

CS583: Deep Learning

Instructor: Shusen Wang

Description

Deep learning (DL) is a family of the most powerful and popular machine learning (ML) methods and has wide real-world applications such as face recognition, machine translation, self-driving car, recommender system, playing the Go game, etc. This course is designed for students either with or without ML background. The course will cover the fundamental ML, computer vision (CV), and natural language (NLP) problems and DL tools for solving the problems. This is a practical course; the students will be able to use DL methods for solving real-world ML, CV, and NLP problems.

Meeting Time:

- Thursday, 6:30 - 9:00 PM, Babbio Center 221
- The classes on these dates are canceled: Jan 31, Feb 28

Office Hours:

- Thursday, 3:00 - 5:00 PM, North Building 205
- The office hours on these dates are canceled: Jan 31, Feb 28

Schedule

- Jan 24, Lecture 1
 - Fundamental ML problems
 - Regression
- Jan 24, **Homework 0** is assigned.
 - [Click here for the assignment](#)
 - Submission is not required.
 - Deadline: finish it before the 1st Quiz. (Otherwise, you might fail.)
- Jan 24, **Homework 1** is assigned (available on Canvas).
 - Submission: submit to Canvas.
 - Deadline: Feb 19.
- Jan 31, **CANCELED** due to the instructor's conference traveling
- Feb 7, Lecture 2
 - Regression (cont.)
 - Classification
- Feb 14, Lecture 3
 - Classification (cont.)
 - Dimensionality reduction
 - The 1st **Quiz**.
- Feb 14, **Homework 2** is assigned (available on Canvas).
 - Submission: submit to Canvas.
 - Deadline: Mar 12.
- Feb 21, Lecture 4
- Feb 28, **CANCELED** due to the instructor's conference traveling

Syllabus and Slides

1. **Machine learning basics.** This part briefly introduces the fundamental ML problems— regression, classification, dimensionality reduction, and clustering— and the traditional ML models and numerical algorithms for solving the problems.
 - ML basics. [slides]
 - Regression. [slides-1 slides-2]
 - Classification. [slides-1 slides-2 slides-3 slides-4]
 - Regularizations. [slides-1 slides-2]
 - Dimensionality reduction. [slides-1 slides-2 slides-3]
2. **Neural network basics.** This part covers the multilayer perceptron, backpropagation, and deep learning libraries, with focus on Keras.
 - Multilayer perceptron. [slides to appear]
 - Keras. [slides to appear]
3. **Convolutional neural networks (CNNs).** This part is focused on CNNs and its application to computer vision problems.
 - CNN basics. [slides to appear]
 - Advanced topics on CNNs. [slides to appear]
 - Popular CNN architectures. [slides to appear]
 - Style transfer. [slides to appear]
4. **Autoencoders.** This part introduces autoencoders for dimensionality reduction and image generation.
 - Autoencoder for dimensionality reduction.[slides to appear]
 - Variational Autoencoders (VAEs) for image generation. [slides to appear]
5. **More image generation methods.** This part covers two image generation approaches in addition to VAE. The training of neural networks takes fixed images (X) as inputs and optimize w.r.t. the weights (W). Conversely, one can fix the network's weights (W) and optimize w.r.t. the input X . In this way, new images are generated to maximize (or minimize) some function; the application includes attacking neural networks and deep dream. Another very different approach is the generative adversarial network (GAN).
 - Attack neural networks. [slides to appear]
 - Deep dream. [slides to appear]
 - Generative adversarial network (GAN). [slides to appear]
6. **Recurrent neural networks (RNNs).** This part introduces RNNs and its applications in natural language processing (NLP).
 - Text processing. [slides to appear]
 - RNN basics and LSTM. [slides to appear]
 - Text generation. [slides to appear]
 - Machine translation. [slides to appear]
7. **Recommender system.** This part is focused on the collaborative filtering approach to recommendation based on the user-item rating data. This part covers matrix completion methods and neural network approaches.
 - Recommender system. [slides to appear]

Project

Every student must participate in one **Kaggle competition**.

- **Details:** [click here](#)
- **Deadlines**
 - (Not required.) Submit a preliminary version to Canvas before April 21th if you want to compete for the presentation and bonus.
 - Submit the final version to Canvas before June 1st.
- **Submissions:** Put your source code and report to a Github repository and submit the links to Canvas.
- **Teamwork policy:** You had better work on your own project. Teamwork (up to 3 students) is allowed if the competition has a heavy workload; the workload and team size will be considered in the grading.

Textbooks

Required:

- Francois Chollet. Deep learning with Python. Manning Publications Co., 2017. (Available online.)

Recommended:

- Y. Nesterov. Introductory Lectures on Convex Optimization Book. Springer, 2013. (Available online.)
- D. S. Watkins. Fundamentals of Matrix Computations. John Wiley & Sons, 2004.
- I. Goodfellow, Y. Bengio, A. Courville, Y. Bengio. Deep learning. MIT press, 2016. (Available online.)
- M. Mohri, A. Rostamizadeh, and A. Talwalkar. Foundations of machine learning. MIT press, 2012.
- J. Friedman, T. Hastie, and R. Tibshirani. The elements of statistical learning. Springer series in statistics, 2001. (Available online.)

Grading Policy

Grading percentages:

- Homework 50%
- Final 15%
- Project 20%
- Quizzes 15%
- Bonus (up to 10%)

Late penalty:

- Late submissions of assignments or project document for whatever reason will be punished. 1% of the score of an assignment/project will be deducted per day. For example, if an assignment is submitted 15 days and 1 minute later than the deadline (counted as 16 days) and it gets a grade of 95%, then the score after the deduction will be: $95\% - 16\% = 79\%$.
- June 1 is the firm deadline of all the homework. Submissions later than June 1 will not be accepted.