Basic Information

Course: EE660: Mathematical Foundations and Methods of Machine Learning
Term: Fall 2025
Instructor: Stephen Tu (stephen.tu@usc.edu)
TAs: Renyan Sun (renyansu@usc.edu)

Please include [EE660] in the subject of any email related to the course.

Time and Location:

Lecture
Tuesday and Thursday, 4:00-5:50pm, OHE 100D.

Discussion
Friday, 2:00-2:50pm, OHE 100D.

Midterm:
10/7, Tuesday, 4:00-5:50pm, Location TBD.

Final:
12/11, Thursday, 4:30-6:30pm, Location TBD.

Office Hours:
Stephen: Thursday, 3:00-4:00pm, EEB 326 and Zoom (Hybrid)
Renyan: Monday, 1:50pm-2:50pm, PHE 320 and Zoom (Hybrid)

Course Prereqs: EE 503, EE 510, and EE 559

Course Description

This course will cover the mathematical foundations of machine learning, including both supervised/unsupervised learning. The focus will be on the mathematical principles underlying machine learning algorithms. Topics covered include (but are not limited to) Perceptron, support vector machines, kernel methods, random features, concentration inequalities, empirical process theory, VC-dimension, stochastic gradient methods, nearest neighbors classification, universal approximation for neural networks, principle component analysis, maximum likelihood estimation, latent variable models, expectation-maximization, energy-based models, score matching, variational autoencoders, normalizing flows. Students will leave this course with a rigorous understanding of classic machine learning methods.

Textbook

There is no textbook for the course. Instead the reading will be from the course lecture notes.

Note: I will frequently update the lecture notes as the course progresses (filling in new content as we move forward in addition to correcting typos). While I will update the DEN with newer versions of the notes, I recommend instead using the link (https://stephentu.github.io/pdfs/EE660_Lecture_Notes.pdf) and refreshing each time you go to read.

Furthermore, as these notes are a work in progress, please feel free to send me an email if you suspect any typos or errors.

Homeworks

There will be 4 homeworks assignments throughout the semester. These homeworks will mostly be pen and paper exercises, with an occasional coding exercise. For any coding exercises, you are free to use whatever language you choose (although we highly encourage the use of Python and its comprehensive scientific computing ecosystem).

The main deliverable will be a PDF file containing your solutions. **It is heavily encouraged that you use LaTeX to typeset your homework (a sample LaTeX template will be provided before the first homework is assigned)!**

Collaboration Policy: You are free to collaborate with other students on the homework assignments. **However, all solutions must be written by yourself.**

Late Policy: You will be allowed to submit one (1) homework at most 48 hours past the due date with no penalty. **This does not apply to the mini-project.**

Mini-project

There will also be a mini-project to be completed over the last month of the course. The mini-project will be open-ended, and allow students to apply the material from the course to their research or other interests. The course staff will also provide a few sample project ideas. The project may be more theoretical in nature, e.g., extending a derivation from the course, or applied, e.g., implementing a generative modeling algorithm for a specific problem instance. The main deliverable for the project will be a project report. More details will be provided as the project is assigned.

Grading breakdown

The following table describes how your final grade will be calculated.

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

AI Chatbot Policy

USC's official stance, as outlined by the Academic Senate, is that each course may determine its own policy on the acceptable use of AI assistants. Accordingly, we outline EE660's policy below. For simplicity, we use ChatGPT as a generic term to refer to AI chatbots such as ChatGPT, Gemini, Claude, Grok, and others.

First, we would like to acknowledge that a flat-out banning of ChatGPT is not only impossible to implement, but also perhaps not a very forward-looking policy. ChatGPT can in fact help students to, for example, better understand the lecture notes by summarization, re-phrasing, coming up with other examples, etc. It can also be used to test your own understanding of a subject/topic, by generating novel practice problems, or to answer specific parts in a derivation from a lecture you are unclear about. While we caution students against heavily relying on ChatGPT (it does still hallucinate facts and can be misleading/inaccurate in its responses), we do see the merit in using ChatGPT as an aid (not a substitute though!) for learning. Hence, the general use of ChatGPT as an additional supplement to lectures, discussions, office hours for learning the course material is permitted.

On the other hand, **ChatGPT is not allowed to be used to directly answer homework problems and/or complete the mini-project.** This does not just include copying and pasting ChatGPT responses—taking ChatGPT's response and re-wording it to look like your own work is not permitted as well. **If we find reasonable evidence that your homework and/or mini-project was completed by ChatGPT, then you will receive an automatic zero for that assignment.** If you are stuck on a particular homework problem, please engage with the course staff through office hours so we can help you get unstuck. Remember, both the midterm and final exam are to be taken in person with formal proctoring. Even if hypothetically you manage to use ChatGPT without getting caught and obtain a perfect grade on both the homeworks and the mini-project, that only accounts for 45% of the total grade. The remaining 55% depends on your performance on proctored exams.

One exception to the use of ChatGPT for homeworks/mini-project is in formatting written documents. You may use ChatGPT to assist with document preparation—e.g., generating a

LaTeX template, fixing LaTeX compilation errors, formatting tables, or typesetting equations. Be extra cautious however to ensure that this usage is restricted to how the content that you created is presented, and does not cross over into ChatGPT generating the actual content.

If you have any specific questions about valid ChatGPT usage, please ask the course staff.

Topics

Supervised Learning

- Perceptron
  - Mistake bound.
  - Generalization bound.
- Support Vector Machines / Kernel Machines
  - SVM primal/dual formulation.
  - Kernels from both SVM and least-squares duality.
  - Positive definite kernels.
  - Random features.
- Empirical Risk Minimization (short introduction to empirical process theory)
  - Basic concentration inequalities.
  - Loss functions and margin theory.
  - Uniform convergence and Rademacher complexity.
  - VC-dimension and the no free lunch theorem.
- First order / stochastic gradient methods
  - Basic framework.
  - Adaptive methods.
- Nearest neighbors and deep learning
  - Near optimality and curse of dimensionality.
  - Universal approximation.
  - Overparameterization and optimization.

Unsupervised Learning and Probabilistic Modeling

- Dimensionality reduction
  - Random projections.
  - Principle components analysis.
- Maximum likelihood estimation
- Latent variable models
  - Expectation-maximization.
  - Evidence lower bound (ELBO) and variational inference.
- Energy-based models
  - Langevin dynamics.
- Score matching and denoising score matching.
- Variational autoencoders.

- Normalizing flows
    - Finite compositions.
    - The log probability ODE.

Schedule

Note: The syllabus will be updated every week with assigned readings corresponding to each lecture. The section(s) in the **Reading and HW** column refer to the lecture notes.

Date	Topics	Reading and HW
8/26	Intro, binary classification, Perceptron algorithm.	Sections 1.1.
8/28	Perceptron mistake bound.	Sections 1.2.1.
9/2	Perceptron risk bound, support vector machines (SVMs).	**HW1 assigned.** Sections 1.2.2, 1.3.1.
9/4	SVM duality.	Section 1.3.1.
9/9	Feature maps and kernels.	Section 1.3.2.
9/11	Random features.	Section 1.3.3.
9/16	Empirical risk minimization (ERM) framework, hinge and logistic losses.	**HW1 due.** Sections 1.4.1 to 1.4.3.
9/18	Generalization for finite classes, Rademacher complexity and symmetrization.	**HW2 assigned.** Section 1.4.4, 1.4.7.
9/23	Explicit computations of Rademacher complexity.	
9/25	Sub-Gaussian random variables, basic concentration inequalities.	
9/30	VC dimension and examples.	
10/2	PAC learnability and the no free lunch theorem.	**HW2 due.**
10/7	**Midterm**	

10/9	**Holiday: Fall Recess**	
10/14	Stochastic optimization, first order methods.	**HW3 assigned.**
10/16	Nearest neighbor classification, near Bayes optimality, curse of dimensionality.	
10/21	Benign over-parameterization.	
10/23	Random projections and Johnson-Lindenstrauss.	
10/28	Principle components analysis.	**HW3 due and HW4 assigned.**
10/30	Maximum likelihood estimation.	
11/4	Latent variable models, expectation maximization (EM).	
11/6	Examples of EM.	
11/11	**Holiday: Veterans Day**	
11/13	Variational inference and the evidence lower bound (ELBO).	**HW4 due and mini-project assigned.**
11/18	Energy-based models, Langevin sampling.	
11/20	Score matching and denoising score matching.	
11/25	Variational autoencoders.	
11/27	**Holiday: Thanksgiving Break**	
12/2	Normalizing flows.	
12/4		

Note: the mini-project is due on 12/5.