Computational Methods Lecture 0:

Some Logistics and Why We Are Doi‘ng This?

Preliminaries & Logistics

- Two lectures and one recitation each week
- No textbook, self-contained slides on Brightspace
- You are encouraged to bring a laptop
- 5-6 coding assignments
- 1 final exam
- I'll do my best to make this course useful
- There is going to be maths

Your Grade

- Your final grade is a convex combination of homework and exam grades
- You choose weight $w \in \{0.3, 0.4, 0.5, 0.6, 0.7\}$ for

$$\text{final grade} = w \times \text{homework} + (1 - w) \times \text{exam}$$

Homeworks

- Homeworks are going to have a maths and a computations part
- For the maths part: show all your work
- For the computations part: submit reproducible code
- You can work in groups, but the write-up must be your own
- Indicate group members in your submission

Programming Language

- We will learn to write code as part of this course.
- The primary programming language is **MATLAB**.
- Other programming languages may be used for assignments.
- No prior programming background is assumed.
- Fundamentals are covered in lecture; recitations go deeper.

Recitations

- Provide background needed for lectures, including math review.
- Go deeper into coding concepts and practice.
- Expand on and build upon material from lectures.
- Briefly review assignments before they are assigned.
- Discuss solutions after assignments are due.

Some General Advice

For those among you struggling:

- Break down the problem into tiny steps (this is a skill that you'll hone with practice)
- I'll do my best to make each step follow from the previous one
- Tell me if I don't

Who is this course for

- The material covered in this course will be of limited value if your career goals lie in
  - real estate
  - retail banking
  - accounting
  - ...
- This course gives you a competitive advantage if you are aiming for a career in
  - academia
  - investment banking or hedge funds
  - research-oriented roles in the private sector
  - data science
  - ...

Our basic toolbox

- Numerical Root Finding: $f(x^\star) = 0$, what is $x^\star$?

  $f(x) = x - 2 \implies x^\star = 2$
- Numerical Optimization: $\max\{f(x)\}$

  $f(x) = -x^2 \implies x^* = 0$
- Numerical Differentiation: $f'(x)$

  $f(x) = x^2 \implies f'(x) = 2\,x$
- Numerical Integration: $\int f(x)dx$

  $f(x) = x \implies \int x\,dx = x^2/2 + c$
- Function Approximation: f is complicated

  $f(x) \approx \Theta(x)\,\boldsymbol{c}$

Our applications

- Simulation: asset and derivative pricing
- Simple Dynamic Programming: value function iteration

What Is an Algorithm?

Algorithm 1 Generic Decision Procedure

Require: Inputs x

if x is missing or invalid **then**
    $y \leftarrow$ fallback outcome
else if x satisfies condition A **then**
    Apply rule set A
    $y \leftarrow$ outcome A
else if x satisfies condition B **then**
    Apply rule set B
    $y \leftarrow$ outcome B
else
    Apply general rule
    $y \leftarrow$ default outcome
end if
return y

Example: Solving a Linear Equation

Algorithm 2 Solve $ax - b = 0$

Require: Real numbers a, b

1: **if** a or b are not real numbers **then**
2: $\quad S \leftarrow$ Error message
3: **else if** $a = 0$ and $b = 0$ **then**
4: $\quad S \leftarrow \mathbb{R}$ ▷ Infinitely many solutions
5: **else if** $a = 0$ and $b \neq 0$ **then**
6: $\quad S \leftarrow \emptyset$ ▷ No solution
7: **else**
8: $\quad x \leftarrow b/a$
9: $\quad S \leftarrow \{x\}$ ▷ Unique solution
10: **end if**
11: **return** S

Motivation

- When do pen and paper fail in economics?
- Let's start from a simple toy model in which agents interact economically
- As soon as we introduce the a sliver of realism, we need to resort to numerics