such as Nash and subgame perfect equilibria.)
- ECON 670 and be concurrently taking AGECE 651. You should have a solid understanding of probability, statistics, and fundamentals of econometrics (e.g., distributions, moments, estimators and their properties, and inference).

In addition, some familiarity with computer programming is desirable but not required. **If you do not satisfy any of the prerequisites for this course, please contact the instructor.**

## Instructor information

---

**Instructor:** Diego S. Cardoso

**Office hours:** Thursdays, 2 PM to 3 PM.

**Email:** [cardosod@purdue.edu](mailto:cardosod@purdue.edu). Please include "AGEC652" in the subject for all emails. I will read and respond to course-related emails in timely fashion during weekdays between 9 AM and 5 PM.

**Office:** Krannert 636

**Phone:** (765) 494-4212

- [Course description](#)
  - [Learning outcomes](#)
  - [Learning resources and technology](#)
    - [Readings](#)
    - [Software](#)
    - [Use of artificial intelligence \(AI\) or Large Language Models \(LLM\)](#)
    - [Hardware](#)
  - [Assignments and grading](#)
    - [Assignments](#)
    - [Grading](#)
  - [Academic integrity](#)
  - [Attendance policy](#)
  - [Course evaluation](#)
  - [Course schedule](#)
  - [Nondiscrimination statement](#)
  - [Accessibility](#)
  - [Mental health and wellness statement](#)
  - [Basic needs security](#)
  - [Emergency preparation](#)
- 

## Course description

---

This course will introduce you to key computational methods used in applied economics research and prepare you to use these methods in your own projects. The overall objective of this course is to expand your skills for applied research by adding numerical methods to your toolset: you will understand why and when computational methods are needed in economics research, learn to identify and model economic problems that can be solved with such methods, and apply the appropriate method for each type of problem.

This course can be conceptually divided into three parts. In the first part, we will cover the fundamentals of scientific computing and best practices for empirical research. In the second part, we will study and apply core computational methods, including numerical calculus and linear algebra, systems of equations, and constrained and unconstrained optimization. In the third and final part, we will see how to take to data the models we learned to solve and introduce structural estimation of economic models.

# Learning outcomes

---

By the end of this course, you should be able to:

1. Set up and manage a scientific computing project
2. Numerically solve derivatives and integrals
3. Find numerical solutions for linear and nonlinear systems of equations
4. Find numerical solutions for constrained and unconstrained optimization problems
5. Program estimators typically used for structural economic modeling

## Learning resources and technology

---

### Readings

This course will be based on lecture notes/slides and select research papers. Course materials and references to research papers will be posted on the course website. While there is no official textbook for this course, I recommend that you consult these references for a deeper understanding of the methods we review in class with limited time.

The second part of the course will be closely related to select chapters from the textbooks listed below.

- [MF02] Miranda, Mario J. and Paul L. Fackler. *Applied Computational Economics and Finance*. Cambridge, MA: MIT Press, 2002.
- [J98] Judd, Kenneth L. *Numerical Methods in Economics*. Cambridge, MA: MIT Press, 1998.
- [NW06] Nocedal, J. and S. J. Wright. *Numerical Optimization*. New York: Springer, 2nd edition, 2006.

Purdue Library offers online access to [MF02] and [NW06] and physical copies of [J98].

The third part will review econometric methods typically used in structural estimation: maximum likelihood, generalized method of moments, and simulation-based methods. These are the recommended references for in-depth exposition of these topics (although most graduate-level econometric textbooks include at least one of these methods).

- [G18] Greene, William H. *Econometric Analysis*. 8th Edition. United Kingdom: Pearson, 2018.
- [W10] Wooldridge, Jeffrey M. *Econometric Analysis of Cross Section and Panel Data*. 2nd Edition. United Kingdom, MIT Press, 2010.

### Software

Coding examples in this course will use [Julia language](#). Julia is a modern open-source programming language with excellent performance for numerical methods. It has an intuitive syntax and offers many packages to automate the methods we will learn and practice in this course. Two good introductions to Julia are:

- Learning Julia (<https://julialang.org/learning>)
- QuantEcon Julia lectures 1 to 5 (<https://julia.quantecon.org/intro.html>).

While Julia is the recommended language for the course, students with prior programming experience may choose to submit their work using a suitable alternative language. Three suitable options are R, Python, and Matlab. If you plan to use an alternative language, please notify me as soon as possible, so I can include references to examples of the methods covered in the course in the language of your choice (when available).

## Use of artificial intelligence (AI) or Large Language Models (LLM)

When used properly, AI tools (such as OpenAI's ChatGPT) can greatly help you understand programming languages and write better code. However, that these tools frequently suggest incorrect or incomplete code and, therefore, require close human supervision. A competent human supervisor must have enough mastery of the subject knowledge to identify the mistakes AI tools make. As a learner of new concepts and ways of thinking, students are generally unable to distinguish correct and incorrect responses at first. Therefore, over-relying on these tools can severely hinder your learning, or worse, teach you incorrect things.

With the points above in mind, this course has three specific rules regarding the use of AI tools. Failure to follow these rules will be considered an academic integrity issue (more details below).

1. **Disclose the use of AI tools.** If you use any AI tools on any submitted assignments, include a statement summarizing which tools you used and how you used them.
2. **Do not submit code or work that was generated by an AI tool.** You may use AI tools to ask you to explain code, generate examples, and critique code you wrote. However, any solution you submit for assessment in this course must fully represent your thought process and must not include---in part or total---a reproduction of a solution generated by AI.
3. **Never submit code or results generated from code you do not fully understand.** Again, AI tools often generate incorrect or incomplete code as examples, misguided suggestions, and wrong diagnostics of errors. It is your responsibility to ensure the code you submit is correct and that you are able to explain what it does. **The instructor reserves the right to ask you to orally explain your code and solution, and your explanation will be used to determine your final assignment score. Failure to demonstrate understanding of the solution submitted might result in a zero score for that assignment.**

I encourage you to use AI tools to enhance your learning, as long as you follow the rules listed above. In particular, there are two use cases for AI tools that could be valuable for your learning and align with the rules of this course. As outlined by [Mollick and Mollick](#), the use cases are:

1. **AI as Feedback Generator.** You can ask AI tools to critique your solutions and suggest ways to improve them. For instance, you could ask an AI tool to help you interpret an error message you find when you run your code. Another example is to ask a tool to identify parts of your code that are hard to read and could be rewritten for clarity.
2. **AI as a Personal Tutor.** You can ask AI tools to act as a tutor and guide you through the steps of learning a new concept or technique. You can instruct the AI to outline key ideas, generate

additional examples, ask you related questions, and evaluate your explanations.

## Hardware

This course is designed to offer several opportunities for you to code and experiment with the methods we study. For this reason, I recommend you bring a laptop to class. Please let me know if you do not have access to a laptop for that purpose. Moreover, to complete assignments, you will need access to a computer where you can install numerical packages in your programming language of choice.

*A note on class conduct and etiquette:* Although you are encouraged to bring a laptop to class, **please limit your use of the laptop to either working on programming examples or taking notes. You are not allowed to use other mobile devices (phones or tablets) as they are distracting to you and others around you.** *If you have an emergency or other reason that requires you to monitor these devices, please let me know in advance.* If a student's use of an electronic device is disruptive to teaching and/or learning, I will ask that the student to discontinue the use of that device or leave the classroom. Please be considerate of those around you.

## Assignments and grading

---

### Assignments

There are four types of assignments: problem sets, midterm exam, mini-projects, and in-class quizzes.

**Problem sets.** There will be three problem sets relative to second part of the course (numerical methods). You may submit your work individually or in pairs. Submissions will be made via GitHub. Further instructions will be posted on the course website.

**Midterm exam.** There will be an exam covering the topics of the second part of the course (numerical methods). Exams are an individual assignment. Further details will be posted in advance on the course website.

**Mini-projects.** There will be two mini-projects covering the third part of the course (structural estimation). Mini-projects are longer problem sets that involve modeling, data generation, and estimation of structural parameters. You are encouraged to work in pairs, but you may also submit your work individually. Groups will be assigned different problems to solve---but with equivalent level of difficulty. More details will be posted on the course website.

**In-class quizzes.** There will be multiple unannounced pop-quizzes throughout the semester. These quizzes will ask you to provide a short written answer to questions related to key points of previous lectures. You will typically have 5 to 10 minutes to answer them in class. Quiz answers will be submitted individually.

### Grading

All graded items are converted to a percentage basis and aggregated using the weights below.

- Three problem sets (group): 30% (10% each)
- Midterm exam (individual): 20%
- Two mini-projects (group): 40% (20% each)
- In-class quizzes (individual): 10% (the lowest quiz score will be dropped)

The course will be graded with letter grades assigned over the traditional ten point intervals of aggregate performance in the course.

| <b>Grade</b> | <b>Score range</b> |
|--------------|--------------------|
| A+           | 97.0 - 100         |
| A            | 94.0 - 96.9        |
| A-           | 90.0 - 93.9        |
| B+           | 87.0 - 89.9        |
| B            | 84.0 - 86.9        |
| B-           | 80.0 - 83.9        |
| C+           | 77.0 - 79.9        |
| C            | 74.0 - 76.9        |
| C-           | 70.0 - 73.9        |
| D+           | 67.0 - 69.9        |
| D            | 64.0 - 66.9        |
| D-           | 60.0 - 63.9        |
| F            | 0 - 59.9           |

The lower boundaries for grade determination may be lowered at the time of final grade assignment at the instructor's discretion (e.g. grades at 78 and above may be awarded a B- ). Any adjustment in the grade scale will be applied uniformly to the aggregate grade (not to individual exams or other graded materials).

### **Score revisions**

If you wish to appeal your grade on an assignment, you must bring it to my attention in writing within 7 days of when the problem set, exam, mini-project, or quiz was submitted. Grades brought to my attention after this will not be eligible for a grade appeal. I reserve the right to regrade the entire assignment and the new grade will be final.

## **Academic integrity**

---

The work you submit in this course must reflect exclusively your effort and must not come significantly from the work of others. I encourage you to study and discuss the course materials and assignments with your peers. However, you are responsible for making sure that you follow the rules laid out in this syllabus and in Purdue University's Academic Integrity definitions (<https://www.purdue.edu/odos/osrr/academic-integrity/index.html>).

If for any reason you find yourself under excessive pressure or stress and feel tempted to break our Academic Integrity rules, **PLEASE STOP AND GET HELP** from me, the instructor. I am available and willing to work with you to help you succeed in this course through your legitimate hard work.

In addition to Purdue's Academic Integrity rules, in this course we will adopt the following specific policies:

- **Problem sets and mini-projects** are team-based and "open book". You are expected to work on a team and collaborate with your teammates, and you are encouraged to consult course materials. You may also discuss the problems with members of other teams, but the work you submit must reflect exclusively the work of your group members. Any evidence of undue collaboration or reproduction of other's solutions, including in previous offerings of this course, will be investigated as a violation of academic integrity.
- **Exams** are individual and "open book". You may consult any course materials, but you must not collaborate with or obtain assistance from another individual. Any evidence of collaboration will be investigated as a violation of academic integrity.
- **Quizzes** are individual and "closed book". You are not allowed to consult any course materials or collaborate with other individuals. Evidence of consulting materials or collaboration will be investigated as a violation of academic integrity.

**Any evidence of plagiarism in any assignment will be investigated as a violation of academic integrity. The instructor reserves the right to**

- **Ask the student to explain orally any code or solution submitted and take their explanation into account for the assignment score.**
- **Use any tools available for detecting plagiarism in code and text submitted in the assignments. You should assume these tools will be used.** These tools include, but are not limited to, [Turnitin](#) and software with [specialized algorithms for detecting code similarity](#). Comparisons might be made between submissions of enrolled students, a bank of solutions, and code available on the Internet.

Academic integrity is one of the highest values that Purdue University holds. Incidents of academic misconduct in this course will be addressed by the course instructor and referred to the Office of Student Rights and Responsibilities (OSRR) for review at the university level. **Any violation of course policies as it relates to academic integrity will result minimally in a failing or zero grade for that particular assignment, and at the instructor's discretion, may result in a failing grade for the course.** In addition, all incidents of academic misconduct will be forwarded to OSRR, where university penalties, including removal from the university, may be considered. More details are available on our course Brightspace under University Policies and Statements.

## Attendance policy

---

This course follows Purdue's academic regulations regarding attendance, which states that students are expected to be present for every meeting of the classes in which they are enrolled. Attendance will not be taken directly, but your engagement will be assessed through your participation during in-class activities and office hours and on-time assignment submissions. If you must miss a class at any point in time during the semester - be it for reasons excused by the University (see the [Academic Regulations and Student Conduct section](#) or otherwise - please reach out to me as soon as possible via Purdue email so that we can communicate about how you can maintain your academic progress. I will analyze and make accommodations on a case-by-case basis. If you find yourself too sick to progress in the course, notify your adviser (or program coordinator) and notify me via Purdue email; we will work together to make arrangements based on your particular situation. If you are unable to make direct contact with me and are unable to leave word with the department because of circumstances beyond your control, you or your representative should contact or go to the [Office of the Dean of Students website](#) to complete appropriate forms for instructor notification.

## Course evaluation

---

During the semester, I will ask for your informal feedback on the course. Participation is anonymous and voluntary. This early feedback will be highly appreciated. It is also an opportunity for your perceptions and concerns to be expressed and for me to make any appropriate adjustments to the course structure before the end of the semester.

Toward the end of this semester, you will be provided with an opportunity to give formal feedback on this course and your instructor. Purdue uses an online course evaluation system. You will receive an official email from evaluation administrators with a link to the online evaluation site and will receive a prompt to complete the survey when you log into Brightspace. Your participation is an integral part of this course, and your feedback is vital to improving education at Purdue University. I strongly encourage you to participate in the evaluation system.

## Course schedule

---

The planned schedule below lists the topics we will cover in this course and the approximate weeks when we will see them. This schedule is subject to change, as we may need to make adjustments along the way.

### **Part I - Introduction to Scientific Computing**

1. Scientific computing for applied research projects (*weeks 1-3*)
  1. Motivation for computational methods in economics
  2. Project organization, reproducibility, version control
  3. Introduction to programming



## **Part II - Core Numerical Methods**

2. Fundamentals of numerical operations (*weeks 3-4*)
  1. Numerical arithmetic
  2. Numerical differentiation
  3. Numerical integration
3. Systems of equations (*weeks 5-6*)
  1. Linear systems of equations and iterative methods
  2. Nonlinear systems of equations: derivative-free methods
  3. Nonlinear systems of equations: Newton and quasi-Newton methods
4. Optimization (*weeks 7-8*)
  1. Unconstrained optimization: derivative-free methods
  2. Unconstrained optimization: line search and trust region methods
  3. Constrained optimization: KKT, penalty methods
  4. Constrained optimization: computational modeling

## **Part III - Structural Estimation**

5. Introduction and overview of methods (*week 9*)

### **Midterm exam: March 7 (tentative)**

(*Week 10: Spring break*)

6. Maximum likelihood estimation (*weeks 11-12*)
  1. Theory and properties
  2. Applications: random utility and discrete choice
7. Generalized method of moments (*weeks 13-14*)
  1. Theory and properties
  2. Applications: estimating games
8. Simulation-based estimation (*week 15*)

### **Additional Topics (time permitting)**

9. Mixed complementarity problems, function approximation, and/or functional equations (*week 16*)

---

## **Nondiscrimination statement**

---

Purdue University is committed to maintaining a community which recognizes and values the inherent worth and dignity of every person; fosters tolerance, sensitivity, understanding, and mutual respect among its members; and encourages each individual to strive to reach his or her potential. In

pursuit of its goal of academic excellence, the University seeks to develop and nurture diversity. The University believes that diversity among its many members strengthens the institution, stimulates creativity, promotes the exchange of ideas, and enriches campus life. Purdue's full Nondiscrimination Policy Statement is available at ([https://www.purdue.edu/purdue/ea\\_eou\\_statement.php](https://www.purdue.edu/purdue/ea_eou_statement.php)).

## Accessibility

---

Purdue University is committed to making learning experiences accessible. If you anticipate or experience physical or academic barriers based on disability, you are welcome to let me know so that we can discuss options. You are also encouraged to contact the Disability Resource Center at: [drc@purdue.edu](mailto:drc@purdue.edu) or by phone: 765-494-1247.

## Mental health and wellness statement

---

If you find yourself beginning to feel some stress, anxiety and/or feeling slightly overwhelmed, try [WellTrack](#). Sign in and find information and tools at your fingertips, available to you at any time.

If you need support and information about options and resources, please contact or see the [Office of the Dean of Students](#). Call 765-494-1747. Hours of operation are M-F, 8 am- 5 pm.

If you find yourself struggling to find a healthy balance between academics, social life, stress, etc. sign up for free one-on-one virtual or in-person sessions with a [Purdue Wellness Coach at RecWell](#). Student coaches can help you navigate through barriers and challenges toward your goals throughout the semester. Sign-up is completely free and can be done on BoilerConnect. If you have any questions, please contact Purdue Wellness at [evans240@purdue.edu](mailto:evans240@purdue.edu).

If you're struggling and need mental health services: Purdue University is committed to advancing the mental health and well-being of its students. If you or someone you know is feeling overwhelmed, depressed, and/or in need of mental health support, services are available. For help, such individuals should contact [Counseling and Psychological Services \(CAPS\)](#) at 765-494-6995 during and after hours, on weekends and holidays, or by going to the CAPS office on the second floor of the Purdue University Student Health Center (PUSH) during business hours. The [CAPS website](#) also offers resources specific to situations such as COVID-19.

## Basic needs security

---

Any student who faces challenges securing their food or housing and believes this may affect their performance in the course is urged to contact the Dean of Students for support. There is no appointment needed and Student Support Services is available to serve students 8 a.m.-5 p.m. Monday through Friday.

## Emergency preparation

---

In the event of a major campus emergency, course requirements, deadlines and grading percentages are subject to changes that may be necessitated by a revised semester calendar or other circumstances beyond the instructor's control. Relevant changes to this course will be posted onto the course website or can be obtained by contacting the instructors via email or phone. You are expected to read your @purdue.edu email on a frequent basis.