CSC2421

Mathematical Foundations of Algorithmic Fairness

Nisarg Shah

Introduction

- People
  - Instructor: Nisarg Shah (/~nisarg, nisarg@cs)
  - TA: Soroush Ebadian (soroush@cs)
- Info
  - Course Page: `cs.toronto.edu/~nisarg/teaching/2421s24/`
  - Discussion Board: `piazza.com/utoronto.ca/winter2024/csc2421`
- Meeting
  - Lectures: SK 720
  - Questions? Schedule 1-1 meeting by emailing me

What?

- Mathematical foundations of algorithmic fairness (duh!)
  - Primarily surveying impactful fairness definitions and provably fair algorithms from two literatures: *computational social choice and machine learning*

- We will focus on:
  - Concrete mathematical definitions for which we can prove non-trivial mathematical facts
  - Ideally, the definition should be satisfiable (at least to an approximate degree) but not trivially so (at least in conjunction with other basic properties)
  - Great if it is applicable to not just one problem, but a wide class of problems

What?

- We will consider a range of applications
  - Resource allocation
  - Voting & participatory budgeting
  - Matching
  - Classification
  - Clustering
  - Federated learning
  - Peer review
  - …

Learning Goals

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

1. Recognize and understand various mathematical notions of fairness across a range of algorithmic decision-making problems,
2. Reason about the relations between different fairness notions,
3. Prove or disprove that an algorithm satisfies a fairness notion,
4. Design algorithms satisfying given fairness notions or prove the impossibility thereof,
5. Understand societal implications of imposing (or not imposing) fairness.

Logistics

Optional Reference Textbooks

- Handbook of Computational Social Choice
  - By Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia
  - Online version available on the course webpage
- Fairness and Machine Learning
  - By Solon Barocas, Moritz Hardt and Arvind Narayanan
  - Online version referenced on the course webpage
- The Ethical Algorithm
  - By Aaron Roth and Michael Kearns

Grading Policy

- 2 assignments: 40%
- Final project: 50%
- Class participation: 10%

Assignments

- The course will be *highly theoretical*
  - The assignments will require deriving intricate mathematical proofs

- We will assume…
  - Strong familiarity with abstract reasoning and proof techniques
  - Adequate familiarity of CS concepts (e.g., algorithm design, worst-case approximation ratios, proof techniques)
  - Adequate familiarity of math concepts (e.g., probability, statistics, and a little bit of linear algebra + calculus)
  - No prior background in social choice or machine learning

Assignments

- Individual assignments
  - Free to discuss with classmates or read online material
  - Must write solutions in your own words
    - Easier if you do not take any pictures/notes from the discussions
  - Plagiarism will be dealt with strictly!

- Citation
  - For each question, you must cite the peer (write the name) or the online sources (provide links) referred, if any
  - Failing to do this is also plagiarism!

Other Policies

- "No Garbage" Policy
  - Borrowed from: Prof. Allan Borodin (citation!)

  1. Partial marks for viable approaches
  2. Zero marks if the answer makes no sense
  3. 20% marks if you admit to not knowing how to solve

- 20% > 0% !!

Course Project

- How?
  - Groups of 1-3
    - Larger groups are better
    - Find partners early, but maybe after the enrollment stabilizes
- What?
  - **Algorithmic:** Mathematical/empirical analysis of algorithms presented in class (or your own)
  - **Definitional:** Mathematical/empirical analysis of existing or novel fairness definitions
  - **Ideal projects** combine both mathematical and empirical analyses, and focus on both definitional and algorithmic aspects

Project Topic

- From your own research area of interest
  - We'll introduce various mathematical definitions of fairness, which you may be able to apply to your own research area
- From the course
  - I'll mention some open problems as we go along and later post sample project ideas
  - You can also explore problems we saw in class (or variants of them) in greater depth

Course Project: Timeline

- Find partners and think about a project idea
- Submission 1: Project proposal
  - Ideally 1 page but up to 2 pages excluding references
  - Outline of the idea, prior work, reasonable goals, ambitious goals
- Mid-project meetings
  - Optional, 1-1 with me, 30-minute
- Class presentations
- Submission 2: Final project report
  - Up to 5 pages excluding references and appendix
  - Focus on quality academic writing

Course Timeline

- (Approximate dates)
- ≈ Feb 25: HW1 due
- ≈ Mar 31: HW2 due
- ≈ March 1: Project proposals due
- ≈ Week of Mar 15: Mid-project check-in
- Last 1.5-2 lectures: Project presentations
- ≈ April 10: Project reports due

Introductions

Brief Introductions

- What to say?
  - Which program?
  - Which year?
  - Who are you working with (if any)?
  - What is your area of interest (if any)?
  - Anything else you'd like to share