CSCI 5525 Machine Learning: Analysis and Methods

Spring 2021

General Information

What are state-of-the-art machine learning methods, and why they work? This graduate-level introductory course to machine learning focuses on the foundations of modern machine learning. We will cover selected topics from supervised learning, unsupervised learning, and interactive learning. For each topic, key algorithmic ideas/intuitions and basic theoretical insights will be highlighted. The end goal is that the students will be able to develop and deploy novel learning methods for their applications, and potentially derive basic theoretical understanding.

- **Prerequisite:** Introduction to machine learning or equivalent (CSCI5521 or 5523) are **highly recommended**. Maturity in linear algebra, multivariate calculus, and basic probability is assumed. Familiarity with Python (esp. Numpy, Scipy) is necessary to complete the homework assignments and final projects.
- **When & Where:** Mon/Wed 1:00–2:15pm, Keller 3-210
- **Who:**

  Professor Ju Sun (Instructor) Email: jusun@umn.edu
  Le Peng (TA) Email: peng0347@umn.edu
  Tiancong Chen (TA) Email: chen6271@umn.edu

- **Office Hours:** Tue/Thur 5–6pm (Instructor) Mon/Wed 3–4pm (Le) Fri 3–4pm (Tiancong)
- **Course Website:** https://sunju.org/teach/ML-Spring-2021/ All course materials, including course schedule, lecture slides and notes, homework assignments, project description, will be posted on the course website. Enrolled students are encouraged to check the website on a regular basis.
- **Communication—minimizing emails: Canvas** is the preferred and most efficient way of communication. Please post all questions and discussions related to the course in Canvas instead of sending emails. If you have to use emails, please begin the subject line with "CSCI 5525", and expect delay in response.

Tentative Topics

The list of topics is subject to change.

- Overview
- Linear prediction
- Supervised learning and generalization
- Support vector machines and kernel methods
- Boosting
- Decision trees

- Neural networks
- Dimension reduction: PCA and kernel PCA
- Unlock relations: ICA and CCA
- Clustering (spectral, hierarchical)
- Generative models (GMM, GAN, variational autoencoder)
- Online/Reinforcement learning

Teaching Format

For the Spring semester, all enrolled students will be able to view live-streaming and downloadable video archives of class meetings with no delay for the length of the semester through the UNITE Media Portal. The video archives will also be available from within Canvas under "Media Gallery".

Access the media for your class through the "LOG IN to UNITE Media Portal" link in the upper right of the UNITE website (www.unite.umn.edu).

Your University Internet ID (a.k.a. "x.500") and password are required (this is what you use to access Canvas, your University of Minnesota email account, etc.). - log in and click on the link for your course. Access these videos through the UNITE Media Portal with your University of Minnesota Internet I.D. and password.

DO NOT ask the instructor or teaching assistants for technical or troubleshooting assistance with these streaming video archives – use the UNITE Troubleshooting FAQ (https://cse.umn.edu/unite/troubleshoot-unite-media) or "Submit a Trouble Report to UNITE" through the link found on all pages within the UNITE Media Portal.

Live streaming video has latency based on UNITE compression and your individual stable internet connection speed, typically ranging from 30 to 60 seconds. The archived UNITE media is typically available within an hour after the end of class meetings.

Recommended References

There is no required textbook. Lecture notes and additional notes will be the primary resources. Recommended reference books are the following, all of which are freely available either online or through library e-access.

- Advanced machine learning textbooks
  - **Foundations of Machine Learning** (2e) by Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. https://cs.nyu.edu/~mohri/mlbook/
  - **Understanding Machine Learning: From Theory to Algorithms** by Shai Shalev-Shwartz and Shai Ben-David. https://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/
  - **The Elements of Statistical Learning: Data Mining, Inference, and Prediction** by Trevor Hastie, Robert Tibshirani, and Jerome Friedman. https://web.stanford.edu/~hastie/ElemStatLearn/
  - **Machine Learning : A Probabilistic Perspective** by Kevin P. Murphy. click here (UMN library access; login required)

- Mathematics foundations
  - **Mathematics for Machine Learning** by Marc Peter Deisenroth, A. Aldo Faisal, and Cheng Soon Ong. https://mml-book.github.io/
  - **Linear Algebra and Optimization for Machine Learning** by Charu C. Aggarwal. https://rd.springer.com/book/10.1007/978-3-030-40344-7 (UMN library access; login required)
- Hand-on in Python
  - **Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems** by Aurélien Géron (2ed). O'Reilly Media, 2019. click here (UMN library access; login required)

Assessment

- Homework 60%: 6 Homeworks (12% each), the top 5 scores will count toward the final grade
- Takehome mid-term 15%
- Course project 25%: proposal (5%) + final report (20%). The project can be survey of a chosen topic not covered in detail in the class, implementation and comparison of existing methods, or novel foundational or applied research

Homework

You have approximately 10 days time to complete each homework (exact due date will be specified in each assignment). No late submissions will be accepted. All submissions *must* be electronic and uploaded via the Canvas system. Written part should be neatly written or typeset and *must* be submitted as PDF files. For students pursuing research in relevant areas, you are strongly encouraged to typeset your solution using LaTeX. Computer programs *must* be submitted in the Python notebook format. Only Python 3 will be used and accepted in this course.

Collaboration on homework problems is strongly encouraged, but each student must ensure that the final submission is prepared individually. *Collaborators should be properly acknowledged in the final submission, at the problem level.* The same applies to computer programs. Plagiarism and cheating will not be tolerated and are subject to disciplinary action. Please consult the student code of conduct for more information: https://regents.umn.edu/sites/regents.umn.edu/files/2019-09/policy_student_conduct_code.pdf

Course Project

The course project is to be performed by teams of 2 students. All students from the same team will get the same score for their course project.

Programming and Computing

Our programming environment will be Python 3. For deep learning frameworks, PyTorch (≥ 1.0) is preferred, but Tensorflow (≥ 2.0) is also accepted and supported. For small-scale experiments, which will be the case for homework assignments, Google Colab (https://colab.research.google.com/) and UMN MSI notebook service will suffice. Local installation of the relevant software packages may be a reasonable alternative. For large-scale course projects, we can use the Minnesota Supercomputing Institute (MSI) GPU computing queues based on our class account.

Other Policies

Please consult this policy link https://policy.umn.edu/education/syllabusrequirements-appa