CSC 580: Principles of Machine Learning

M W 3:30pm

Revised: August 19, 2020

Description of Course

Students will learn why machine learning is a fundamentally different way of writing computer programs from traditional programming, and why this is often an attractive way of solving practical problems. Machine learning is all about automatic ways for computers to find patterns in datasets; students will learn both advantages and unique risks that this approach offers. They will learn the fundamental computational methods, algorithms, and perspective which underlie current machine learning methods, and how to derive and implement many of them.

Course Prerequisites or Co-requisites

- Linear algebra or equivalent
    - You will need to understand the relationship between linear operators, linear transformations, change of bases, and matrices
    - We will make repeated use of matrix decompositions such as the SVD. Often we will need to use properties of the eigendecomposition of a matrix.
- Multivariate calculus or equivalent:
    - You will need to understand the relationship between the total derivative, the gradient, and how to take advantage of the fact that the derivative is a linear operator
- Probability and statistics:
    - You will need to understand (conditional) expectation and (conditional) independence of random variables.
- Programming
    - You will need a good amount of programming experience: a programming maturity is expected.

Instructor and Contact Information

- Kwang-Sung Jun
- kjun@cs.arizona.edu
- Gould-Simpson 746

- Office Hour: Tuesday 4-5pm. The instructor is also available by appointment.
- Course homepage: https://kwangsungjun.github.io/teach/20.2.csc580
- Gradescope: https://www.gradescope.com/courses/163532 , entry code: MXD4D2
- D2L: https://d2l.arizona.edu/d2l/home/947630

Course Format and Teaching Methods

Synchronous online lectures (zoom), individual assignments, written exams, projects, in-class discussions.

Course Objectives and Expected Learning Outcomes

A successful student will be able to implement and explain the limitations of many of the central methods and techniques in machine learning:

- Basic binary classifiers: decision trees, logistic regression
- Supervised vs. unsupervised learning - what's possible in the absence of labels
- Reductions - how to handle imbalanced data; how to build multiclass classifiers
- Practical issues - how to detect overfitting and underfitting; how and when to use feature engineering
- Efficiency issues - how to create classifiers that work well in the presence of large training sets, and large feature sets
- Modern techniques - students will be introduced, via classroom materials and projects, to recent methods in machine learning (this could include, for example, deep learning, reinforcement learning, A/B testing, and multi-armed bandits)

For a more granular description of the learning objectives, see the week-by-week schedule and the description of the assignments below.

Machine Learning is a big field, and there is no way we can cover all of it in one course. With that said, this course covers a large amount of material, and the assignments are a central part of the course. **Students are expected to dedicate a significant amount of time on the course outside of the classroom, especially if they have background deficiencies to make up**.

Expected Learning Outcomes

The expected learning outcomes of the course is:

- To be able to explain why supervised learning is expected to generalize.
- To be able to detect overfitting and underfitting.
- To be able to explain the risk of test set reuse.
- To be able to compute the error bar and discuss statistical significance of a given evaluation result.
- To be able to list the common supervised and unsupervised learning algorithms and the practical values of each.
- To be able to explain the key challenge in reinforcement learning that supervised/unsupervised learning paradigms did not have to worry about.

Absence and Class Participation Policy

The UA's policy concerning Class Attendance, Participation, and Administrative Drops is available at http://catalog.arizona.edu/policy/class-attendance-participation-and-administrative-drop. The UA policy regarding absences for any sincerely held religious belief, observance or practice (http://policy.arizona.edu/human-resources/religious-accommodation-policy) will be accommodated where reasonable. Absences pre approved by the UA Dean of Students (or dean's designee) will be honored. See the dean of students's website (https://deanofstudents.arizona.edu/absences) for details.

Makeup Policy for Students Who Register Late

If you register late for this class, contact me as soon as you do. You will be expected to submit all missed assignments within a week of your registration. It is your responsibility to catch up to the class content.

Course Communications

We will use D2L for communications and discussion. Make sure your D2L account is up to date - class announcements are sent through the website.

Required Texts or Readings

The required textbook is Hal Daumé's Course in Machine Learning (http://ciml.info/), fully and freely available online.

Assignments and Examinations: Schedule/Due Dates

- 08/31: HW1
- 09/21: HW2
- 10/14: HW3
- 10/21: Project proposal
- 10/30: HW4
- 11/18: HW5
- 12/14: Final exam (take home)
- 12/17: Final project due

Final Examination or Project

The final exam will consist of the learned material in the class. The final project for the course will involve the implementation of a recently-published machine learning technique of the student's choice, subject to the approval of the instructor. The instructor will provide a list of suggested techniques for students who prefer not to make the choice from scratch. The instructor expects these techniques to be drawn from top research venues in the field, such as NeurIPS, ICML, ICLR, CVPR, ICCV, NAACL, EMNLP etc.

Grading Scale and Policies

As mentioned above, you will be assessed based on your performance on programming assignments, one final exam, and one project.

The instructing staff will grade your assignments, project, midterms, and final exam on a scale from 0 to 100, with the following weights:

- Assignments: 50%
- Project: 25%
- Final Exam: 25%

Your final grade in the course of be the best of a per-class grading curve and overall performance:

- 90% or better: A;
- 80% or better: B;
- 70% or better: C;
- 60% or better: D;
- below 60%: F.

By your last day to withdraw, you will know more than 40% of your grade by weight.

Graded homework will be returned before the next homework is due. The homework is due in 10 days and will be returned to students before the next homework is due. Exams will be returned within two weeks. Grading delays beyond promised return-by dates will be announced as soon as possible with an explanation for the delay. As a rule, homework will not be accepted late except in case of documented emergency or illness.

There will be a total of 5 math/programming assignments and 1 project proposal assignment (total 6), 4 of them will be before the end of 10th week and you will know their grades. There will be a project proposal due by the end of 10th week. Each assignment will be due at least one week after it is posted. The final project report is due at the last day of class.

Requests for incomplete (I) or withdrawal (W) must be made in accordance with University policies, which are available at http://catalog.arizona.edu/policy/grades-and-grading-system#incomplete and http://catalog.arizona.edu/policy/grades-and-grading-system#Withdrawal, respectively. Dispute of Grade Policy: If you wish to dispute your grade for an assignment, midterm or project, you have two weeks after the grade has been turned in. In addition, even if only you dispute one portion of the grading for that unit, I reserve the right to revisit the entire unit (assignment, midterm, or project).

Scheduled Topics/Activities

See the Schedule (https://kwangsungjun.github.io/teach/20.2.csc580/schedule.html) page for details.

- Week 1
    - Lecture: Introduction, motivation, course mechanics
    - Lecture: Basics - Decision Trees, algorithms for learning
        - Learning Objectives. Explain the difference between memorization and generalization • Implement a decision tree classifier • Take a concrete task and cast it as a learning problem, with a formal notion of input space, features, output space, generating distribution and loss function.
- Week 2
    - Lecture: Limits - Optimal Bayes rate and classifier; overfitting and underfitting
        - Learning Objectives. Define “inductive bias” and recognize the role of inductive bias in learning • Illustrate how regularization trades off between underfitting and overfitting • Evaluate whether a use of test data is “cheating” or not.
    - Lecture: Geometry, nearest-neighbor classifiers, k-means (unsupervised learning preview)
        - Learning Objectives • Describe a data set as points in a high dimensional space • Explain the curse of dimensionality • Compute distances between points in high dimensional space • Implement a K-nearest neighbor model of learning • Implement the K-means algorithm for clustering.
    - A1: Decision Trees, Naive nearest neighbors, k-means
- Week 3
    - Lecture: The perceptron (1/2)
        - Learning Objectives. Describe the biological motivation behind the perceptron • Classify learning algorithms based on whether they are error-driven or not • Implement the perceptron algorithm for binary classification • Draw perceptron weight vectors and the corresponding decision boundaries in two dimensions • Contrast the decision boundaries of decision trees, nearest neighbor algorithms and perceptrons • Compute the margin of a given weight vector on a given data set.
    - Lecture: The perceptron (2/2)
- Week 4
    - Lecture: Practical Issues (1/2) - performance measures, underfitting, overfitting, cross validation, prediction confidence via statistical tests and bootstrapping, debugging ML models
        - Learning Objectives. Translate between a problem description and a concrete learning problem • Perform basic feature engineering on image and text data • Explain how to use cross-validation to tune

hyperparameters and estimate future performance • Compare and contrast the differences between several evaluation metrics.
  - Lecture: Practical Issues - (2/2)
  - A2: Perceptron and Feature Selection
- Week 5
  - Lecture: Bias-variance decomposition, and friends
    - Learning Objectives. Understand how classification errors naturally split in approximation error and estimation errors • Understand how error decompositions are useful for debugging.
  - Lecture: Reductions (1/3)
    - Learning Objectives. Represent complex prediction problems in a formal learning setting • Be able to artificially “balance” imbalanced data • Understand the positive and negative aspects of several reductions from multiclass classification to binary classification • Recognize the difference between regression and ordinal regression.
- Week 6
  - Lecture: Reductions (2/3)
  - Lecture: Reductions (3/3)
- Week 7
  - Lecture: Linear Models (1/2)
    - Learning Objectives. Define and plot four surrogate loss functions: squared loss, logistic loss, exponential loss and hinge loss • Compare and contrast the optimization of 0/1 loss and surrogate loss functions • Solve the optimization problem for squared loss with a quadratic regularizer in closed form • Implement and debug gradient descent and subgradient descent
  - Lecture: Linear Models (2/2)
  - A3: Reduction / Linear Models
- Week 8
  - Lecture: Kernel Methods (1/2)
    - Learning Objectives. Explain how kernels generalize both feature combinations and basis functions • Contrast dot products with kernel products • Implement kernelized perceptron • Derive a kernelized version of regularized least squares regression • Implement a kernelized version of the perceptron • Derive the dual formulation of the support vector machine.
  - Lecture: Kernel Methods (2/2)
  - A4: Project proposal
- Week 9
  - Lecture: Probability, Naive Bayes, and Graphical Model Basics (1/3)
    - Learning Objectives. Define the generative story for a naive Bayes classifier • Derive logistic loss with an l2 regularizer from a probabilistic perspective.
  - Lecture: Probability, Naive Bayes, and Graphical Model Basics (2/3)

- Week 10
    - Lecture: Probability, Naive Bayes, and Graphical Model Basics (3/3)
    - Lecture: Bias and Fairness
        - Learning Objectives. Identify how disparity along training/test data can generate bias/unfairness • Understand how a bad choice of metric to optimize can cause bias/unfairness • Identify how careless data collection practices can perpetuate bad decisions • Identify how different assumptions about the world change the way data should be processed for an ML method • Understand how feedback loops can cause arbitrarily bad predictions
    - A5: Kernel Methods / Probability
- Week 11
    - Lecture: Neural Networks and Back-Propagation (1/2)
        - Learning Objectives. Explain the biological inspiration for multi-layer neural networks • Construct a two-layer network that can solve the XOR problem • Implement the back-propagation algorithm for training multi-layer networks • Explain the trade-off between depth and breadth in network structure • Contrast neural networks with radial basis functions with k-nearest neighbor learning.
    - Lecture: Neural Networks and Back-Propagation (2/2)
- Week 12
    - Lecture: Ensembling
        - Learning Objectives. Implement bagging and explain how it reduces variance in a predictor • Explain the difference between a weak learner and a strong learner • Derive the AdaBoost algorithm • Understand the relationship between boosting decision stumps and linear classification.
    - Lecture: Efficiency
        - Learning Objectives. Understand and be able to implement stochastic gradient descent algorithms • Compare and contrast small versus large batch sizes in stochastic optimization • Derive subgradients for sparse regularizers • Implement feature hashing.
    - A6: Ensembling and Efficiency
- Week 13
    - Lecture: Unsupervised Learning (1/2)
        - Learning Objectives. Explain the difference between linear and non-linear dimensionality reduction • Relate the view of PCA as maximizing variance with the view of it as minimizing reconstruction error • Implement latent semantic analysis for text data • Motivate manifold learning from the perspective of reconstruction error • Understand K-means clustering as distance minimization • Explain the importance of initialization in k-means and furthest-first heuristic • Implement agglomerative clustering • Argue whether spectral clustering is a clustering algorithm or a dimensionality reduction algorithm.
    - Lecture: Unsupervised Learning (2/2)

- Week 14
  - Lecture: Learning Theory (1/2)
    - Learning Objectives. Explain why inductive bias is necessary • Define the PAC model and explain why both the “P” and “A” are necessary • Explain the relationship between complexity measures and regularizers • Identify the role of complexity in generalization • Formalize the relationship between margins and complexity
  - Lecture: Learning Theory (2/2)
- Week 15
  - (Reserved for catch-up)
  - (Reserved for catch-up)

Department of Computer Science Code of Conduct

The Department of Computer Science is committed to providing and maintaining a supportive educational environment for all. We strive to be welcoming and inclusive, respect privacy and confidentiality, behave respectfully and courteously, and practice intellectual honesty. Disruptive behaviors (such as physical or emotional harassment, dismissive attitudes, and abuse of department resources) will not be tolerated. The complete Code of Conduct is available on our department web site. We expect that you will adhere to this code, as well as the UA Student Code of Conduct, while you are a member of this class.

Classroom Behavior Policy

To foster a positive learning environment, students and instructors have a shared responsibility. We want a safe, welcoming, and inclusive environment where all of us feel comfortable with each other and where we can challenge ourselves to succeed. To that end, our focus is on the tasks at hand and not on extraneous activities (e.g., texting, chatting, reading a newspaper, making phone calls, web surfing, etc.). Students are asked to refrain from disruptive conversations with people sitting around them during lecture. Students observed engaging in disruptive activity will be asked to cease this behavior. Those who continue to disrupt the class will be asked to leave lecture or discussion and may be reported to the Dean of Students. Some learning styles are best served by using personal electronics, such as laptops and iPads. These devices can be distracting to other learners. Therefore, students who prefer to use electronic devices for note-taking during lecture should use one side of the classroom.

Threatening Behavior Policy

The UA Threatening Behavior by Students Policy prohibits threats of physical harm to any member of the University community, including to oneself.
See http://policy.arizona.edu/education-and-student-affairs/threatening-behavior-students.

Notification of Objectionable Materials

This course will contain material of a mature nature, which may include explicit language, depictions of nudity, sexual situations, and/or violence. The instructor will provide advance notice when such materials will be used. Students are not automatically excused from interacting with such materials, but they are encouraged to speak with the instructor to voice concerns and to provide feedback.

Accessibility and Accommodations

At the University of Arizona we strive to make learning experiences as accessible as possible. If you anticipate or experience physical or academic barriers based on disability or pregnancy, you are welcome to let me know so that we can discuss options. You are also encouraged to contact Disability Resources (520-621-3268) to explore reasonable accommodation. If our class meets at a campus location: Please be aware that the accessible table and chairs in this room should remain available for students who find that standard classroom seating is not usable.

Code of Academic Integrity

Students are encouraged to share intellectual views and discuss freely the principles and applications of course materials. However, graded work/exercises must be the product of independent effort unless otherwise instructed. Students are expected to adhere to the UA Code of Academic Integrity as described in the UA General Catalog.
See http://deanofstudents.arizona.edu/academic-integrity/students/academic-integrity. The University Libraries have some excellent tips for avoiding plagiarism, available at http://new.library.arizona.edu/research/citing/plagiarism.

UA Nondiscrimination and Anti-harassment Policy

The University is committed to creating and maintaining an environment free of discrimination; see http://policy.arizona.edu/human-resources/nondiscrimination-and-anti-harassment-policy. Our classroom is a place where everyone is encouraged to express well-formed opinions and their reasons for those opinions. We also want to create a tolerant and open environment where such opinions can be expressed without resorting to bullying or discrimination of others.

Additional Resources for Students

UA Academic policies and procedures are available at http://catalog.arizona.edu/policies Student Assistance and Advocacy information is available at http://deanofstudents.arizona.edu/student-assistance/students/student-assistance

Confidentiality of Student Records

Please see http://www.registrar.arizona.edu/personal-information/family-educational-rights-and-privacy-act-1974-ferpa?topic=ferpa for information on confidentiality of student records. This has concrete consequences for you if you give my name as a reference! In other words, if you intend to give my name as a reference, please contact me ahead of time so we can discuss.

Subject to Change Statement

Information contained in the course syllabus, other than the grade and absence policy, may be subject to change with advance notice, as deemed appropriate by the instructor.

Crisis Contacts

Where to go, who to call if you're in crisis:

- Located in Tucson? Call the Community-Wide Crisis Line 24 hours a day, 7 days a week at 520-622-6000.
- Are you a University of Arizona student? If it is not an emergency and you are a UA student, call or walk-in to Counseling and Psych Services at 520-621-3334 Monday - Friday. Walk-in triage is available between 9 am and 4 pm Monday - Friday.
- Are you a concerned friend? Concerned friends can find out more about helping a friend who might be experiencing problems through our Friend 2 Friend website.

Resources for sexual assault, relationship violence, and stalking.

24-Hour Hotlines:

- The National Suicide Prevention Lifeline is a 24-hour, toll-free, confidential suicide prevention hotline available to anyone in suicidal crisis or emotional distress. By dialing 1-800-273-TALK (8255), the call is routed to the nearest crisis center in our national network of more than 150 crisis centers. The Lifeline's national network of local crisis centers provides crisis counseling and mental health referrals day and night.
- Crisis Text Line: Text HOME to 741741 from anywhere in the United States, anytime, about any type of crisis. A live, trained Crisis Counselor receives the text and responds, all from a secure online platform. Find out more about how it works at crisistextline.org.
- Suicide Prevention for LGBTQ Youth through the Trevor Project:
    - The Trevor Lifeline is a 24/7 suicide hotline: 866-4-U-TREVOR (1-866-488-7386)
    - TrevorChat: Online instant messaging available 7 days a week, 3 pm - 10 pm ET (12 pm – 7 pm PT)
    - TrevorText: Confidential and secure resource that provides live help for LGBTQ youth with a trained specialist, over text messages. Text TREVOR to 1-202-304-1200 (available 7 days a week, 3 pm - 10 pm ET, 12 pm – 7 pm PT)
- Veterans' Suicide Prevention Lifeline: 1-800-273-TALK (1-800-273-8255)

- SAMHSA Treatment Referral Hotline (Substance Abuse): 1-800-662-HELP (1-800-662-4357)
- National Sexual Assault Hotline: 1-800-656-HOPE (1-800-656-4673)
- Loveisrespect (National Dating Abuse Helpline): Call 1-866-331-9474 (TTY: 1-866-331-8453).
- Text LOVEIS to 22522 - you'll receive a response from a peer advocate prompting you for your question. Go ahead and text your comment or question and we will reply.