Logistics and Overview

David S. Rosenberg

NYU: CDS

January 23, 2021

Contents

Logistics

Logistics

- Course webpage: https://brightspace.nyu.edu/d2l/home/26911
  - Syllabus on the website
- edstem: https://edstem.org/us/courses/4278/discussion/
  - **All class announcements via edstem**
  - Ask all questions on edstem
- Lecture Times
  - Wednesday: 5:20 - 6:20pm
  - 19 W 4th St Room 101

Course Staff

- TAs:
  - Yunzi (Alex) Ding
  - Weicheng (Jack) Zhu
- Course assistant: Metarya Ruparel

Lab Sections

- Alex and Jack lead lab sections on Thursday
  - Review, practice, and questions.
- Some online, some live – you should know your lab assignment.

Evaluation

- About 4 or 5 homeworks (50%)
- Weekly quizzes (20%)
  - Due every Friday
- Project (30%)
  - Groups of 3-4.
  - More information in a few weeks.

Knowledge Prerequisites

- Probability theory
- Basic ideas of statistics (confidence intervals, hypothesis testing)
- Machine Learning (DS-GA 1003 level)

Course was designed for people who have taken 1003, but most 1003 topics are not required knowledge.

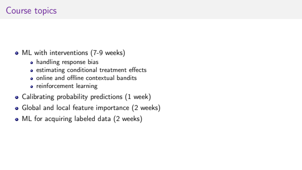
Course overview

Compared to DS-GA 1003

- DS-GA 1003 goes deep into
  - many methods for classification and regression
  - a bit of unsupervised learning
  - core techniques in ML
- In this class we try to address a more diverse collection of settings.
- We chose settings that have at least some overlap in the techniques used.

Course topics

- ML with interventions (7-9 weeks)
  - handling response bias
  - estimating conditional treatment effects
  - online and offline contextual bandits
  - reinforcement learning
- Calibrating probability predictions (1 week)
- Global and local feature importance (2 weeks)
- ML for acquiring labeled data (2 weeks)



This course deals with a range of topics that come up when applying machine learning in practice.

- Roughly half the course will cover topics connected to machine learning with interventions, such as counterfactual learning, reinforcement learning, and causal inference.
- Inverse propensity methods for handling biased samples and control variate methods for reducing variance will be given special attention, as these form a common thread of techniques relevant to each of these topics.
- We will also cover calibrating probability forecasts, interpreting machine learning models, active learning, crowdsourcing and "data programming", as time permits.

Response bias

- We field a survey asking who each person will vote for
- Get a low response rate
- Different types of people have different response rates
  - $\Longrightarrow$ Averaging respondents will be biased
- How to estimate the overall fraction of people who will vote for a candidate?
- Methods
  - inverse propensity weighting (IPW)
  - self-normalized IPW (SN-IPW)
  - regression imputation
  - doubly robust estimators

Response bias

- We field a survey asking who each person will vote for
- Get a low response rate
- Different types of people have different response rates
  - $\Longrightarrow$ Averaging respondents will be biased
- How to estimate the overall fraction of people who will vote for a candidate?
- Methods
  - inverse propensity weighting (IPW)
  - self-normalized IPW (SN-IPW)
  - regression imputation
  - doubly robust estimators

- There are certain challenging ideas and techniques that come up repeatedly in the first part of our course (in causal inference, counterfactual learning, and reinforcement learning).
- We will introduce them here in the simplest possible setting: estimating the mean of a population with a biased sample.

Randomized controled trials

- Simplest question we can ask about two **interventions**:
    - Which is better?
- In basic statistics class, we randomly assign treatment and control.
- Looking at the difference between groups,
    - we estimate the "average treatment effect" (ATE)
- How do we handle it if certain types of people are more likely to be assigned to treatment?
- What if there are **hetereogeneous treatment effects**?
    - i.e. different effects on different types of people?
- We'll discuss some very recent approaches to using machine learning models to
    - estimate hetereogeneous treatment effects

Randomized controled trials

- Simplest question we can ask about two **interventions**:
  - Which is better?
- In basic statistics class, we randomly assign treatment and control.
- Looking at the difference between groups,
  - we estimate the "average treatment effect" (ATE)
- How do we handle it if certain types of people are more likely to be assigned to treatment?
- What if there are **hetereogeneous treatment effects**?
  - i.e. different effects on different types of people?
- We'll discuss some very recent approaches to using machine learning models to
  - estimate hetereogeneous treatment effects

- When machine learning is applied in practice, it is often used to guide **interventions** in the world that we hope will improve some outcome measure.
- When we start making interventions, one of the most basic questions we can ask is which of two interventions (such as a treatment and a control) is better.
- In a basic statistics class, we learn how to estimate the "average treatment effect" (ATE) when individuals are assigned to a treatment or control group with equal probability.
- In this module, we discuss how to estimate the ATE when individuals are assigned to interventions with probabilities that depend on covariates (i.e. characteristics/features of the individuals).
- Beyond that, interventions may have better or worse performance depending on characteristics of the individuals. We will also discuss how to estimate these "conditional average treatment effects".

Exploration vs exploitation for bandits

- Suppose we have an intervention that seems to work well
  - e.g. suggesting comedy movies to user X
- Can we balance "exploiting" that intervention with "exploring" new interventions?
  - e.g. suggesting action movies
- We'll introduce a new problem setting: **bandits**
  - A bandit problem is one where you only get feedback on the intervention you take
  - No feedback or label that tells you what the "best" intervention would have been
- We'll study various approaches to this explore/exploit problem in the bandit setting.

Exploration vs exploitation for bandits

- Suppose we have an intervention that seems to work well
  - e.g. suggesting comedy movies to user X
- Can we balance "exploiting" that intervention with "exploring" new interventions?
  - e.g. suggesting action movies
- We'll introduce a new problem setting: **bandits**
  - A bandit problem is one where you only get feedback on the intervention you take
  - No feedback or label that tells you what the "best" intervention would have been
- We'll study various approaches to this explore/exploit problem in the bandit setting.

- How can we balance "exploiting" interventions that worked well before (e.g. suggesting comedy movies for a particular individual) with "exploring" new intervention strategies (e.g. suggesting action movies) that may have better outcomes?
- In this module, we explore approaches to this classic "explore/exploit" problem.
- We will start with a focus on the simple "Bernoulli bandit" setting.
- Then we will introduce the more general contextual bandit setting, and discuss explore/exploit methods for that case as well.
- We'll see how control variate techniques similar to those we used for response bias can help reduce variance in this setting as well.

Counterfactual policy evaluation

- Different interventions are preferable for different "contexts"
  - A context could be an individual at a particular time of day
- We want a **policy** that assigns the optimal intervention for each context
  - depending on features of the context
- We can compare two policies with an **A/B test**
  - basically means deploying the two policies and seeing how they do
- A/B tests can be costly in various ways... (e.g. bad recommendations can lose customers)
- In this module, we show how we can
  - estimate the performance of a new policy without actually deploying it
  - using data that was collected with the policy that's already deployed (the **logging policy**)

Counterfactual policy evaluation

- Different interventions are preferable for different "contexts"
  - A context could be an individual at a particular time of day
- We want a **policy** that assigns the optimal intervention for each context
  - depending on features of the context
- We can compare two policies with an **A/B test**
  - basically means deploying the two policies and seeing how they do
- A/B tests can be costly in various ways... (e.g. bad recommendations can lose customers)
- In this module, we show how we can
  - estimate the performance of a new policy without actually deploying it
  - using data that was collected with the policy that's already deployed (the **logging policy**)

For A/B testing – not only can it be dangerous or costly to deploy a suboptimal policy (think about self-driving cars or a medical setting, or an online shopping setting), there's also a practical limit to how many policies we can test out and still get a reasonable estimate of the performance of each.

Counterfactual learning

- Something we'll learn about counterfactual policy evaluation:
  - The more "different" the policy we're evaluating is from the logging policy, the less certainty we'll have about the evaluation.
- How can we account for this uncertainty when **learning** a new policy from logged data.
- We can apply all the methods we've developed so far: IPW, SN-IPW, regression imputation, doubly robust estimation.

Introduction to reinforcement learning

- In the bandit setting, we assume contexts are i.i.d.
- In the reinforcement learning setting,
  - sequences of contexts are grouped together into **episodes**
  - actions we take at one step in the episode may affect the next context we observe
- In this module, we study "policy gradient" approaches for learning policies in this setting
  - REINFORCE
  - possibly some actor-critic methods

Calibrated probability predictions

- Suppose we have a model that makes probabilistic predictions
- How can we ensure that they are both **calibrated**
  - i.e. the "70%" outcomes actually occur 70% of the time
- and **sharp**
  - i.e. the probability predicted varies appropriately depending on the input features
- It turns out, even assessing whether a model is calibrated is nontrivial
- We discuss classic and modern approaches to calibration and to assessing calibration

Feature importance

- A popular topic... but what does it even mean?
- We'll discuss
  - Permutation feature importance
  - Partial dependency plots
  - Individual conditional expectation
  - Issues with all the above methods

Explaining model predictions / local feature importance

- SHAP is all the rage now for explaining model predictions
- What is it? How does it work? What does it really tell us?
- There's plenty of debate about SHAP and we'll discuss some of this too
- Also we'll look at "Local Interpretable Model-agnostic Explanations" (LIME)

Crowdsourcing and answer aggregation

- For many real world ML problems, a major expense is
  - collection of labeled data
- In this module, we discuss how we can use "crowd workers"
  - generally non-expert, and with varying error rates
- to generate reasonably reliable labels for our data.
- How many crowd workers should we get to label each example?
- How do we automatically resolve disagreements?
- Application to aggregating predictions from expert-generated **rules** (e.g. SNORKEL)

Active learning

- The **active learning** problem is the following:
  - Given a large pool of unlabeled examples, and
  - a finite budget for labeling these examples,
  - can we do better than randomly sampling unlabeled examples to be labeled?
- We'll discuss some classic approaches as well as some refinements