ORIE 6170: Engineering Societal Systems

Lecture 2: Introduction, continued

Nikhil Garg

Course webpage: https://orie6170.github.io/Spring2025/

Announcements

Paper discussion signup + instructions out soon

Next time: "Great ideas and papers" relevant to the area

Please read (some of):

- The Economist as Engineer: Game Theory, Experimentation, and Computation as Tools for Design Economics - Roth - 2002 – Econometrica
- Rittel and Webber. "Dilemmas in a General Theory of Planning" sympoetic.net/Managing_Complexity/complexity_files/1973 Rittel and Webber Wicked Problems.pdf

Organization: likely topics

- Introduction
- Prediction
- Optimization + resource allocation
- Matching & recommendations; market design broadly
- Pricing
- Miscellaneous methodologies and applications
  - Generative AI, human-AI collaboration, experimentation + evaluation, etc

Special focus this year

- algorithms in government
- Limitations and pitfalls, but also cautious “success” stories

Prediction

Machine learning/predictive systems that aim to predict future outcomes to help make decisions, broadly in use in health care, criminal justice, education.

- Aspects of "successful" deployed predictive systems
- "Against predictive optimization"; pitfalls in deploying predictive systems
- Comparing human and machine predictions

Case studies/papers

- Risk assessments for bail decisions in criminal justice
- Prediction models in child welfare
- Student educational "tracking" and predicting educational attainment

Optimization + resource allocation

Optimization and scheduling approaches used for large scale system design such as placement of public transportation and bike share systems, school bus routing, emergency management, and others. Resource allocation systems broadly such as in social services, informed by prediction models.

Case studies/papers

- Vehicle/school bus routing, for equity and efficiency constraints; Emergency vehicle routing and allocation
- Resource allocation in social services, e.g., for housing and other services
    - Resource allocation over space and time
    - Inspections and maintenance resources
- Gerrymandering/optimization for democracy

Ranking/Matching/recommendation/market design

Stable matching systems or algorithmic recommendations to help match people to each other, such as for residencies, schools, jobs, dating, etc. Market design more broadly. Ranking design for social network feeds

Case studies/papers

- School choice (matching + recommendations)
- Ranking design for social network feeds (Bridging/community notes, personalization)
- Refugee matching

Pricing

Pricing/monetary decisions by government or regulated by government, including congestion pricing, taxation auditing, regulation of algorithmic pricing collusion, etc.

Case studies/papers

- Congestion pricing
- Wireless spectrum auctions
- Tax auditing
- Algorithmic pricing collusion
- Personalized Pricing and regulation/discrimination broadly

Limits of technical approaches

- What are the limits to engineering methodologies?
- What (and who) is missed when we try to mathematize/optimize societal problems?
- How do we incorporate qualitative methods?
- What are the major criticisms made by others of market designers?
- Can we meaningfully address “wicked problems” through technology?
- How do we bring “systems level thinking” to these systems

Miscellaneous

- Algorithms in NYC government as an extended case study
- Experimentation + evaluation
- Human-AI collaboration
- Generative AI in high-stakes settings
- Limits of technical approaches
- Voting and social choice
- Other applications, such as online marketplaces

Cross-cutting methodology

Questions you need to answer

- What is your [the system's] *lever?*
- What is your *objective function?*
- How do people *react* to your lever?
  - What are people's *preferences?*
  - What are people's *strategy spaces?*
- How do people affect *each other?*
- What is the information space?
  - What is being predicted? What do you care about?
  - What do you know? What data do you have?
  - What do people know?
  - How do you acquire more information?
  - Will this change over time?
- *What happens when your system is wrong?*

Common tasks

- Understand your domain
- Write a model for the ?s on the left
- Think about "equilibria"
- Estimate preferences from historical data
- Simulate counter-factual worlds
- Experiment/Pseudo-experiment
- Evaluate as close to "real world" as possible
- Deploy a system

Methods used

- Applied modeling/stochastics
- Game theory/mechanism design
- Optimization, Algorithms
- Machine learning/statistics/data science
- Online learning/decision-making
- Experimentation
- Qualitative methods

Course themes

Be able to articulate what matters in a system

- “All models are wrong”, “The map is not the territory”
- Why did the authors include/exclude certain things in their model? What would change if they made different choices?
  What data do people have? What data is needed to answer this question?

Different questions require different methods

- Sometimes theory, sometimes empirical, sometimes qualitative
- Often a mix: how do we do research at the intersection?
- Why did the authors choose the methods they did? What would the paper look like if it was a theory/empirical/qualitative paper instead?

What is this class not?

This is not an algorithmic game theory class, or even a mechanism design class

Tim Roughgarden: [Algorithmic Game Theory (Lecture 1: Introduction and Examples) – YouTube]

We won't cover details of auctions, Gale Shapley matching

It is also not a machine learning or optimization or a "methods" course

- We're not going to go deep on any particular method in lecture
- The papers of course use (advanced) methods; we will discuss them; I will provide further resources; and in your paper reviews you will go deep on understanding a paper's methods
  - You are welcome to focus your assignments on papers whose methods you find most appropriate for your goals

Syllabus

[ORIE6170 syllabus Spring 2025 - Google Docs]

Assignments + Grading

Final project: 40%

- Project proposal/presentation, report, class presentation, peer review of a classmate's project
- Option of participating in a joint review research paper with a larger class group

Paper review + presentation: 20%

- Read a paper and write a journal-level review for it [suggested list posted soon]
- Give a 10-15 minute presentation to the class on your chosen paper

Presentation feedback: 10%

Watch two presentations and give feedback

- A paper review presentation by a classmate
- A presentation available online by an established researcher

Paper reading and discussion: 10%

- Choose 2 papers that we'll discuss in class and be a discussion leader [list posted on rolling basis]

Attendance + Participation: 10% **Each**

- Attendance mandatory. **We will take attendance in many of the classes.**
- **Participation is also mandatory.** I don't expect everyone to read each discussed paper in detail, but I expect you to read abstract + intro for almost all of them, read the model/setup for most, read main results and methods for many, and go into proof/method details for a few. **Especially important for those of you in Ithaca.**

Class structure

~10-15 days discussing papers

~3-5 lectures by me

~3-5 guest lectures

~3 days paper review presentations

~1 day project proposal presentations

~2-3 days project presentations

Course communication

~~**Course Slack channel:** First place for any question/comment~~

Ed Discussion (linked from Canvas)

Office hours: Happy to chat about anything – sign up on link in syllabus

Email: Try to avoid; but preferred over private message on Slack.

Classroom norms

- Take space, make space: allow others to join the conversation, but please contribute as you feel comfortable.
- Embrace a growth mindset. Not understanding something in a paper is the default.
- Ask questions!
- Be willing to give and receive feedback respectfully.
- Zoom norms

  Feel free to take video-off breaks as necessary, and a couple lectures of video off the entire time. But I expect you to mostly keep video on and participate.

Why active engineering of societal systems is needed

[Even beyond machine learning]

Interlocking questions at every stage

Data generation → Predictive modeling → Decisions → Deployment & Evaluation → Societal Equilibrium

What are we predicting? Is that the right target?
What data do we have for prediction?
How do human generated data biases affect predictions?
Will our predictions hold up over time?

How to use noisy/accurate predictions to make decisions?
How do prediction errors lead to decision errors?
How should we allocate scarce resources?
How do we make fair decisions under constraints and uncertainty?

What are the *population* effects of individual models?
What happens when the world changes?
Can people respond to your system to change what they're doing?

How do we design and evaluate real-world systems and policies?

Can we measure the thing we care about?

Is the computational system better than the human alternative? Under what metric?

Claims for why ML

- More standardized [humans are biased]
  - Can “hope to” remove bias from ML models
- More scalable [humans are time limited]
- “Better” predictions [ML can find patterns in data that humans can’t]

But is ML enough?

Beyond ML: prediction is not enough

Susan Athey at a recent panel:

"online marketplaces have mostly been started by libertarians...the most consistent theme is that people think marketplaces will be fine just with prices, but then they realize they need to impose rules for the marketplace to succeed"

Markets and Societal systems often fail

Why?

- Lack of *coordination* – selfish behavior by everyone makes everyone worse off. “Price of anarchy”
- Lack of *information/time* – individuals have constraints
- Fraud/other strategic behavior – manipulation by some makes others worse off
- Multiple equilibria, including some `bad’ ones

One role of the designer/engineer is to prevent such failures

More generally: predict → sort by predictive target → allocate is not effective in almost any real-world setting

An example: Market for lemons

George Akerlof, “The Market for Lemons: Quality Uncertainty and the Market Mechanism”

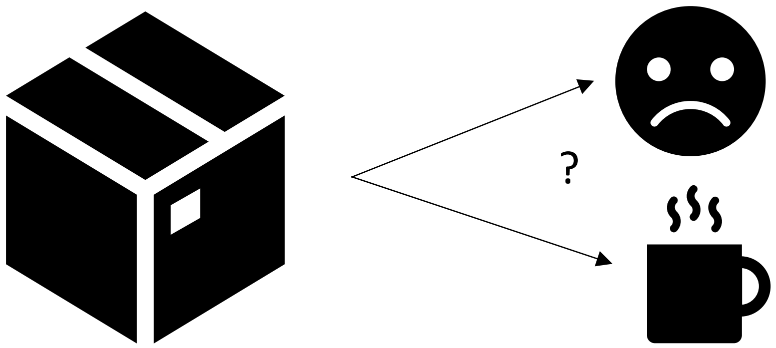
Main Idea – single seller

- Suppose I have a box

  With equal probability, either:

    - Nothing
    - Cup of coffee

  I know what's in the box, but you don't



- Coffee is worth

  $1 to me

  $1.50 to you

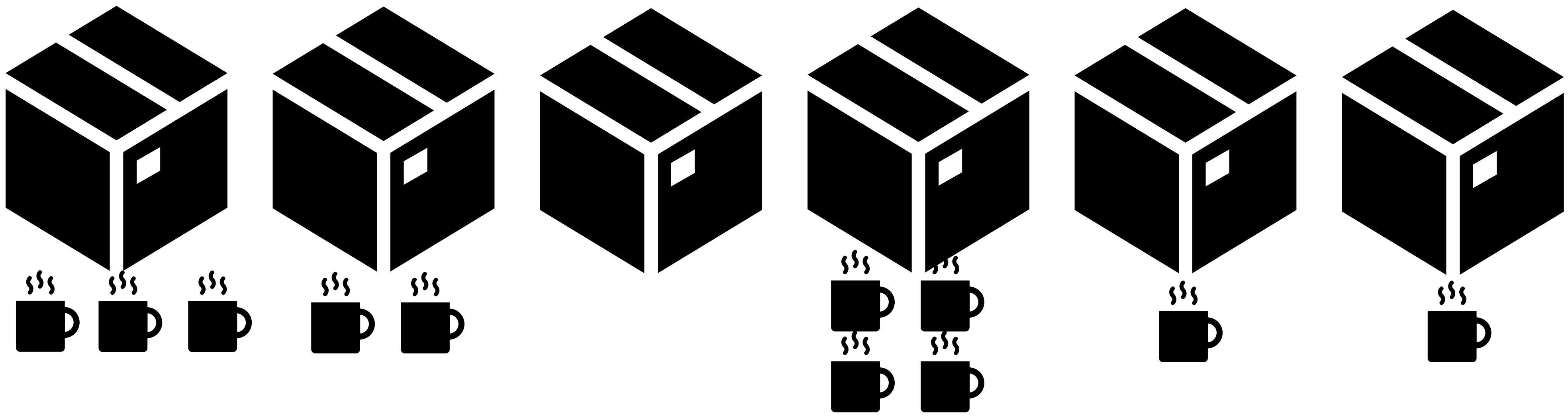
- Does a sale happen?

  If I can guarantee coffee, you pay me somewhere in [$1, $1.50] and we're both happy

  If I can't, you offer me $0.75…Do I say yes?

  Knowing the above, would you ever offer me $0.75?

Extending this to a market

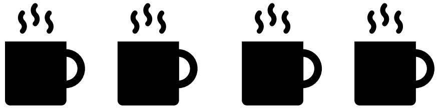


Random between 0 and 4 cups → you'd be willing to pay $3 (2*1.5)
→ No one with 4 cups would accept
Between 0 and 3 cups → you'd be willing to pay $2.25 (1.5*1.5)
→ No one with 3 cups would accept

…Eventually only the empty box is left, the "lemon"

Modern Example – Healthcare

Young & Healthy



Elderly and Sick

- What happens when insurers can’t price discriminate based on status?

Average prices go up…so young & healthy flee the market

Ok…so you *mandate* that everyone buys insurance, and hope the mandate is strong enough

Of course…healthcare is a bad example of a free market. Mandate was effectively repealed, with little apparent effect.

Main Ideas

Main concepts:

1. Information Asymmetry & Adverse Selection
2. Price reveals information → rational expectations
3. Markets can *unravel*; converge to and then stuck in non-Pareto efficient allocations
4. How can design help us here? What if we had a reputation system with repeat transactions? Or an independent auditor?

Markets for Lemons are everywhere, even/especially with machine learning

- Zillow's home-buying debacle shows how hard it is to use AI to value real estate | CNN Business
  - Bad machine learning? Or incentives/asymmetric information?

More involved example: Residency matching

The market failure

Similar *unraveling* to what we saw above:

- Hospitals (especially less glamorous ones) started hiring future residents earlier and earlier during medical school (~1940s)
- In response, other hospitals also started hiring earlier. The market unraveling, such that students were being offered residency positions before they knew what they wanted to do (this is clearly bad)
- Many other labor markets have seen the same unraveling [Roth and Xing (1994)]

Initial solution desiderata

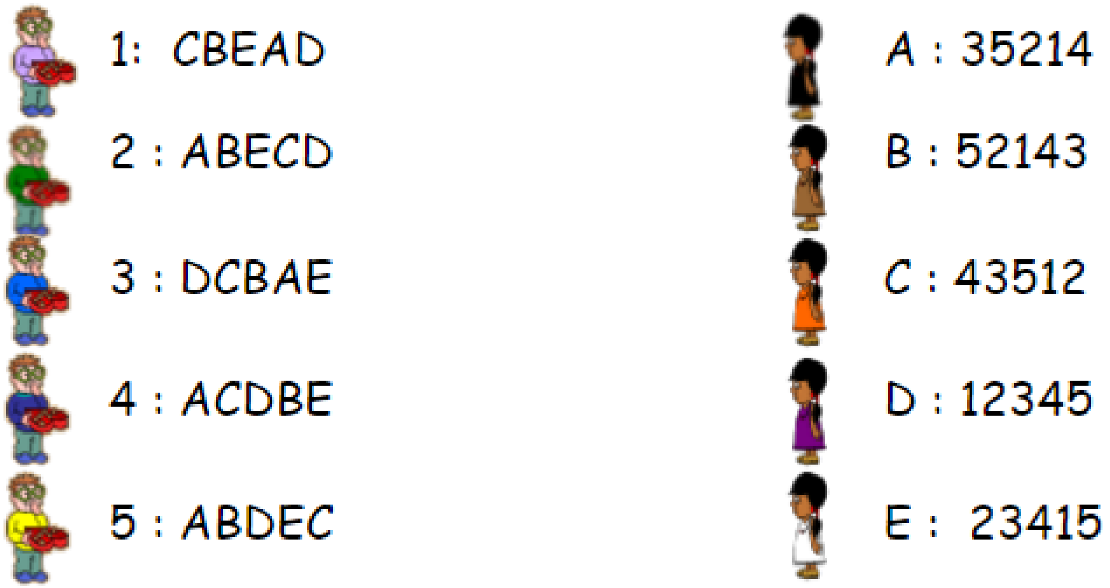
- In 1940s, a few attempts to fix the problem that didn't work well
- By 1951, a *centralized algorithm* would match doctors and hospitals

What are participants *action spaces*/what is the system *lever*?

- *Doctors* and *hospitals* each give ranked lists of the other side
  - Anyone can lie about their true preferences
- [Algorithm does the matching; assigns each doctor to a hospital]
- *Doctors* and *hospitals* can break their match
  - Suppose algorithm produces matches (A, X) and (B, Y). But Doctor A likes Y>X, and hospital Y likes A>B.
  - Then, (A,Y) can defect from the system and instead match with each other

Initial solution: Gale-Shapley in 1 slide

- Desiderata:
  - Doctors and Hospitals each submit *truthful* ranked lists
  - After the algorithm produces its solution, there is no “blocking pair”: there exists no (A,Y) that would want to leave their assigned partner for each other
- How it works:



Summary of theory of Gale-Shapley

- Theorem: A stable matching always exists [No “blocking pairs”]
- Theorem: Gale-Shapley algorithm finds a stable matching
- Theorem: Participants don’t want to lie about their preferences, *but only the “proposing” side.*

Problem solved? By 1970s, people started complaining again…

- Increase in *couples* in medical schools wanting to match together
- Implementation of algorithm favors hospitals? (Over doctors?)
- Concerns about strategic behavior by doctors

The redesign of the market (1990s)

Favoring doctors over favoring hospitals:

“Simple” change to algorithm: have doctors “propose” instead of hospitals. Rest of theoretical results hold. (And now, it’s strategy-proof for doctors!).

Couples: harder case -- Stable matching may not exist

BUT, some hope:

- Computational simulations suggest that in practice stable matchings almost always exist
- Follow up theory (e.g., Nguyen & Vohra 2018): “for any student preferences, we show that each instance of a matching problem has a "nearby" instance with a stable matching”

Problem solved forever?

Theory differs from practice in many ways

- Empirically, many doctors still mis-report their preferences. Why?
- The algorithm assumes people *know everything* about the other side, and can rank as many as they want
  - In practice, this is not true: you must *interview* at a school to even have a chance. So, if you only interview at top schools and don't get any…
  - Opposite problem: what if the same top 10 doctors interview everywhere (50 hospitals, each with 1 position)?
    - Evidence that this happened last few years due to Zoom interviewing
    - Recent proposal to have a centralized matching system for interviews: Explaining a Potential Interview Match for Graduate Medical Education | Journal of Graduate Medical Education (allenpress.com)
- Problems more severe in other places stable matching is used: e.g., school matching!

Lessons from residency matching

- Systems can fail due to seemingly minor incentives issues (causes bad equilibria)
- Designing even a limited system is hard and often politically tense
- Interplay between theory, empirical analysis, and simulation
  - "It turned out that the simple theory offered a surprisingly good guide to the design, and approximated the properties of the large, complex markets fairly well. Field and laboratory data showed that the static idea of stability went a long way towards predicting which kinds of clearinghouse could halt the dynamics of unraveling. And computation showed that many of the departures from the simple theory were small, and that some of the most severe problems that the counterexamples anticipated, such as the possibility that no stable matching would exist, were rare in large markets. Computation also revealed that large markets could achieve even nicer incentive properties than anticipated by the simple theory."
- Objective functions matter! (Doctor vs hospital optimal)
- People often don't behave "optimally," but that doesn't make theory useless
- Even if one component of the system is provably optimal, surrounding components (here, interview process) might undo benefits

Some nice quotes from Al Roth (2002)

- The largest lesson in all this is that design is important because markets don't always grow like weeds—some of them are hothouse orchids. Time and place have to be established, related goods need to be assembled, or related markets linked so that complementarities can be handled, incentive problems have to be overcome, etc. If game theory is going to be as important a part of design economics as it is a part of economic theory, we'll have to develop tools not just to develop conceptual insights from simple models, but also to understand how to deal with these complications of real markets.

- In the long term, the real test of our success will be not merely how well we understand the general principles that govern economic interactions, but how well we can bring this knowledge to bear on practical questions of microeconomic engineering. Just as chemical engineers are called upon not merely to understand the principles that govern chemical plants, but to design them, and just as physicians aim not merely to understand the biological causes of disease, but their treatment and prevention, a measure of the success of microeconomics will be the extent to which it becomes the source of practical advice, solidly grounded in well tested theory, on designing the institutions through which we interact with one another

So what has changed since 2002?

- Rise of online marketplaces (and more generally, computational systems) has made the type of work he described ubiquitous
- What’s difference? These applications are far “messier”
    - Faster (many times a second, as opposed to once a year)
    - Many complexities; not a ‘closed’ system. (in ride-hailing, pricing affects matching affects wages affects long term driver capacity…)
- The field is more mature, new methods, incorporation of data science
- We’re going to focus on these more “modern” applications and especially as it interfaces with prediction
- Critiques of approach, as enters more sensitive areas

Announcements

Paper discussion signup + instructions out soon

Next time: “Great ideas and papers” relevant to the area

Please read (some of):

- The Economist as Engineer: Game Theory, Experimentation, and Computation as Tools for Design Economics - Roth - 2002 – Econometrica
- Rittel and Webber. “Dilemmas in a General Theory of Planning” sympoetic.net/Managing_Complexity/complexity_files/1973 Rittel and Webber Wicked Problems.pdf