MATH 392: Intro to Neural Networks

Instructor: Arvind Suresh (arvindsuresh@arizona.edu)
When: Mon-Wed, 11 – 12:15 pm
Where: Modern Languages, Rm 201.
Office hours: Mon 12:30 – 1:30 and Wed 9:30 – 10:30 in MATH 319.
Credits: 3 (counts toward the 'Application Course" requirement for math majors)

Goal:
To provide students with a self-contained introduction to the mathematics and practical implementation of neural networks, which are a fundamental class of machine learning models that underlie modern AI chatbots like ChatGPT.

Course Objectives:

- Understand the key mathematical concepts used in neural networks, including linear algebra, gradient descent, and backpropagation.
- Learn to build and implement simple neural networks using libraries like PyTorch.
- Analyze and evaluate neural network models, with an emphasis on model optimization, regularization, and hyperparameter tuning.
- Gain experience in the research method (namely, asking questions and being able to hunt down answers or resources).
- Prepare for independent research by developing the ability to approach problems related to neural networks and machine learning with a solid mathematical framework.

Learning outcomes:

- Have an understanding of the basic math topics from linear algebra, calculus, probability, and statistics that are used in machine learning.
- Have an understanding of the architecture and implementation of neural networks, and associated aspects such as training, regularization, and so on.
- Learn by experience how to prepare a complete data science project addressing a question or problem.

Required materials:
There is no assigned textbook. Instead, notes will be posted every week on the class Github repo. Occasionally, readings may be assigned/suggested from chapters in freely available online textbooks. You will need a laptop to access VSCode and Github.

Important links:

- Course github repo:
  https://github.com/arvindsuresh-math/MATH-392-Intro-to-neural-networks.git
- Link to sign up for GitHub Education (which grants access to Copilot Pro for free):
  https://education.github.com/discount_requests/application

Grading:

1. We will follow the usual letter-grade scale of UA:
   A (90 – 100%), B (80 -- 89%), C (70 – 79%), D (60 – 69%), F (below 60%).
2. Requests for incomplete (I) or withdrawal (W) must be made in accordance with University policy, which is available at:
   https://catalog.arizona.edu/policy/courses-credit/grading/grading-system

Coursework:

The overall grade is comprised of four components:

1. (20%) Written homework.
   a. There will be 6-7 written homeworks covering the mathematical material.
   b. The frequency will be (on average) once every two weeks.
   c. To turn it in you can use LaTeX and submit a pdf to the assignment on D2L. (Alternatively, you can also hand-write solutions, take a picture, and upload to D2L.)
2. (20%) Class attendance.
   a. The course will be quite (very) discussion based, with us working together on problems and concepts (not so much in the style of a usual lecture format), so it's important that you attend regularly.
   b. **2 absences will be excused, no questions asked.**
   c. Note: this is ultimately quite flexible. Since our group is small, we can also try to re-schedule class on occasion if the need arises, so don't hesitate to reach out!
3. (20%) First project.
   a. The first project will involve developing a machine learning model to predict something.
   b. The exact topic will be decided in due course, and will depend on how fast we proceed in class. Thus, it may involve a simpler linear/logistic regression model, or it could be a rudimentary neural network.
   c. It will be due around week 8 or 9 (so, end-March), but there is flexibility to accommodate our respective schedules/needs.
   d. The "official submission" for the project will consist in a single zip file containing the components (Jupyter notebook, dataset, code for programs, etc.), and this zip file will be submitted to D2L. More details to be provided in due course.
   e. The goal is to create an interesting, clearly written, and well-documented project of data science that you are proud of, and can showcase to people (friends, family, employers, etc.)
4. (40%) Final project.
   a. Similar to the first project, but will be due at the end of the semester (at the start of finals week).

    b. The goal will be to implement a neural network to solve a machine learning problem.

Mathematical topics covered:

1. Linear algebra
    a. Basics, including vectors, matrices, dot products, norms, addition, scaling and multiplication operations.
    b. Tensors (as tools for representing data), and operations on tensors (illustrated with PyTorch).
    c. Time-permitting, more advanced concepts like matrix factorizations.
2. Statistics and probability theory
    a. Basics from descriptive statistics, including mean, variance, standard deviation, covariance, correlation, histograms, and visualizing data using Python.
    b. Basics on probability spaces, basic notions like conditional probability and Baye's Theorem.
    c. Random Variables and probability distributions, expected value, variance, some special distributions, Central Limit Theorem.
    d. The idea behind maximal likelihood estimation.
    e. Understanding probability concepts in the language of linear algebra.
3. Calculus
    a. Partial derivatives, Jacobians, Hessians.
    b. Gradient descent in the context of loss functions.
    c. Chain rule in the context of back-propagation.

Concepts from Machine Learning:

1. General stuff
    a. Handling and visualizing data in Python, processing data for ML (train-test splits, cross-validation).
    b. Overview of ML problems – regression and classification.
    c. Framework of supervised ML, including model selection and choice of loss functions.
    d. Theory and implementation of certain models (linear regression, logistic regression for binary classification).
    e. Bias-variance trade-off, hyperparameter tuning, regularization.
    f. Evaluating model performance.
2. Neural networks
    a. History and origins, including perceptrons and the XOR problem.
    b. General architecture of feed-forward networks, including layers, weights and biases, activation functions.
    c. Training NNs with back-propagation.
    d. High-level overview of more advanced NN architectures (Convolutional, Recurrent, Transformer).

Classroom behavior policy:

To foster a positive learning environment, I request that you refrain from accessing social media/other distractions during class-time.

Safety on campus and in the classroom:

1. For a list of emergency procedures for all types of incidents, please visit the website of the Critical Incident Response Team (CIRT): https://cirt.arizona.edu/case-emergency/overview
2. Also watch the video available at https://arizona.sabacloud.com/Saba/Web_spf/NA7P1PRD161/common/learningeventdetail/crtfy000000000003560

University-wide policies link:

Links to the following UA policies are provided at http://catalog.arizona.edu/syllabus-policies:

- Absence and Class Participation Policies
- Threatening Behavior Policy
- Accessibility and Accommodations Policy
- Code of Academic Integrity
- Nondiscrimination and Anti-Harassment Policy
- Subject to Change Statement