DATA 37200: Learning, Decisions, and Limits
(Winter 2026)

Lecture 1: Intro and the First Problem

Instructor: Frederic Koehler

Outline

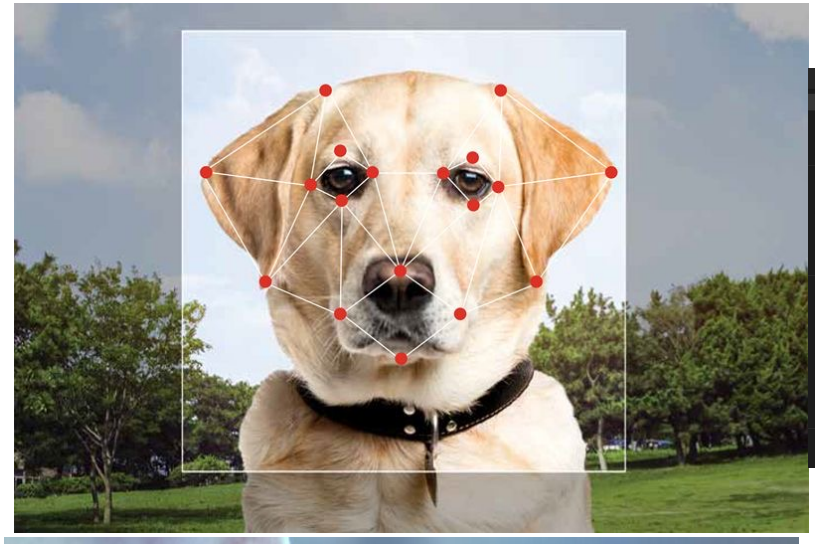
- Course Overview
- Administrivia
- The First Problem

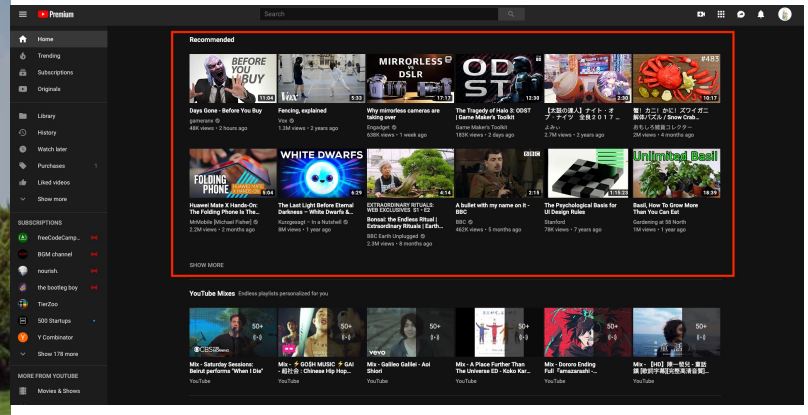
Credit

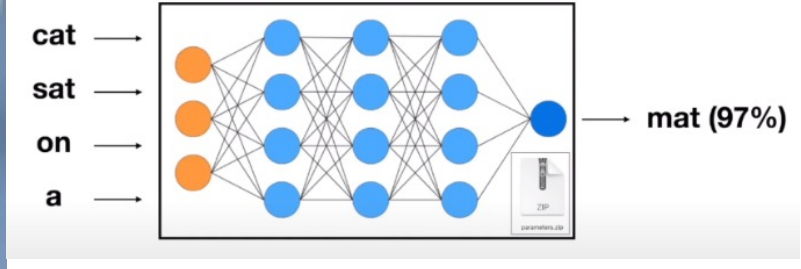
- This course was created last year by Haifeng Xu & myself (Frederic)
  - This year's course is broadly similar to last year's
  - Not identical. In particular, grading/coursework has been changed. (Should be streamlined.)
- The content overlaps and draws upon different fields such as (e.g.) computer science, information theory, statistics, control theory, ...
- Let's start with some quick motivation & overview.

Recall: Classic Machine Learning Problems

Preference learning for recommendations, Image recognition, Next token/word prediction, Speech recognition



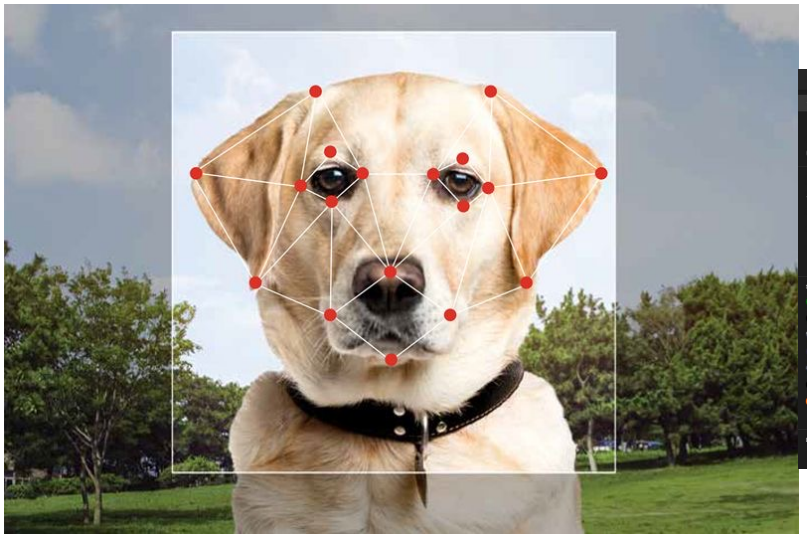


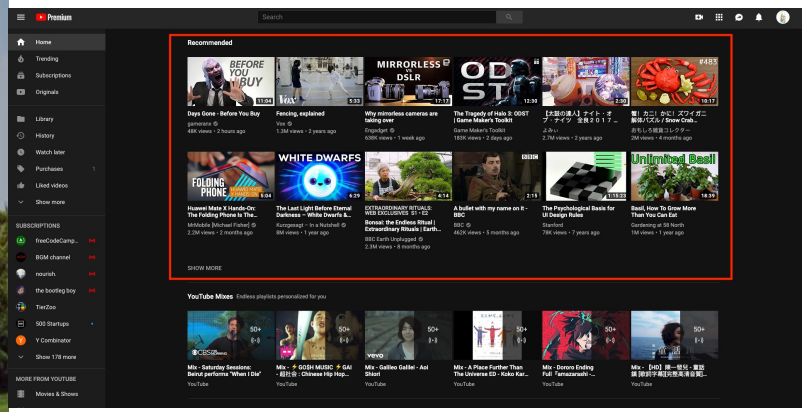


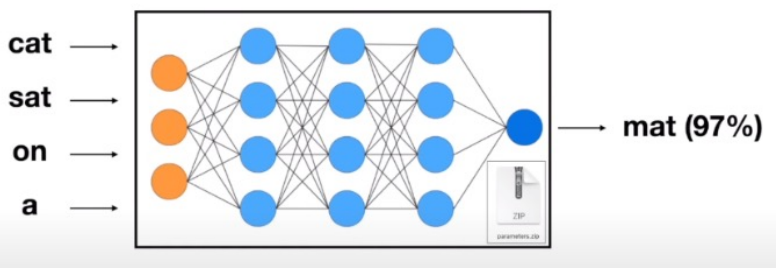
Recall: Classic Machine Learning Problems

These are recognition-based learning problems

- Task environments are usually static
- Often use supervised learning
- Relatively mature by now, and quite successful in both theory and applications





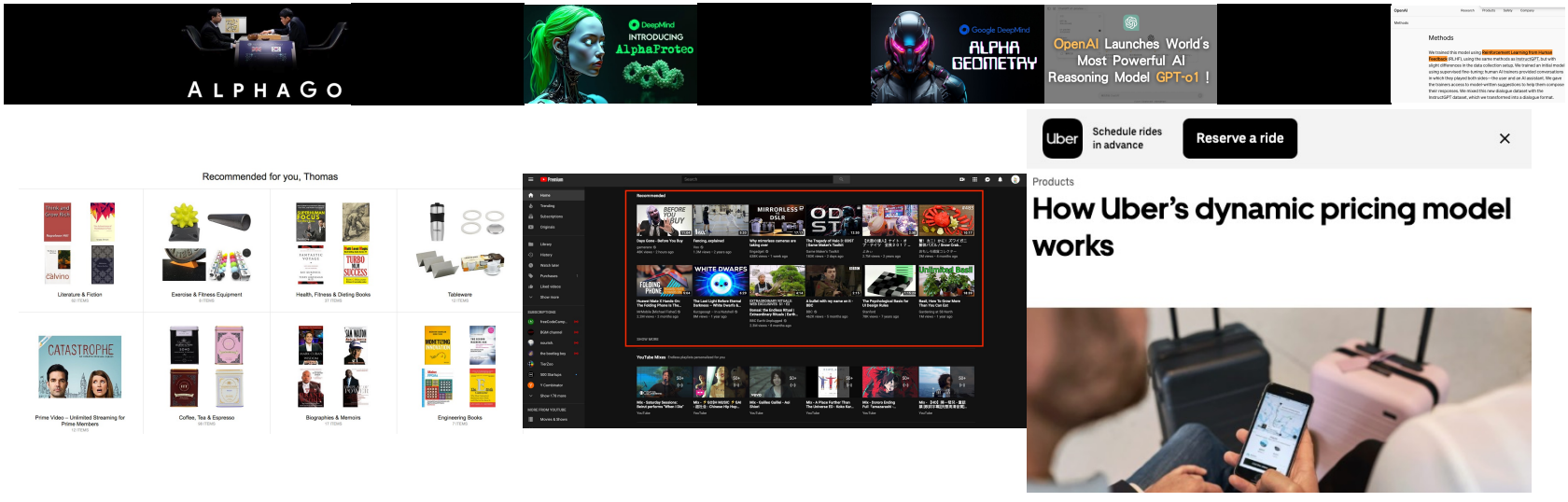


This Course: Decision-Based Learning Tasks

- Often use quite different design principles and learning techniques
  - Will see in our first learning problem why new design ideas are necessary
  - A well-known field studying this is reinforcement learning (RL), about which this course will cover a lot, though also beyond
  - Problems are often more complex
- Why more complex? To learn decisions, we have to consider many factors beyond just accuracy:
  - Rewards/payoffs/costs/utilities
  - Decision consequences – your learned decisions act on (hence change) the environments
  - Conflicting interests/incentives
  - Societal issues: fairness, alignment, welfare-efficient...

Why Important?

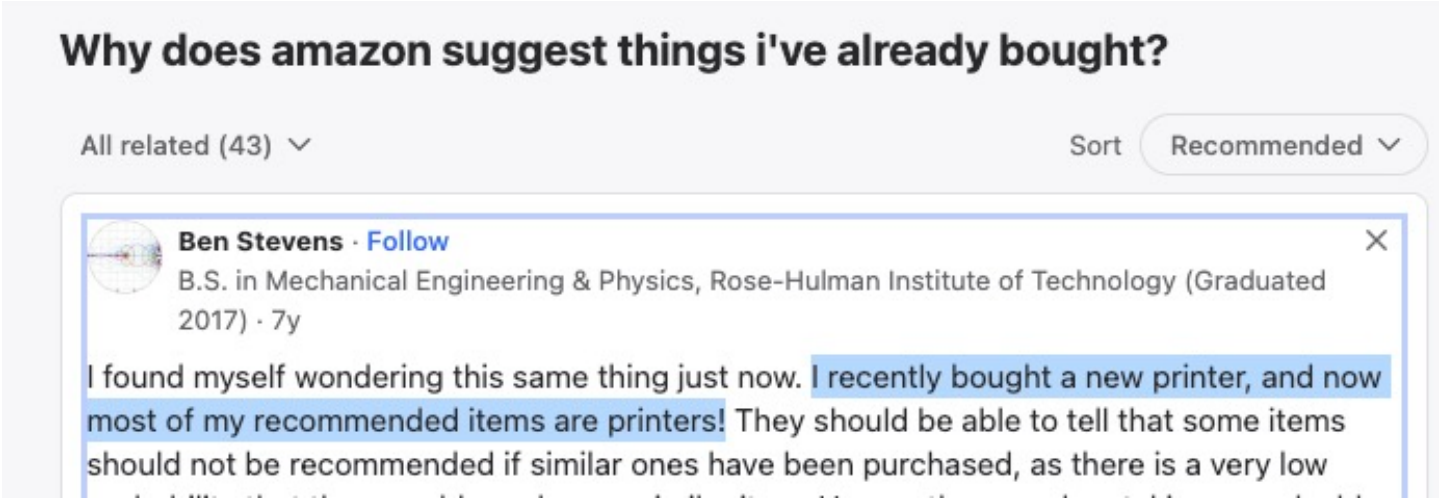
- Core decision-based learning techniques are underlying many breakthrough research and billions$-scale industrial applications

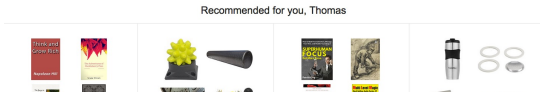


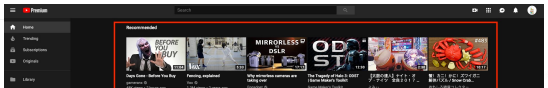
Why Important?

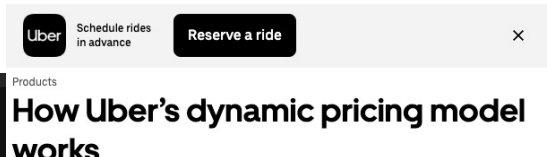
- Core decision-based learning techniques are underlying many breakthrough research and billions$-scale industrial applications

Challenge: demand/supply a price a changes
demand/supply
Recommendation needs consider its action consequences
Dynamic pricing based on
Product/content recommendation
traffic/supply/demand prediction





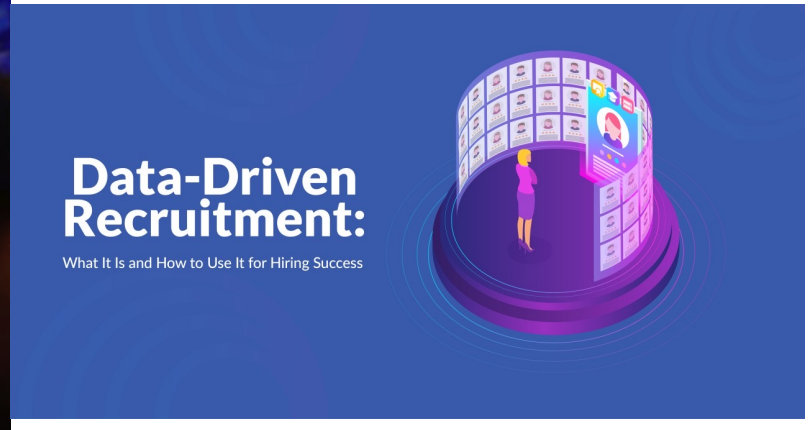




Why Important?

- Core decision-based learning techniques are underlying many breakthrough research and billions$-scale industrial applications

Not to mention many data-driven policy/decision making problems in critical societal, health and security applications



Wow, cool! So... after this course, will I become the hero to work towards Nobel, or solving Google's/Amazon's problems?

- Not immediately...
  - Those are not easy problems to solve
  - This is designed to be a foundational (theory-focused) course
  - (Programming/implementation is also important, just not our focus)
- Goal of this course is to build your foundational understandings about
  - What key factors to consider while learning optimal decisions
  - Basic design principles of optimal learning algorithms
  - What is possible, and what is not possible
  - Along the way, also enrich your statistical and algorithmic toolkits

Learning Objectives

- Understand how to mathematically formulate and analyze models for interactive learning problems; learn how to apply core techniques from probability, statistics, optimization, etc.
- Understand key difficulties/challenges with solving RL problems
- Understand basic principles underlying relevant cutting-edge technologies, such as Reinforcement Learning from Human Feedback (RLHF), AlphaGo training, ...
- Be well-prepared to understand state-of-the-art papers about online learning, RL and data-driven decision making
- Have the foundations to work on relevant practical applications

Tentative Topics of the Course

- Stochastic bandits, martingale concentration
- Basics of information-theoretic limits, KL divergence
- Online versions of prediction/regression, online optimization, connections to game theory
- Contextual bandits (context-driven decision making)
- Fundamentals of reinforcement learning: Markov decision processes, finite state space guarantees, probably a little bit of control theory
- A bit about modern developments such as multiplayer RL, RLHF, ...

Targeted Audience of This Course

- Anyone planning to do research in machine learning (theoretical or empirical)
  - The course is theory-focused, but largely focused on shared "fundamentals" which should also benefit applied researchers
  - *Does* require some amount of "mathematical maturity"
  - Even if you do not work on interactive decision learning, it is part of the basic ML toolkit.
- Anyone who wants to grasp the basics about how ML can be used for recommendation, preference alignment, dynamic pricing, etc.
- Anyone who want to see what other useful ML paradigms there are beyond supervised learning via large neural networks
  - Offer you a more comprehensive view about machine learning
  - Deep learning is super useful and powerful, but real industrial success also crucially hinges on other equally critical techniques

Outline

- Course Overview
- Administrivia
- The First Problem

Basic Information

- Course time: Tue/Thu, 11:00–12:20 pm at JCL 011
- Lecture: in person (unless further instruction)
- Instructor: Frederic Koehler (fkoehler@uchicago.edu)
- Office Hour: Frederic (Wed 3:30-4:30 pm, tentatively)
- TA: Joonhyung Shin (joons@uchicago.edu)
- Couse website: https://frkoehle.github.io/data37200-w2026/
- Easy to find from my webpage.
- References: linked papers/notes on website, no official textbooks, materials from last year.
- Slides will be posted after lectures

Prerequisites

- Mathematically mature: be comfortable with proofs
- Sufficient exposures to probabilities and algorithms/optimization
- Algorithms (CMSC 27200/27220 or equivalent)
- Linear algebra (CMSC 25300 or equivalent)
- Probability (STATS 25100 or equivalent).
- If not sure, consult with the instructor. Note that no background on learning theory is required.

Requirements and Grading

- Part I (30%): Around 3 mostly proof/theory-based assignments (may have some light empirical component)
- Part II (70%): Midterm and Final
- Midterm: tentatively in class on Thursday Feb 5. Please let me know ASAP if you cannot make the midterm!
- Split: better of (30%,40%) and (20,50%). So you can make up for weak midterm grade with strong final.

Notes on Relevant Materials

- There are courses (and blogs) online that overlap with materials of this course
- These are great resources for extra reading, but it is still very useful for you to follow lectures as closely as you can because
  - Different instructors interpret the same knowledge differently
  - This will shape your way of thinking differently, which we think is the most valuable thing to learn from a course

If you have any suggestions/comments/concerns,
feel free to email me.

Outline

- Course Overview
- Administrivia
- The First Problem

The Stochastic Multi-Armed Bandit Problem

- Named after a gambling game
- A foundational RL problem with a simple and elegant formulation

Stochastic Multi-Armed Bandit Problem

Formulation of the Multi-Armed Bandit (MAB)

- A set of k arms, denoted as k= $\{1,2, \cdots, k\}$
- Pulling arm i once generates a random reward r_i drawn from distribution D_i

Useful notations: let $\mu_i = \mathbb{E}[R_i]$ and $\mu^* = \max \mu_i$

- As the algorithm designer, you decide which arm to pull to maximize your expected reward.
- Nice to phrase in terms of *minimizing regret*

$$Regret = \sum_{t=1}^{T} (\mu_i^* - r_{i(t)})$$

where $i(t)$ is the arm we chose at round t.

Stochastic Multi-Armed Bandit Problem

Question: if you know $D_1, \ldots, D_k$ (or even just $\mu_1, \ldots, \mu_k$), what would be your optimal strategy?
Ans: always pull the arm $\arg\max_i \mu_i$, with expected reward μ^*.
This achieves maximum possible expected reward (minimizes expected regret).

- Challenges arise when we do not know μ_i, and need to learn from samples of D_i
- This leads to formulation of the MAB problem

The Stochastic Multi-Armed Bandit Problem

Why this is a learning problem?

- Do not know μ_is in advance, hence need to learn them

Why this is not just a learning problem?

- Likely we need to learn μ_i to some extent, but that's not final goal
- It is possible to achieve very high reward without needing to learn every μ_i well
- Btw, this makes a lot of sense in real life – we find effective ways to do things with needing to fail a lot at every other alternative

A Little History of MAB

- This is a very clean and elegant problem
- Despite "bandit" in its name, MAB was initially motivated by designing reward-maximizing clinic trials, where an arm = a medical treatment
  - Started by William R. Thompson in 1930s whose designed the first algorithm for MAB, now called "Thompson Sampling"
  - Modern formulation due to Robbins: "Some aspects of the Sequential Design of Experiments" 1952
- Extensively studied in the past few decades
  - Many different variants considered due to different application considerations (e.g. change over time)
  - MAB is a nice special case of RL
- Has really a lot of applications, even in many of today's real systems

Next: Concentration Inequalities
"A nonasymptotic theory of independence" [BLM '13]

Balancing Reward and Risk is Crucial in Decisions

- In many real decision-making problems, we only receive random rewards, but optimal decisions depends on underlying expected reward
- MAB is such an example; so is buying stocks
- Sample average (also called empirical mean) is a good proxy of true mean, but not always accurate (i.e., some risk that true mean is fairly different from empirical mean)
- Intuitively, the more samples, typically the closer empirical mean is to true mean (thus less risk)

We want a rigorous quantitative statement for the above intuition! This will be provided by *martingale concentration inequalities.* (Essentially, the probability theory of iterated betting games.)

Warmup: a "dumb"/suboptimal strategy

- Naive solution to explore-exploit tradeoff: pure exploration followed by pure exploitation.
- NAIVE-EE: (1) Pull each arm m times, (2) For the rest of time, pull the arm with the largest sample mean.
- Achieves nontrivial but suboptimal regret bound.
- Good to analyze as a warmup...

LLN, CLT, Concentration

- NAIVE-EE: (1) Pull each arm m times, (2) For the rest of time, pull the arm with the largest sample mean.
- Suppose $m \to \infty$ and $m = o(T)$. Then by the *law of large numbers* $\hat{\mu}_i \to \mu_i$
- So this achieves $o(T)$ regret.
- We can compare different strategies in terms of the *rate* at which their average regret goes to zero. E.g. $1/\log(T)$ goes to zero much slower than $1/T$.
- We will analyze the regret more precisely using *concentration inequalities* (think: one kind of finite sample version of the Central Limit Theorem).

Recall: Chebyshev inequality

If $X_1, \ldots, X_n$ are independent with mean μ then

$$\Pr(|\sum_i X_i - \sum_i \mathbb{E}X_i| > t) \leq \frac{\sum_i Var(X_i)}{t^2}.$$

Example: if $X_1, \ldots, X_n$ are valued in $[0,1]$ then $Var(X_i) \leq 1/4$ (why?) and so $Var(\sum_i X_i) \leq n/4$. I.e. standard deviation is $\sqrt{n}/2$.

Suboptimal dependence on t in many cases. (Compare to CLT.) This will matter a lot in bandits (and other) applications. Let's see something more powerful.

Hoeffding's inequality (bounded independent sums)

Theorem (Hoeffding)

Let $X_1, \ldots, X_n$ be independent random variables with $X_i \in [a_i, b_i]$ almost surely. Let $S_n = \sum_{i=1}^n X_i$ and $\mu = \mathbb{E}S_n = \sum_{i=1}^n \mathbb{E}X_i$. Then for every $t > 0$,

$$\Pr\big(S_n - \mu \geq t\big) \;\leq\; \exp\Big(-\frac{2t^2}{\sum_{i=1}^n (b_i - a_i)^2}\Big)$$

and

$$\Pr\big(S_n - \mu \leq -t\big) \;\leq\; \exp\Big(-\frac{2t^2}{\sum_{i=1}^n (b_i - a_i)^2}\Big).$$

So

$$\Pr\big(|S_n - \mu| \geq t\big) \;\leq\; 2\exp\Big(-\frac{2t^2}{\sum_{i=1}^n (b_i - a_i)^2}\Big).$$

Special case: if $X_i \in [0,1]$ then $\Pr(S_n - \mu \geq t) \leq \exp(-2t^2/n)$.