SYLLABUS – EE 541: Spring 2026 (2 units)

A Computational Introduction to Deep Learning

Instructor:	Brandon Franzke	Office:	EEB 504B
Email:	franzke@usc.edu	Hours:	Monday: 13:00 - 14:45
			Thursday: 12:00 - 13:30

Machine learning using large datasets is the most transformative technology of the 21st century. Advances in generative ML promise solutions to almost any problem imaginable. Machine learning proficiency requires software skills as well as an understanding of the underlying mathematics and theoretical foundations. This class introduces important aspects of deep learning using a computation-first approach. It emphasizes using frameworks to solve reasonably well-defined machine learning problems.

Lecture Monday (section: 31249) 16:00 – 17:50

Enrollment is in-person ONLY. Attendance is mandatory to all lectures. Taping or recording lectures or discussions is strictly forbidden without the instructor's explicit written permission.

Teaching assistants

TA:	Xinyu Wang
Email:	xwang350@usc.edu
Office:	*see Course Website*

Course resources

Course website https://ee541.usc-ece.com.

Gradescope https://gradescope.com. Electronically submit all homeworks. You will be automatically added to Gradescope during the first week of classes. Contact Dr. Franzke with technical issues.

Course materials

[1] *Dive into Deep Learning*, Zhang, A., Lipton, Z. C., Li, M., Smola, A. J., Cambridge University Press, 2024. online, https://d2l.ai/d2l-en.pdf.

[2] *Mathematics for Machine Learning*, Deisenroth, M., Faisal, A., Ong, C., Cambridge University Press, 2020. online, https://mml-book.github.io/book/mml-book.pdf.

[3] *Deep Learning with PyTorch*, Stevens, E., Antiga, L., Viehmann, T., Manning, 2020. online, https://pytorch.org/assets/deep-learning/Deep-Learning-with-PyTorch.pdf.

[4] *Python Programming And Numerical Methods: A Guide For Engineers And Scientists*, Kong, Q., Siauw, T., Bayen, A., Elsevier, 2020. online, https://pythonnumericalmethods.berkeley.edu/notebooks/Index.html.

"AI" policy. You may use AI-powered tools in this course to enhance your learning and productivity. Use AI as a collaborative tool for understanding concepts, generating ideas, and troubleshooting. Approach AI-generated content critically and use it responsibly. Engage with AI as you would with a knowledgeable peer or tutor, using iterative conversations to deepen your understanding. You must attribute all AI-generated content in your work, including the prompts you used. You are fully accountable for the accuracy and appropriateness of any AI-assisted work. AI should supplement, not substitute, your own critical thinking and problem-solving. For assignments, you may use AI to clarify concepts or resolve issues, but submitted work must be your own. Submitting AI-generated work as your own without proper attribution or understanding is academic misconduct and will be treated as such.

You must develop complete mastery of all course material independent of AI assistance. Your knowledge and skills will be evaluated in contexts where AI tools are not accessible, mirroring real-world scenarios where you must rely solely on your own expertise. This ensures you can perform effectively in any situation, with or without AI support. Violations of this policy will result in severe academic penalties. The goal is to prepare you to use AI effectively in your future work while ensuring you develop a strong, self-reliant foundation in the course material.

Course Outline

	Topics	Required Reading	Homework
Week 1 12 Jan	Machine Learning inventory. Configuring your Python environment.	[1] Ch. 1–2, [2] Ch. 8.1–8.4, [4] Ch. 1, Ch. 12.1.	HW 1.
(19 Jan)	**No class, Martin Luther King Jr. Day.**		
Week 2 26 Jan	Getting started with Python. Numerical Python.	[4] Ch. 2–5, 10.	HW 1 due. HW 2.
Week 3 02 Feb	Estimation and MMSE.	[2] Ch. 2–4, 6, [4] Ch 14–15.	HW 3.
Week 4 09 Feb	Regression and maximum likelihood.	[1] Ch. 3, [2] Ch. 9, 11, [4] Ch 11, 16.	HW 4.
(16 Feb)	**No class, Presidents' Day.**		HW 4 due.
Week 5 23 Sep	Logistic regression. Multilayer perceptron networks (MLPs).	[1] Ch. 4–5.	HW 5.
Week 6 02 Mar	MLP backpropagation (scalar, vector/tensor).	[1] Ch. 12.1–12.5, [2] Ch. 5.	HW 6
Week 7 09 Mar	**Quiz #1 (weeks 1-6).**		HW 6 due.
(16 Mar)	**No class, Spring Break..**		
Week 8 23 Mar	PyTorch: Introduction.	[3] Ch. 3, 4.1, 5–7 [4] Ch. 7.	HW 7.
Week 9 30 Mar	PyTorch: Building MLPs.	[1] Ch. 6.	HW 8.
Week 10 06 Apr	Convolutional Neural Networks (CNN).	[1] Ch. 7.	HW 9.
Week 11 13 Apr	Project overview. Convolutional architectures.	[1] Ch. 8.1–8.6.	**Project proposal due (17 Apr)**.
Week 12 20 Apr	PyTorch: Optimizing training. Data engineering.	[1] Ch. 12.6–12.11, Ch. 14.1–14.2 [2] Ch. 10.	
Week 13 27 Apr	Auto-encoders and embedding. Recurrent neural networks (RNN).	[1] Ch. 15.1–15.4.	
Week 14 04 May	**Quiz #2 (weeks 8-13).**		
Friday 08 May	**Project deliverables, due 17:00.**		

Grading Procedure

Homework (50%). Assignments include analytic and programming problems and encourage experimentation and curiosity. Your total homework score sums your best homework scores (as a percentage) after removing the two lowest scores (of minimum 50%). You may discuss homework problems with classmates but each student must submit their own original work. Cheating warrants an "F" on the assignment. Turning in substantively identical homework solutions counts as cheating.

Late homework is accepted with a 0.5% deduction per hour, up to 48-hours — **no exceptions**. Technical issues while submitting are not grounds for extension. No submissions will be accepted 48-hours after the due date. It is your responsibility to ensure you submit the correct files and that they are accessible. Graders score what is submitted and will not follow up if the file is incorrect, incomplete, or corrupt.

Quizzes (30%). Quizzes are non-cumulative and cover the most recent material (approximately 6-weeks). They test your ability to apply major principles and demonstrate conceptual understanding. They occur during weeks 7 and 14 (tentative). You are expected to bring a scientific (non-graphing) calculator. You may use both-sides of one 3.5" $\times$ 5" reference card. You may not use any additional resources.

Quizzes include multiple-choice and short answer questions. They also include free-response or open-ended questions to demonstrate conceptual understanding. You are expected to write reasonably correct Python code as well as determine expected behavior of novel computer code. Grading primarily follows correct reasoning but may include deductions for major syntax errors, algorithmic inefficiency, or poor implementation.

Final project (20%). This course culminates with a final project in lieu of a final exam. Teams of two students apply deep-learning to problem selected from a set of instructor-defined options. Instructor defined problems will include a complete starter dataset. Teams must experiment and document network architecture search, hyper-parameter optimization, and dataset augmentation. Students should treat the final project as a multi-week in-depth homework assignment and integrate concepts from the entire semester.

Course Grade

A if 90 - 100 points, **B** if 80 - 89 points, **C** if 70 - 79 points, **D** if 60 - 69 points, **F** if 0 - 59 points.
("+" and "–" at $\approx$ 1.5% of grade boundary).

Cheating

Cheating is not tolerated on homework or exams. Penalty ranges from F on exam to F in course to recommended expulsion.

Final Project

Requirements. All projects must use PyTorch as the primary deep learning framework unless approved explicitly in writing by the instructor. Projects may use additional languages and frameworks for tooling and support. The instructor will provide additional requirements when introducing the final project assignment.

Scoring and Milestones.

Topic proposal	week 11	10%
Project report	final	35%
Model and source code		50%
Model card		5%

Project Deliverables.

Topic proposal: Restate the selected problem and include a preliminary discussion about methods, techniques, or proposed variations/modificaitons/extensions to the problem handout. Proposals should be written in anticipation of fulfilling final report requirements. But the proposal is merely a guidepost and reasonable deviations in method, approach, and scope are expected.

Written report: A properly referenced written report that incorporates all source code as well as links to any external code sources. The project report should summarize the topic, provide relevant background (theoretical or applied), timeline and contributions, and document challenges and future extensions. It should provide discussion sufficient for an uninformed expert to understand the problem, model and training decisions, and implementation. Teams should provide quantifiable metrics to justify architectural choices and engineering tradeoffs.

The report must include the following additional elements

(1) *literature survey* of work already done in your problem domain. Understand network architectures used in similar models,

(2) experimental results with various models (both ML and DL) and performance comparison. Use different architectures and hyperparameters,

(3) evidence of good data handling and processing, i.e., maintain separate test, train, and validation sets.

Source code: Submitted as GitHub repository to Gradescope. It must include README file(s) that describe the repository structure, execution instructions, and special technical requirements. It must not include any training data or model files. Details will be provided by the instructor.

Model card: A model card is a short document that provides context and transparency about your trained model. It summarizes the intended use, performance, and limitations of your model. It must include intended use, factors, metrics, evaluation data, training data, and ethical considerations.

Academic Accommodations

Any student requesting academic accommodations based on a disability is required to register with to Office of Student Accessibility Services (OSAS) each semester. A letter of verification for approved accommodations can be obtained from OSAS. Please be sure the letter is delivered to me as early in the semester as possible. OSAS is located in GFS 120 and is open 08:30 - 17:00, Monday through Friday. The phone number for DSP is (213) 740-0776.

Support Systems

A number of USC's schools provide support for students who need help with scholarly writing. Check with your advisor or program staff to find out more. Students whose primary language is not English should check with the *American Language Institute* http://dornsife.usc.edu/ali, which sponsors courses and workshops specifically for international graduate students. *The Office of Disability Services and Programs* http://sait.usc.edu/academicsupport/centerprograms/dsp/home_index.html provides certification for students with disabilities and helps arrange the relevant accommodations. If an officially declared emergency makes travel to campus infeasible, *USC Emergency Information* http://emergency.usc.edu will provide safety and other updates, including ways in which instruction will be continued by means of brightspace, teleconferencing, and other technology.

Discrimination, sexual assault, and harassment are not tolerated by the university. You are encouraged to report any incidents to the *Office of Civil Rights Compliance* http://ocrc.usc.edu or to the *Department of Public Safety* https://dps.usc.edu/contact/feedback/. This is important for the safety of the whole USC community. Another member of the university community - such as a friend, classmate, advisor, or faculty member - can help initiate the report, or can initiate the report on behalf of another person.

Academic Conduct

The University of Southern California is foremost a learning community committed to fostering successful scholars and researchers dedicated to the pursuit of knowledge and the transmission of ideas. Academic misconduct is in contrast to the university's mission to educate students through a broad array of first-rank academic, professional, and extracurricular programs and includes any act of dishonesty in the submission of academic work (either in draft or final form).

This course will follow the expectations for academic integrity as stated in the *USC Student Handbook*. All students are expected to submit assignments that are original work and prepared specifically for the course/section in this academic term. You may not submit work written by others or "recycle" work prepared for other courses without obtaining written permission from the instructor(s). Students suspected of engaging in academic misconduct will be reported to the *Office of Academic Integrity*.

Other violations of academic misconduct include, but are not limited to, cheating, plagiarism, fabrication (*e.g.*, falsifying data), knowingly assisting others in acts of academic dishonesty, and any act that gains or is intended to gain an unfair academic advantage.

Academic dishonesty has a far-reaching impact and is considered a serious offense against the university. Violations will result in a grade penalty, such as a failing grade on the assignment or in the course, and disciplinary action from the university itself, such as suspension or even expulsion.

For more information about academic integrity see the student handbook https://policy.usc.edu/studenthandbook/, or the Office of Academic Integrity's website https://academicintegrity.usc.edu/, and university policies on Research and Scholarship Misconduct https://policy.usc.edu/research-and-scholarship-misconduct/.