ECE472, Deep Learning – Syllabus, Fall 2025, Room 505, Th 6-9

tldr: We will introduce the concepts relevant to so-called "deep learning" — our fundamental processes are based on computations performed over differentiable graphs, where nodes correspond to operations and edges correspond to operands. We will use the Microsoft Teams site: "ECE-472-1-Deep Learning-2025FA"

Instructor Chris Curro, EE '15, MEE '16; `christopher.curro@cooper.edu` (primary); `professor@curro.cc` (backup to bump);

Reference Textbook Christopher Bishop and Hugh Bishop. 2024. *Deep Learning: Foundations and Concepts.* Springer. `https://www.bishopbook.com/`

Assignments There will be a handful of programming assignments. You will print out the source code, any plots or reports, and include a cover sheet detailing what you completed or did not complete, and if you received an approved extension (and until when). Please staple them well.

Citations Plagiarism will not be tolerated. All cases of suspected plagiarism will be submitted to the Dean's office for investigation. Feel free to ask questions of your peers, but please cite them for any help you receive. Cite resources you may utilize from the web and elsewhere.

Quizzes There will be quizzes most weeks. These quizzes will test understanding of assigned research papers. If you must miss a quiz, please let me know beforehand and we will arrange appropriate make-up accommodations, otherwise you will receive a zero for that quiz. Except for extreme cases, you will be limited to 4 make-up quizzes.

Grading Grading breakdown in table at the bottom of the page. If you fail to submit an assignment you will fail the course. All assignments will be graded out of 5 points. Unexcused late assignments will have a single full point deducted per 2 days late. The maximum grade for any tardy assignment is a 4. No work can be submitted after the last day of the semester; it will not be considered.

Attendance We will not take attendance, but it may factor into your participation score. The participation score is multifaceted. We will discuss this during the first class. If you are sick, do not come to class; send me an email.

Office hours We will arrive at an appropriate schedule during the first class. Expect 1 or 2 hours per week. Additional hours by appointment. Office hours will be conducted remotely on Microsoft Teams.

Grading	
Assignments	45%
Quizzes	45%
Participation	10%

Boilerplate

Required links

- `https://cooper.edu/sites/default/files/uploads/assets/site/files/2020/Cooper-Union-Policy-Upholding-Human-Rights-Title-IX-Protections.pdf`
- `https://cooper.edu/students/student-affairs/disability`
- `https://cooper.edu/students/student-affairs/health/counseling`

Student Outcomes

- Ability to
  - discuss contemporary research in an intelligent way
  - recognize failings in a given experiment and synthesize follow-up experimentation
  - synthesize hypotheses on ablative and compositional experiments
  - argue in an evidence-based way and make conclusions
  - communicate mathematical concepts in a narrative
  - identify situations in which deep learning may or may not be appropriate over other machine learning techniques

We will assess the aforementioned abilities through class discussions, quizzes, and assignment submissions.

Prerequisite Skills

- Knowledge of a programming language (Python preferred)
- Knowledge of differentiation in multivariate calculus
- Knowledge of basic linear algebra and probability (e.g., matrix multiplication, distributions)

Approximate list of topics

Fundamentals Linear regression with basis functions. Gradient descent and optimizers (e.g., Adam). Automatic differentiation. Multi-layer perceptrons. Activation functions. Loss functions. Regularization (L1/L2, Dropout). Normalization layers (Batch, Group). Weight initialization.

Convolutional Architectures Convolutional layers, pooling, strided convolutions. Receptive fields. Residual connections.

Sequence Models and Transformers Attention and multi-head attention. Tokenization. The Transformer architecture. Vision Transformers (ViT). Parameter-Efficient Fine-Tuning (PEFT) and LoRA.

Generative Modeling Autoencoders (sparse, variational). Generative Adversarial Networks (GANs). Diffusion Models. Applications in text-to-image synthesis and style transfer.

Reinforcement Learning RL basics. World models. Applications in games and scientific discovery.

Large Language Models Generative pre-training. Chain-of-Thought (CoT) prompting. Retrieval-Augmented Generation (RAG). Long-context models. Reinforcement Learning from Human Feedback (RLHF).

Interpretability and Analysis Probing and feature visualization. Sparsity and monosemanticity. Understanding model behavior and emergent abilities.

AI Safety, Alignment, and Ethics Model control and persona shaping. Prompt security. Monitoring and agentic risks. Societal impact, bias, and persuasion.

General Homework Requirements

1. Write tightly scoped classes/functions. When working in Flax NNX, inherit from `nnx.Module`.
2. Homework assignments will be due digitally at 10 PM the evening before class. You will submit a single PDF containing your entire assignment. Use `a2ps` and `ps2pdf` or similar to generate. *However you must bring a hard copy to submit at class time.* I will be delivering feedback on the hard copy.
3. I will grade the assignments in a comprehensive and holistic manner.
4. I may return general class-wide feedback on each assignment.
5. Each assignment should be reproducible. (i.e., running the code twice should return the exact same result)
6. Submission of "notebooks" is forbidden.
7. The *only* framework references you will need for completing the core assignments are the official Flax, JAX, and Optax documentation. Do not go searching for guides on YouTube, Medium, etc. Be sure to use the modern `nnx` API, not the legacy `flax.linen` API. Here are some good places to start:
   - `https://flax.readthedocs.io/en/latest/index.html`
   - `https://flax.readthedocs.io/en/latest/nnx_basics.html`
   - `https://docs.jax.dev/en/latest/index.html`
   - `https://optax.readthedocs.io/en/latest/`
   - `https://flax.readthedocs.io/en/latest/api_reference/flax.nnx/training/optimizer.html`
8. Use the Python docs liberally as well: `https://docs.python.org/3/`
9. Do not use AI products to write your homework assignments. We are studying how to *make* these products.

Assignment 1 — Due: Sept. 10 at 10 PM

tldr: Perform linear regression of a noisy sine wave using a set of Gaussian basis functions with learned location and scale parameters. Model parameters are learned with stochastic gradient descent. Use of automatic differentiation is required. Hint: note your limits!

Problem Statement Consider a set of scalars $\{x_1, x_2, \ldots, x_N\}$ drawn from $\mathcal{U}(0, 1)$ and a corresponding set $\{y_1, y_2, \ldots, y_N\}$ where:

$$y_i = \sin(2\pi x_i) + \epsilon_i \tag{1}$$

and ϵ_i is drawn from $\mathcal{N}(0, \sigma_{\text{noise}})$. Given the following functional form:

$$\hat{y}_i = \sum_{j=1}^{M} w_j \phi_j \left(x_i \mid \mu_j, \sigma_j\right) + b \tag{2}$$

with:

$$\phi(x \mid \mu, \sigma) = \exp \frac{-(x-\mu)^2}{\sigma^2} \tag{3}$$

find estimates $\hat{b}$, $\{\hat{\mu}_j\}$, $\{\hat{\sigma}_j\}$, and $\{\hat{w}_j\}$ that minimize the loss function:

$$J(y, \hat{y}) = \frac{1}{2}(y - \hat{y})^2 \tag{4}$$

for all (x_i, y_i) pairs. Estimates for the parameters must be found using stochastic gradient descent. A framework that supports automatic differentiation must be used. Set $N = 50, \sigma_{\text{noise}} = 0.1$. Select M as appropriate. Produce two plots. First, show the data points, a noiseless sine wave, and the manifold produced by the regression model. Second, show each of the M basis functions.

Requirements Create a `Linear` module. Create a `BasisExpansion` module. Plots must be of suitable visual quality.

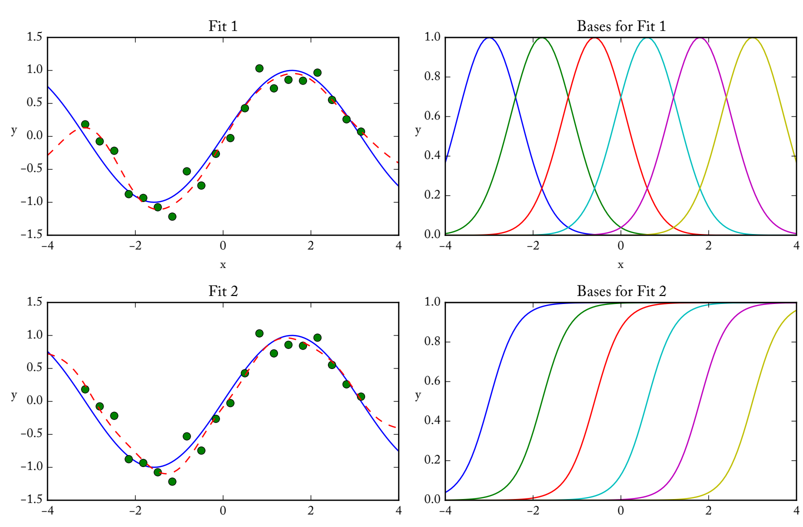

Figure 1: Example plots for models with equally spaced sigmoid and Gaussian basis functions.

Assignment 2 — Due: Sept. 17 at 10 PM

tldr: Perform binary classification on the spirals dataset using a multi-layer perceptron. You must generate the data yourself.

Problem Statement Consider a set of examples with two classes and distributions as in Figure 2. Given the vector $x \in \mathbb{R}^2$ infer its target class $t \in \{0, 1\}$. As a model use a multi-layer perceptron f which returns an estimate for the conditional density $p(t = 1 \mid x)$:

$$f : \mathbb{R}^2 \to [0, 1] \tag{5}$$

parameterized by some set of values θ. All of the examples in the training set should be classified correctly (i.e. $p(t = 1 \mid x) > 0.5$ if and only if $t = 1$). Produce one plot. Show the examples and the boundary corresponding to $p(t = 1 \mid x) = 0.5$. The plot must be of suitable visual quality. It may be difficult to find an appropriate functional form for f, write a few sentences discussing your various attempts.

Requirements

1. Generate data using an instance of `numpy.random.Generator`. Note how many times my spirals lap the origin.
2. Create an `MLP` class. The `MLP` class should inherit from `nnx.Module`. You may find `nnx.scan` useful for building repetitive network structures. To use `nnx.scan` effectively, you will likely want to use `nnx.vmap` to instantiate your layers and `nnx.split_rngs` to manage your PRNG keys. It should have the following interface:

```
MLP(
  num_inputs,
  num_outputs,
  num_hidden_layers,
  hidden_layer_width,
  hidden_activation=nnx.identity,
  output_activation=nnx.identity,
)
```

3. Learn how to use `sklearn.inspection.DecisionBoundaryDisplay`
4. Your network must operate on Cartesian coordinates. Do not transform the coordinates to be polar.

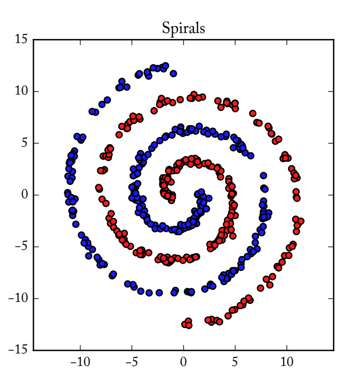

Figure 2: Sample spiral data.

Assignment 3 — Due: Sept. 24 at 10 PM

tldr: Classify MNIST digits with a convolutional neural network. Get at least 95.5% accuracy on the test set.

Problem Statement Consider the MNIST dataset consisting of 50,000 training images, and 10,000 test images. Each instance is a 28×28 pixel handwritten digit zero through nine. Train a (optionally convolutional) neural network for classification using the training set that achieves at least 95.5% accuracy on the test set. Do not explicitly tune hyperparameters based on the test set performance, use a validation set taken from the training set as discussed in class. Use dropout and an L^2 penalty for regularization. Note: if you write a sufficiently general program the next assignment may be very easy.

Use the `tensorflow_datasets` package to load the MNIST dataset.

Requirements

1. Create a `Conv2d` class that inherits from `nnx.Module` and uses `nnx.Conv`. *Do not try to write your own convolution implementation.*
2. Create a `Classifier` class that inherits from `nnx.Module`. You may find `nnx.scan` useful for building repetitive network structures. To use `nnx.scan` effectively, you will likely want to use `nnx.vmap` to instantiate your layers and `nnx.split_rngs` to manage your PRNG keys. The interface for `Classifier` should at a minimum be:

```
Classifier(
  input_depth: int,
  layer_depths: list[int],
  layer_kernel_sizes: list[tuple[int, int]],
  num_classes: int,
)
```

Extra challenge (optional) In addition to the above, the student with the fewest number of parameters for a network that gets at least 80% accuracy on the test set will receive a prize. There will be an extra prize if anyone can achieve 80% on the test set with a single-digit number of parameters. For this extra challenge you can make your network have any crazy kind of topology you'd like, it just needs to be optimized by a gradient-based algorithm.

Assignment 4 — Due: Oct. 8 at 10 PM

tldr: Classify CIFAR10. Achieve performance similar to the state of the art. Classify CIFAR100. Achieve a top-5 accuracy of 90%.

Problem Statement Consider the CIFAR10 and CIFAR100 datasets which contain 32×32 pixel color images. Train a classifier for each of these with performance similar to the state of the art (for CIFAR10). It is your task to figure out what is state of the art. Feel free to adapt any techniques from papers you read. Write a paragraph or two summarizing your experiments. Hopefully you'll be able to reuse your MNIST program.

Requirements

1. Experiment with data augmentation.
2. Use your `Conv2d` class from the previous assignment.
3. Create a `GroupNorm` class that inherits from `nnx.Module`.
4. Create a `ResidualBlock` class that inherits from `nnx.Module` around your `Conv2d` and `GroupNorm` classes.
5. Modify your `Classifier` class to use the new `ResidualBlock` class.

Assignment 5 — Due: Oct. 15 at 10 PM

tldr: Classify the AG News dataset.

Problem Statement Consider the AG News dataset at `https://huggingface.co/datasets/ag_news` which contains headlines and descriptions for a large set of news articles. Create a model to categorize the articles. Perform proper cross-validation. You may start from pre-trained models.

Assignment 6 — Due: Nov. 5 at 10 PM

Problem Statement Implement a `MultiHeadAttention` class and a `TransformerBlock` class. Assume 1-D case only. Provide a sufficient set of tests to prove that they work correctly.

Assignment 7 — Due: Nov. 19 at 10 PM

Problem Statement Review the following paper:

Adly Templeton et al. "Scaling Monosemanticity: Extracting Interpretable Features from Claude 3 Sonnet". In: *Transformer Circuits Thread* (2024). URL: `https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html`

Consider the sparse autoencoder structure and how you might apply the techniques to induce sparsity in an MLP in a text classifier (see HW 5). Attempt to implement this sparse structure. Determine what must be true about the dimensionality of the sparse layer in relation to the size of the dataset for useful monosemanticity to hold. Identify interpretable features, and discuss.

Final Paper — Due: Dec 18 at 6 PM

tldr: Write a formal position paper arguing for an assigned stance (affirmative or negative) on a debatable topic concerning the ethics and application of modern artificial intelligence.

Problem Statement The goal of this paper is for you to develop a rigorous, evidence-based argument from an assigned perspective. Rather than choosing your own topic, I will assign you a specific, debatable statement (a "position") and a stance (either affirmative or negative). You must construct the most compelling and well-supported argument for your given side, regardless of your personal beliefs.

Positions will cover contemporary issues in our field, such as: "It is ethical to use AI to write programming assignments for school," or "It is ethical to use AI coding assistants in a professional software engineering environment." I will divide the class evenly on these topics to ensure students argue both the affirmative and negative cases for each position.

Your success in this assignment depends on the strength and clarity of your argument, your use of credible evidence, and your ability to anticipate and rebut counterarguments. This assignment will assess how you engage critically with the non-technical implications of the technologies we study in this course and communicate complex ideas in a formal written style.

Requirements

1. Evidence: You must conduct bibliographic research to support your arguments, and appropriately cite these works. You may augment scholarly sources with anecdotal evidence or other non-scholarly works (e.g., blogs from reputable developers or writers), however you must be appropriately critical of these resources.
2. Structure: Your paper must include the following logical components:
   - An introduction that clearly states the position and your thesis.
   - A body of several paragraphs, each presenting a distinct point that you support with evidence from your research.
   - A section where you acknowledge and refute at least one major counterargument to your position.
   - A conclusion that summarizes your argument and offers a final persuasive thought.
3. Submission: You must submit the final paper as a hard copy at the final class. I cannot accept late submissions, due to college rules.

Papers

This paper list, for Fall 2025, is up-to-date as of September 4, 2025. Papers marked with a † are **optional** pre-reading, but will be discussed during class time.

"Official" BibTeX citations used where provided by the original publisher.

Week 1

Read by Sept. 11

1. Atilim Gunes Baydin, Barak A. Pearlmutter, and Alexey Andreyevich Radul. "Automatic differentiation in machine learning: a survey". In: *CoRR* abs/1502.05767 (2015). arXiv: `1502.05767`. URL: `http://arxiv.org/abs/1502.05767`
   - Read up to, but not including, section 4.2.
2. Leon Bottou. "Stochastic Gradient Descent Tricks". In: *Neural Networks, Tricks of the Trade, Reloaded*. Neural Networks, Tricks of the Trade, Reloaded. Vol. 7700. Lecture Notes in Computer Science (LNCS). Springer, Jan. 2012, pp. 430–445. URL: `https://www.microsoft.com/en-us/research/publication/stochastic-gradient-tricks/`
3. Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: `1412.6980 [cs.LG]`

† Ilya Loshchilov and Frank Hutter. *Decoupled Weight Decay Regularization*. 2019. arXiv: `1711.05101 [cs.LG]`

Week 2

Read by Sept. 18

4. Kaiming He et al. *Identity Mappings in Deep Residual Networks*. 2016. arXiv: `1603.05027 [cs.CV]`
5. Kaiming He et al. *Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification*. 2015. arXiv: `1502.01852 [cs.CV]`
6. Andre F. de Araújo, Wade Norris, and Jack Sim. "Computing Receptive Fields of Convolutional Neural Networks". In: *Distill* (2019). URL: `https://distill.pub/2019/computing-receptive-fields`

Week 3

Read by Sept. 25

7. Tomas Mikolov et al. "Distributed Representations of Words and Phrases and their Compositionality". In: *Advances in Neural Information Processing Systems*. Ed. by C.J. Burges et al. Vol. 26. Curran Associates, Inc., 2013. URL: `https://proceedings.neurips.cc/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf`
8. Ashish Vaswani et al. "Attention Is All You Need". In: *CoRR* abs/1706.03762 (2017). arXiv: `1706.03762`. URL: `http://arxiv.org/abs/1706.03762`
9. Alec Radford et al. "Language Models are Unsupervised Multitask Learners". In: (2019)
10. Taku Kudo and John Richardson. "SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing". In: *CoRR* abs/1808.06226 (2018). arXiv: `1808.06226`. URL: `http://arxiv.org/abs/1808.06226`

† Jordan Hoffmann et al. *Training Compute-Optimal Large Language Models*. 2022. DOI: `10.48550/ARXIV.2203.15556`. URL: `https://arxiv.org/abs/2203.15556`

† Gheorghe Comanici et al. *Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multimodality, Long Context, and Next Generation Agentic Capabilities*. 2025. arXiv: `2507.06261 [cs.CL]`. URL: `https://arxiv.org/abs/2507.06261`

† *System Card: Claude Opus 4 & Claude Sonnet 4*. 2025. URL: `https://www-cdn.anthropic.com/07b2a3f9902ee19fe39a36ca638e5ae987bc64dd.pdf`

† OpenAI. *ChatGPT agent System Card*. July 2025. URL: `https://openai.com/index/chatgpt-agent-system-card/`

Week 4

Read by Oct. 2

11. Alexey Dosovitskiy et al. "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale". In: *CoRR* abs/2010.11929 (2020). arXiv: `2010.11929`. URL: `https://arxiv.org/abs/2010.11929`
12. Alec Radford et al. *Learning Transferable Visual Models From Natural Language Supervision*. 2021. DOI: `10.48550/ARXIV.2103.00020`. URL: `https://arxiv.org/abs/2103.00020`
13. Andrew Jaegle et al. *Perceiver: General Perception with Iterative Attention*. 2021. arXiv: `2103.03206 [cs.CV]`

† Andrew Jaegle et al. *Perceiver IO: A General Architecture for Structured Inputs and Outputs*. 2021. arXiv: `2107.14795 [cs.LG]`

Week 5

Read by Oct. 9

14. Nathan Lambert et al. *Tulu 3: Pushing Frontiers in Open Language Model Post-Training*. 2025. arXiv: 2411.15124 [cs.CL]. URL: https://arxiv.org/abs/2411.15124
15. Nikhil Kandpal et al. *The Common Pile v0.1: An 8TB Dataset of Public Domain and Openly Licensed Text*. 2025. arXiv: 2506.05209 [cs.CL]. URL: https://arxiv.org/abs/2506.05209
16. Matteo Cargnelutti et al. *Institutional Books 1.0: A 242B token dataset from Harvard Library's collections, refined for accuracy and usability*. 2025. arXiv: 2506.08300 [cs.CL]. URL: https://arxiv.org/abs/2506.08300
17. David Silver and Richard S Sutton. "Welcome to the Era of Experience". In: (2025). URL: https://storage.googleapis.com/deepmind-media/Era-of-Experience%20/The%20Era%20of%20Experience%20Paper.pdf

† Sanjay Surendranath Girija et al. *Optimizing LLMs for Resource-Constrained Environments: A Survey of Model Compression Techniques*. 2025. arXiv: 2505.02309 [cs.LG]. URL: https://arxiv.org/abs/2505.02309

Week 6

Read by Oct. 16

18. Edward J. Hu et al. *LoRA: Low-Rank Adaptation of Large Language Models*. 2021. arXiv: 2106.09685 [cs.CL]
19. Hao Liu, Matei Zaharia, and Pieter Abbeel. *Ring Attention with Blockwise Transformers for Near-Infinite Context*. 2023. arXiv: 2310.01889 [cs.CL]. URL: https://arxiv.org/abs/2310.01889
20. Sachin Goyal et al. *Think before you speak: Training Language Models With Pause Tokens*. 2024. arXiv: 2310.02226 [cs.CL]. URL: https://arxiv.org/abs/2310.02226
21. Shibo Hao et al. *Training Large Language Models to Reason in a Continuous Latent Space*. 2024. arXiv: 2412.06769 [cs.CL]. URL: https://arxiv.org/abs/2412.06769
22. Biao Zhang et al. *Encoder-Decoder Gemma: Improving the Quality-Efficiency Trade-Off via Adaptation*. 2025. arXiv: 2504.06225 [cs.CL]. URL: https://arxiv.org/abs/2504.06225

† Sabri Eyuboglu et al. *Cartridges: Lightweight and general-purpose long context representations via self-study*. 2025. arXiv: 2506.06266 [cs.CL]. URL: https://arxiv.org/abs/2506.06266

† Terry Koo, Frederick Liu, and Luheng He. *Automata-based constraints for language model decoding*. 2024. arXiv: 2407.08103 [cs.CL]. URL: https://arxiv.org/abs/2407.08103

† Michael Poli et al. *Hyena Hierarchy: Towards Larger Convolutional Language Models*. 2023. arXiv: 2302.10866 [cs.LG]

Week 7

Read by Oct. 23

23. Charlie Snell et al. *Scaling LLM Test-Time Compute Optimally can be More Effective than Scaling Model Parameters*. 2024. arXiv: 2408.03314 [cs.LG]. URL: https://arxiv.org/abs/2408.03314
24. Kaiming He et al. *Masked Autoencoders Are Scalable Vision Learners*. 2021. arXiv: 2111.06377 [cs.CV]
25. Oriane Siméoni et al. "DINOv3". In: (Aug. 2025). URL: https://ai.meta.com/research/publications/dinov3

† Niklas Muennighoff et al. *s1: Simple test-time scaling*. 2025. arXiv: 2501.19393 [cs.CL]. URL: https://arxiv.org/abs/2501.19393

† Adam Pearce, Asma Ghandeharioun, and Nada Hussein. *Do machine learning models memorize or generalize?* URL: https://pair.withgoogle.com/explorables/grokking/

† Preetum Nakkiran et al. "Deep Double Descent: Where Bigger Models and More Data Hurt". In: *CoRR* abs/1912.02292 (2019). arXiv: 1912.02292. URL: http://arxiv.org/abs/1912.02292

Week 8

Read by Oct. 30

26. Rawal Khirodkar et al. *Sapiens: Foundation for Human Vision Models*. 2024. arXiv: 2408.12569 [cs.CV]. URL: https://arxiv.org/abs/2408.12569
27. Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. *A Neural Algorithm of Artistic Style*. 2015. arXiv: 1508.06576 [cs.CV]
28. Robin Rombach et al. *High-Resolution Image Synthesis with Latent Diffusion Models*. 2021. arXiv: 2112.10752 [cs.CV]. URL: https://arxiv.org/abs/2112.10752
29. Aditya Ramesh et al. *Hierarchical Text-Conditional Image Generation with CLIP Latents*. 2022. DOI: 10.48550/ARXIV.2204.06125. URL: https://arxiv.org/abs/2204.06125

- † Xun Huang and Serge Belongie. *Arbitrary Style Transfer in Real-time with Adaptive Instance Normalization*. 2017. arXiv: 1703.06868 [cs.CV]
- † Tero Karras et al. *Analyzing and Improving the Image Quality of StyleGAN*. 2020. arXiv: 1912.04958 [cs.CV]
- † Dani Valevski et al. *Diffusion Models Are Real-Time Game Engines*. 2024. arXiv: 2408.14837 [cs.LG]. URL: https://arxiv.org/abs/2408.14837
- † Aäron van den Oord et al. "WaveNet: A Generative Model for Raw Audio". In: *CoRR* abs/1609.03499 (2016). arXiv: 1609.03499. URL: http://arxiv.org/abs/1609.03499

Week 9

Read by Nov. 6

30. Jinhyuk Lee et al. *Gemini Embedding: Generalizable Embeddings from Gemini*. 2025. arXiv: 2503.07891 [cs.CL]. URL: https://arxiv.org/abs/2503.07891
31. Darren Edge et al. *From Local to Global: A Graph RAG Approach to Query-Focused Summarization*. 2025. arXiv: 2404.16130 [cs.CL]. URL: https://arxiv.org/abs/2404.16130
32. Orion Weller et al. *On the Theoretical Limitations of Embedding-Based Retrieval*. 2025. arXiv: 2508.21038 [cs.IR]. URL: https://arxiv.org/abs/2508.21038

- † Kai Arulkumaran et al. "A Brief Survey of Deep Reinforcement Learning". In: *CoRR* abs/1708.05866 (2017). arXiv: 1708.05866. URL: http://arxiv.org/abs/1708.05866
- † Julian Schrittwieser et al. "Mastering Atari, Go, Chess and Shogi by Planning with a Learned Model". In: *CoRR* abs/1911.08265 (2019). arXiv: 1911.08265. URL: http://arxiv.org/abs/1911.08265
- † Lili Chen et al. "Decision Transformer: Reinforcement Learning via Sequence Modeling". In: *CoRR* abs/2106.01345 (2021). arXiv: 2106.01345. URL: https://arxiv.org/abs/2106.01345
- † Danijar Hafner et al. "Mastering Atari with Discrete World Models". In: *CoRR* abs/2010.02193 (2020). arXiv: 2010.02193. URL: https://arxiv.org/abs/2010.02193

Week 10

Read by Nov. 13

33. Alexander Novikov et al. *AlphaEvolve: A coding agent for scientific and algorithmic discovery*. 2025. arXiv: 2506.13131 [cs.AI]. URL: https://arxiv.org/abs/2506.13131

- † Jonas Degrave et al. "Magnetic control of tokamak plasmas through deep reinforcement learning". In: *Nature* 602.7897 (Feb. 2022), pp. 414–419. DOI: 10.1038/s41586-021-04301-9. URL: https://doi.org/10.1038%2Fs41586-021-04301-9
- † Julien Perolat et al. "Mastering the game of Stratego with model-free multiagent reinforcement learning". In: *Science* 378.6623 (Dec. 2022), pp. 990–996. DOI: 10.1126/science.add4679. URL: https://arxiv.org/abs/2206.15378
- † Gemini Robotics Team et al. *Gemini Robotics: Bringing AI into the Physical World*. 2025. arXiv: 2503.20020 [cs.RO]. URL: https://arxiv.org/abs/2503.20020
- † Anthropic. *How Anthropic teams use Claude Code*. 2025. URL: https://www-cdn.anthropic.com/58284b19e702b49db9302d5b6f135ad8871e7658.pdf
- † Josh Abramson et al. "Accurate structure prediction of biomolecular interactions with AlphaFold 3". In: *Nature* 630.8016 (May 2024), pp. 493–500. ISSN: 1476-4687. DOI: 10.1038/s41586-024-07487-w. URL: http://dx.doi.org/10.1038/s41586-024-07487-w

Week 11

Read by Nov. 20

34. Adly Templeton et al. "Scaling Monosemanticity: Extracting Interpretable Features from Claude 3 Sonnet". In: *Transformer Circuits Thread* (2024). URL: https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html
35. Jack Lindsey et al. "On the Biology of a Large Language Model". In: *Transformer Circuits Thread* (2025). URL: https://transformer-circuits.pub/2025/attribution-graphs/biology.html
36. Ujjwal Upadhyay et al. *Time Blindness: Why Video-Language Models Can't See What Humans Can?* 2025. arXiv: 2505.24867 [cs.CV]. URL: https://arxiv.org/abs/2505.24867

- † Muzammal Naseer et al. "Intriguing Properties of Vision Transformers". In: *CoRR* abs/2105.10497 (2021). arXiv: 2105.10497. URL: https://arxiv.org/abs/2105.10497
- † Rynaa Grover et al. *HueManity: Probing Fine-Grained Visual Perception in MLLMs*. 2025. arXiv: 2506.03194 [cs.CV]. URL: https://arxiv.org/abs/2506.03194
- † Joshua Vendrow et al. *Do Large Language Model Benchmarks Test Reliability?* 2025. arXiv: 2502.03461 [cs.LG]. URL: https://arxiv.org/abs/2502.03461
- † Shivalika Singh et al. *The Leaderboard Illusion*. 2025. arXiv: 2504.20879 [cs.AI]. URL: https://arxiv.org/abs/2504.20879

Week 12

Read by Nov. 25–Note, modified schedule.

37. Usman Anwar et al. *Foundational Challenges in Assuring Alignment and Safety of Large Language Models*. 2024. arXiv: 2404.09932 [cs.LG]. URL: https://arxiv.org/abs/2404.09932
38. Edoardo Debenedetti et al. *Defeating Prompt Injections by Design*. 2025. arXiv: 2503.18813 [cs.CR]. URL: https://arxiv.org/abs/2503.18813
39. Tomek Korbak et al. *Chain of Thought Monitorability: A New and Fragile Opportunity for AI Safety*. 2025. arXiv: 2507.11473 [cs.AI]. URL: https://arxiv.org/abs/2507.11473
40. Peter Barnett, Aaron Scher, and David Abecassis. *Technical Requirements for Halting Dangerous AI Activities*. 2025. arXiv: 2507.09801 [cs.AI]. URL: https://arxiv.org/abs/2507.09801

- † Santiago (Sal) Díaz, Christoph Kern, and Kara Olive. *Google's Approach for Secure AI Agents*. Tech. rep. 2025. URL: https://storage.googleapis.com/gweb-research2023-media/pubtools/1018686.pdf
- † Rohin Shah et al. *An Approach to Technical AGI Safety and Security*. 2025. arXiv: 2504.01849 [cs.AI]. URL: https://arxiv.org/abs/2504.01849
- † Jan Kulveit et al. *Gradual Disempowerment: Systemic Existential Risks from Incremental AI Development*. 2025. arXiv: 2501.16946 [cs.CY]. URL: https://arxiv.org/abs/2501.16946
- † Aengus Lynch et al. "Agentic Misalignment: How LLMs Could be an Insider Threat". In: *Anthropic Research* (2025). https://www.anthropic.com/research/agentic-misalignment

Week 13

Read by Dec. 4

41. Miles Wang et al. *Persona Features Control Emergent Misalignment*. 2025. arXiv: 2506.19823 [cs.LG]. URL: https://arxiv.org/abs/2506.19823
42. Runjin Chen et al. *Persona Vectors: Monitoring and Controlling Character Traits in Language Models*. 2025. arXiv: 2507.21509 [cs.CL]. URL: https://arxiv.org/abs/2507.21509
43. Jon Saad-Falcon et al. *Shrinking the Generation-Verification Gap with Weak Verifiers*. 2025. arXiv: 2506.18203 [cs.CL]. URL: https://arxiv.org/abs/2506.18203

Week 14

Read by Dec. 18

44. Kiran Tomlinson et al. *Working with AI: Measuring the Occupational Implications of Generative AI*. 2025. arXiv: 2507.07935 [cs.AI]. URL: https://arxiv.org/abs/2507.07935
45. Hannah Rose Kirk et al. *The PRISM Alignment Dataset: What Participatory, Representative and Individualised Human Feedback Reveals About the Subjective and Multicultural Alignment of Large Language Models*. 2024. arXiv: 2404.16019 [cs.CL]. URL: https://arxiv.org/abs/2404.16019
46. Kobi Hackenburg et al. *The Levers of Political Persuasion with Conversational AI*. 2025. arXiv: 2507.13919 [cs.CL]. URL: https://arxiv.org/abs/2507.13919
47. Lily Hong Zhang et al. *Cultivating Pluralism In Algorithmic Monoculture: The Community Alignment Dataset*. 2025. arXiv: 2507.09650 [cs.LG]. URL: https://arxiv.org/abs/2507.09650
48. Peter Salib and Simon Goldstein. "AI Rights for Human Flourishing". In: (2025). DOI: 10.2139/ssrn.5353214. URL: http://dx.doi.org/10.2139/ssrn.5353214

- † Kenneth Payne and Baptiste Alloui-Cros. *Strategic Intelligence in Large Language Models: Evidence from evolutionary Game Theory*. 2025. arXiv: 2507.02618 [cs.AI]. URL: https://arxiv.org/abs/2507.02618
- † Joel Z. Leibo et al. *Societal and technological progress as sewing an ever-growing, ever-changing, patchy, and polychrome quilt*. 2025. arXiv: 2505.05197 [cs.AI]. URL: https://arxiv.org/abs/2505.05197
- † Philipp Schoenegger et al. *Large Language Models Are More Persuasive Than Incentivized Human Persuaders*. 2025. arXiv: 2505.09662 [cs.CL]. URL: https://arxiv.org/abs/2505.09662