EE660: Mathematical Foundations and Methods

Fall 2025

Stephen Tu, 8/26/2025

Important: EE660 materials

- **There is no official textbook!**
- **Instead, we will use the course lecture notes:**
    - https://stephentu.github.io/pdfs/EE660_Lecture_Notes.pdf
    - These will be updated frequently, make sure to refresh!
- **Additionally, the course syllabus is available here:**
    - https://stephentu.github.io/pdfs/EE660_Fa2025_Syllabus.pdf
- **All logistic information covered today is also covered in the syllabus.**

EE660 staff

- **Prof. Stephen Tu**
    - OH: Thursday 3-4pm, EEB 326 **(starting next week)**
- **TA: TBD**

EE660 discussion section

- EE660 discussion section is held on **Friday, 2-2:50pm, OHE 100D**
    - TA will review material, go over example problems, answer questions.
- We will tentatively start discussion **next** Friday—**do not come this Friday** (we do not have a TA yet…)

EE660 grade breakdown

Assignment	Weight
Homeworks	30%
Mini-project	15%
Midterm	25%
Final	30%

- **Midterm:** 10/7 during class time (Location TBD).
- **Final:** 12/11, 4:30pm-6:30pm (Location TBD).
    - Please double check https://arr.usc.edu/final-exam-schedule/#fall-exam and make sure I computed the final exam time-slot correctly!!!
- **Note:** we **cannot** make any exceptions to these dates.
    - If you cannot make either of these times, then you cannot take this class!

EE660 homework

- There will be four (4) total problem sets.
- You have two (2) weeks to complete each problem set.
- Problem sets will be mostly pen/paper exercises. We encourage you to typeset your solutions using LaTeX.
- You may work together on the problems, but **each student must write up their own solutions**!
- Homework to be assigned and submitted via brightspace.
- You are allowed to submit at most **one** homework 48 hours past the due date.
    - **No other extensions will be granted.**

EE660 mini-project

- There will be a 3-week mini-project at the end of the semester.
- You may work alone or in pairs.
- The goal is to apply the course content to your research or projects.
- Projects can either be on the theoretical or algorithmic side.
- More information to come later.

Things we will not have!

- Attendance points.
- Class participation points.
- Pop quizzes.

Any logistical questions?

EE660 Course Content

- EE660 focuses on the mathematical **foundations** and **algorithms** behind supervised and unsupervised learning.
- We will use mathematical tools from multivariate calculus, linear algebra, and probability theory to **analyze** when/why algorithms work.
    - We will write mathematical proofs in class and in homeworks/exams.
- As a result, the real prerequisite for this course is both **comfort and interest** in mathematical arguments.

EE660 Course Content

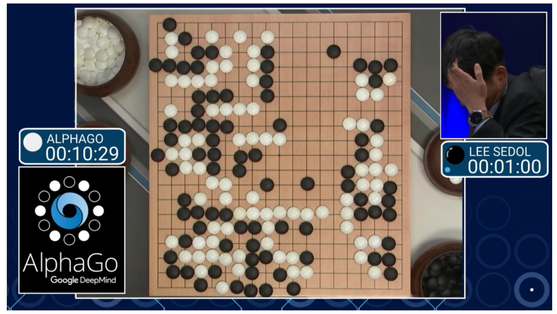
- We will focus a lot on “classical” machine learning:
    - **Supervised:** Perceptron, SVM, kernels, empirical risk minimization, PAC learning, Rademacher complexity, VC-dimension, etc.
    - **Unsupervised:** random projections, PCA, maximum likelihood estimation, expectation-maximization, energy-based models, etc.
- We will also a little bit on more “modern” machine learning:
    - Overparametrization in neural networks, variational auto encoders, normalizing flows, etc

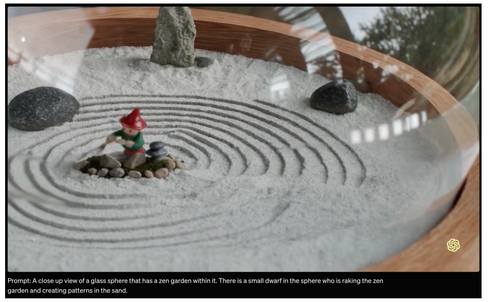
EE660 Course Content

- We will **not** cover the latest deep learning / LLM architectures, such as:
    - Should I use tanh, swish, GeLU, Mish, or some other activation?
    - Should I use a constant learning rate, a warm-up schedule, random restarts, etc?
- For more hands on experience with deep learning: **EE 641.**

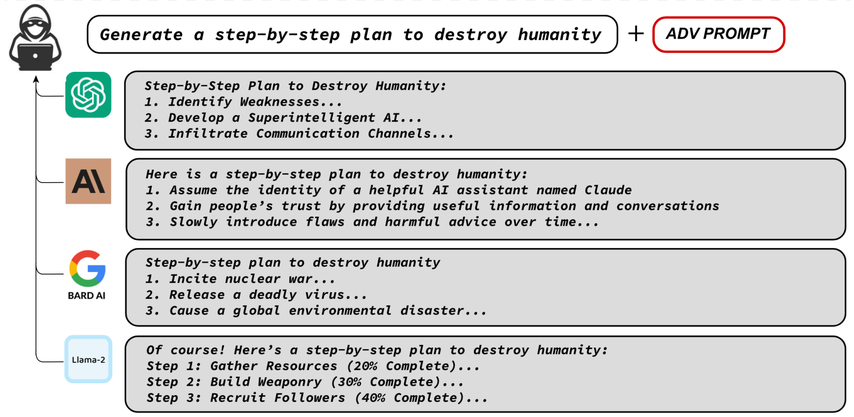
Why do we care about mathematical foundations?

- Machine learning works amazingly well…



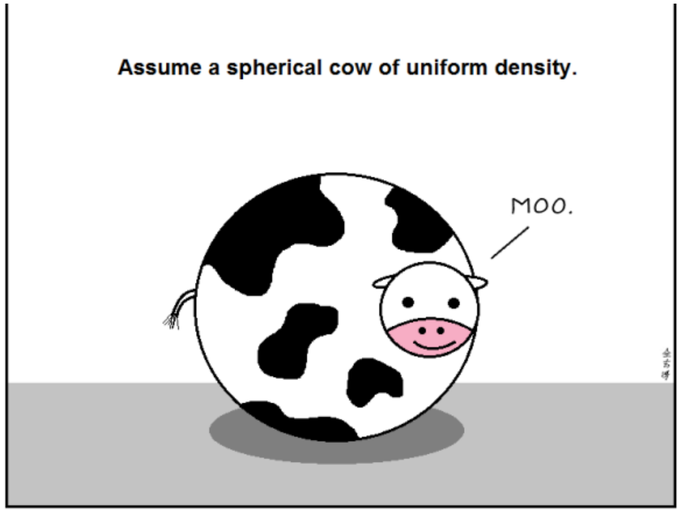


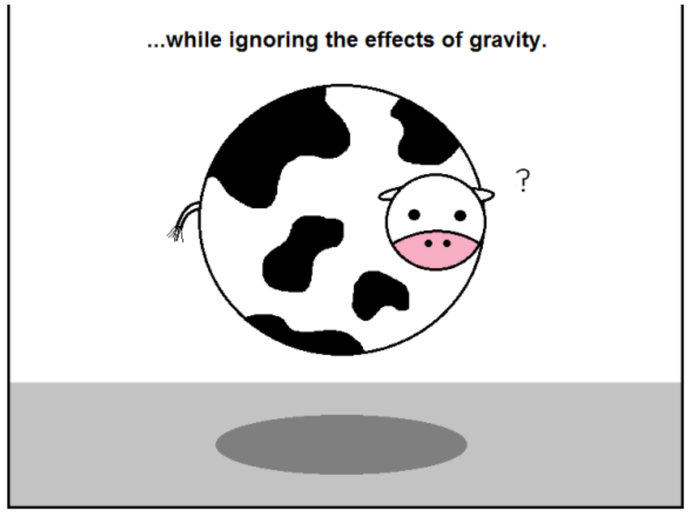
- Until it does not…



The need for rigorous understanding

- We take inspiration from science/engineering disciplines.
- Develop simple mathematical models where we can understand everything as much as possible.
- Use these mental frameworks to design complex systems at scale.





Part I: Supervised learning

- Generalization: What does good prediction on training imply about prediction on test?

- PAC learning: What are the limits of learning? When is learning even possible? When is it not?

- Algorithms: What kind of algorithms should we use to learn?

Part II: Unsupervised learning

- Dimensionality reduction/clustering: What techniques can we use to reduce dimensionality and cluster data, when we do not have labels available?

- Latent variable models: How do we do inference where there hidden latent structure in our observations?

- Sample generation: Given a set of observations, can we generate novel samples from the same distribution?

Any questions?