Foundations of AI: Deep Learning with Applications, Spring 2026

Professor: Enric Boix-Adsera
Teaching Assistants: Sivangi Chatterjee, Zhehang Du, Sovesh Mohapatra, Sanskriti Sarkar

1. Course Logistics

Lecture times:

Section 1: Tuesday/Thursday, 1:45-3:15pm
Section 2: Tuesday/Thursday, 3:30pm-5pm

Course Q&A: The best place to ask questions about the course (both its content and its logistics) is on EdDiscussion, which will be monitored by the course staff. EdDiscussion is accessible through the course's Canvas website.

Office hours: for up-to-date times/places see course calendar on Canvas

- Office hour with Prof. Boix typically 4pm-5pm on Wednesdays in ARB 433
- Weekly TA office hours (schedule TBD – see course calendar)

Laptop policy: Please do not use phones, laptops and other electronic devices in class apart from on the in-class lab days.

Wharton lunches: Undergraduate (UG) and master-level (MBA, MS) students enrolled in the course are encouraged to sign up for lunch with the course instructor (in groups of 3 to 7 students) as part of the Wharton School's Faculty-Student Meal Program. Students may only participate once per semester per course. Students from outside Wharton (e.g., SAS, SEAS) are allowed to participate. Links to sign up for slots will be provided on Canvas.

2. Course Description

This course serves as an introduction to Deep Learning, which is the technology at the heart of modern AI. Topics include: what is a neural network and how to train it, generative AI, failure modes and safety of deep learning, efficient deep learning.

Prerequisites. Calculus, linear algebra (familiarity with matrices and matrix multiplication), programming experience with Python.

Learning goals. By the end of the course, you should be able to reason intelligently about deep learning and AI and have a solid conceptual grasp on what goes on under the hood in state-of-the-art AI models. You should expect to learn:

- what a neural network is, and how neural networks are "trained"
- how generative AI is trained, and how it can be customized with fine-tuning

- several of the failure modes of AI, as well as safety & reliability mitigations
- some emerging topics, such as leading methods to make neural networks more efficient

Additionally, this course has a programming component, through which you will get experience implementing and experimenting with several of the methods discussed in the course. Since the emphasis of this course is on conceptual understanding rather than on hands-on implementation, the assignments will have a lot of templated code that just has to be filled out.

Course organization. The course will be split into 4 modules, outlined below.

1. Foundations. The first module lays the conceptual groundwork for the rest of the class. We will cover historical background on deep learning, and introduce the three fundamental concepts that we will need throughout this class: loss functions, optimizers, and neural networks.
    a. We will cover four of the most important neural network architectures: fully-connected nets, recurrent nets, convolutional nets, and transformers.
    b. We will understand how these neural networks can be trained to solve classification problems.
    c. We will cover transfer learning, whereby knowledge from a network can be applied to a domain on which it was not trained.

2. Generative AI. In this second module, we will move beyond the basic supervised-learning setting, and cover neural networks that generate data such as language and images. These networks are at the heart of generative AI, which is the kind of AI that most students may have been exposed to through AI chatbots and image-generation models.
    a. We will first provide a DIY guide to large language models (such as ChatGPT). We will cover how the model is (1) pretrained, and (2) instruction-tuned, and (3) fine-tuned.
    b. Next, we will cover multimodal text-to-image and image-to-text models.

3. Failure modes and safety. In the third module, we will consider deployed neural network systems, and examine several of their failure modes and possible solutions.
    a. Among the safety failures covered will be: the remarkable non-robustness of neural networks to adversarial neural networks, LLMs' surprising tendency to "hallucinate" incorrect responses, and alignment issues in deploying models.
    b. We will consider mitigations of these issues, including: chain-of-thought and reasoning models for improved performance on logical tasks, and mechanistic interpretability to peek into the mechanisms by which models make their decisions.

4. Emerging topics. In our fourth module, we will turn our attention to the exorbitant energy, hardware, and data requirements from training a model.
    a. We will cover distillation and mixtures of experts as methods to reduce computational costs.

b. Then we will discuss data-efficient emerging architectures, such as looped transformers.
c. I have reserved 3 buffer days so that we can cover any particularly interesting deep learning developments contemporary to the course or topics that students may be interested in (students will vote on these from a list of proposed topics). These buffer days can also be used in case some material takes longer to cover than expected.

3. Assignments, exams, and grading policy

Assignments. The first 3 modules will have an accompanying coding project so that you can get hands-on experience with deep learning systems. The contents of these assignments are subject to change at the discretion of the instructor.

- *Assignment #1* ***(FOUNDATIONS)***:
    - Students will train their own neural networks to classify images, and investigate intriguing properties of these networks.

- *Assignment #2* ***(GENERATIVE AI)***:
    - Students will fine-tune an LLM to customize it to a task.

- *Assignment #3* ***(FAILURE MODES & SAFETY)***:
    - Probably: Students will either program retrieval-augmented generation, or work on mechanistically interpreting a trained model.

In-class exams. Two midterms and one final. Please contact the instructor as soon as possible at the beginning of the semester if you will need special accommodations.

Grading policy. The breakdown for grades is as follows.

- 30% Homework assignments (10% each)
    - We discourage AI assistance, since the goal of the assignments is to help you build understanding of the course material. Any students who choose to use AI assistance should take care to understand the code that they are submitting. The final exam will include a section with questions that specifically test whether the student has completed the programming projects and has a conceptual understanding of the concepts covered in them.
- 20% Midterm 1
    - Covers Module 1
- 20% Midterm 2
    - Covers Module 2
- 30% Final
    - Covers the entire course, with emphasis on Modules 3 and 4

Late policy. You are granted a **7-day total budget** of late days to use across all three assignments.

- *No Penalty:* As long as your total late days across the semester are ≤ 7.
- *Linear Penalty:* Once the 7-day budget is exhausted, each additional day an assignment is late results in a **10% reduction** of that assignment's final score.
- *Grade Optimization:* At the end of the term, we will automatically allocate your 7 grace days so as to maximally benefit your grade.

  Example Calculation If you were to submit assignments **3, 6, and 4 days late**, we would apply your budget to your highest-scoring work first. If your raw scores were $S_1 < S_2 < S_3$, we would apply 4 days to S_3, and 3 to S_2. After this adjustment, **the assignments would count as being 3, 3, and 0 days late**. Your final assignment scores after the penalty would be $S_1 \times 0.7$, $S_2 \times 0.7$, and S_3.

4. Course Schedule [tentative]

NOTE: The course content is subject to change at the discretion of the instructor, depending on the pace and interests of the class.

Date		**Topic**
Module 1		**Foundations**
Thu	1/15	*Introduction* ● Course overview ● Historical background ● A first peek at the simplest neural network: the single neuron **Assignment #0 (not graded) released**
Mon	1/19	**Please submit Assignment #0 (not graded) so that I can get to know you and your background, and calibrate the course appropriately.**
Tue	1/20	*What is a neural network?* ● The single neuron (perceptron) ● Two-layer & deep fully-connected models ● Expressivity separation
Thu	1/22	*How to train a neural network? (Part I: optimizers)* ● Loss functions ● Optimizers: gradient descent, stochastic gradient descent, Adam
Tue	1/27	*How to train a neural network? (Part II: backpropagation)* ● Computational graphs ● Backpropagation ● The computational graph associated to evaluating a network
Thu	1/29	***In-class laboratory session [Part 1]***

Date		Topic
		● The goal of this lecture is to give you basic familiarity you with Pytorch and get you ready to work on Assignment #1, which I will describe at the end of the class. **Assignment #1 released (due 2/27)**
Tue	2/3	***In-class laboratory session [Part 2]*** ● The goal of this lecture is to give you basic familiarity you with Pytorch and get you ready to work on Assignment #1.
Thu	2/5	*What is the transformer architecture?* ● Attention module ● Transformers for text classification
Tue	2/10	**In-Class Midterm #1 [covers topics until, and including, 2/5 lecture]**
Module 2		**Generative AI**
Thu	2/12	*What is a language model?* ● Generating language with next-token-prediction classification ● Pretraining
Tue	2/17	*Why make language models large?* ● Scaling laws ● Compute-optimal models
Thu	2/19	*How to make your model follow instructions or customize it?* ● Supervised fine-tuning ● LoRA
Mon	2/23	**Drop Period Ends**
Tue	2/24	*How to make a model multi-modal?* ● Transformers for images ● Contrastive learning of representations (CLIP)
Thu	2/26	*How to make a model generate images?* ● Diffusion models
Fri	2/27	**Assignment #1 due**
Tue	3/3	***In-class laboratory session*** and intro to Assignment #2 ● The goal of this lecture is to get you ready to work on Project #2, which I will describe at the end of the class. **Assignment #2 released (due 3/27)**
Thu	3/5	**In-Class Midterm #2 [covers topics until, and including, 2/26 lecture]**
No Classes – Spring Break		

Date		**Topic**
Module 3		**Failure modes and safety**
Tue	3/17	*Are neural network models robust?* ● Adversarial examples in images, and robust image models ● Adversarial examples in text ● Jailbreaks
Thu	3/19	*What are hallucinations, and can we avoid them?* ● Hallucinations ● Retrieval-augmented generation and semantic search
Tue	3/24	*Mitigating hallucinations with reasoning* ● Chain-of-thought ● Reasoning models, and how to train them / new scaling laws
Thu	3/26	*Alignment* ● Alignment-faking ● Sandbagging
Fri	3/27	**Assignment #2 due**
Tue	3/31	*Mechanistic interpretability* ● Linear representation hypothesis and linear probes ● Steering (or sparse autoencoders)
Thu	4/2	***In-class laboratory session*** and intro to Assignment #3 ● We will collectively walk through and fill in the Pytorch code for a mechanistic interpretability project where you will be able to examine model internals and understand what is going on under the hood / or a project where you examine failure modes of these models with respect to jailbreaking or alignment. ● The goal of this lecture is to get you ready to work on Project #3, which I will describe at the end of the class. **Assignment #3 released (due 4/24)**
Tue	4/7	*What part of training data is most responsible for my model's behavior?* ● Data attribution
Module 4		**Emerging topics (e.g. efficiency, …)**
Thu	4/9	*Can we make models smaller?* ● Distillation
Tue	4/14	*Can we make models more efficient?* ● Mixture of experts models

Date		Topic
Thu	4/16	*Can we make models more data-efficient at reasoning?* ● Looped transformers / tiny recursive machines
Tue	4/21	Buffer day / reserved for emerging developments in deep learning
Thu	4/23	Buffer day / reserved for emerging developments in deep learning
Fri	4/24	**Assignment #3 due**
Tue	4/28	Buffer day / reserved for emerging developments in deep learning
Final Exam [date TBA]		

5. Reading list [optional]

Since we will cover state-of-the-art topics in this course, there is no one textbook that contains all of the material. Therefore, the following readings are optional, but may be useful for a curious student.

Module 1 (FOUNDATIONS):

- *Deep Learning* by John D. Kelleher, The MIT Press Essential Knowledge series.
- *Deep Learning* by Ian Goodfellow and Yoshua Bengio and Aaron Courville, MIT Press.
- First chapter of *Deep Learning: Foundations and Concepts* by Christopher Bishop and Hugh Bishop
- Deep Learning: A Practitioner's Approach by Gibson and Patterson
- *The Deep Learning Revolution* by Sejnowski.

Module 2 (GENERATIVE AI):

- On the opportunities and risks of foundation models: https://arxiv.org/pdf/2108.07258
- Scaling laws: https://arxiv.org/pdf/2001.08361

Module 3 (FAILURE MODES AND SAFETY):

- Adversarial robustness: https://arxiv.org/pdf/1312.6199
- Retrieval-augmented generation: https://arxiv.org/abs/2005.11401
- Reasoning models: https://arxiv.org/abs/2501.19393

Module 4 (EMERGING TOPICS):

- Distillation: https://www.ttic.edu/dl/dark14.pdf
- Tiny Recursive Machines: https://arxiv.org/abs/2510.04871
- Looped Transformers: https://arxiv.org/abs/2409.15647

Last updated: 12/21/25