CS 4641 Machine Learning
Fall 2017

Zsolt Kira
Architecture (East) 123

TAs:
Namkha Norsang (Head TA)
Shivam Agarwal
Hongzhao Guan
Andrea Hu
Varsha Partha
Erik Wijmans

Office Hours:
Instructor: CCB 260 Thursday 12pm - 1pm
TAs: Please see Piazza post #9 for updated locations

Required Text:

Machine Learning
by Tom Mitchell, McGraw Hill, 1997

Schedule

Week	Topic	Reading	Slides (on t-square)	Assignments
1 (08/21)	Introduction to ML, Decision Trees	Linear Algebra Mitchell Ch. 1 & 3 Recommended: Book exercises Recommended: Udacity ML Overview and Decision Trees	01, 02	
2 (08/28)	Linear Regression, Neural Networks	Mitchell Ch. 4 Recommended: Book exercises, Recommended: Udacity Regression and Classification, Neural Networks	03	Assignment 1 Released
3 (09/04)	Instance-Based Learning, Boosting and Support Vector Machines	Mitchell Ch. 8 Schapire Intro Jiri Matas and Jan Sochman's Slides Recommended: Book exercises, Recommended: Udacity Instance-based Learning, Ensemble Learning		
4 (09/11)	Support Vector Machines and Boosting Bayesian Learning	ICML Tutorial Burges Tutorial Scholkopf NIPS Tutorial Mitchell Ch. 6 Recommended: Udacity Kernel Methods		
5 (09/18)	Bayesian Learning and Inference	Mitchell Ch. 7 Recommended: Udacity Bayesian Learning, Bayesian Inference		Assignment 1 due (09/20)
6 (09/25)	Bayesian Inference, Randomized Optimization	Recommended: Udacity Randomized Optimization		Assignment 2 released (Thursday)
7 (10/02)	Randomized Optimization, Computational Learning Theory, VC Dimensions	Mitchell Ch. 9 No Free Lunch Theorem Recommended: Udacity Computational Learning Theory, VC Dimensions		
8 (10/09)	Note: Holiday 10/10 Mid-term Review (NOTE: Mid-term moved to 10/17)			
9 (10/16)	Mid-Term 10/17, Clustering	Intuitive Explanation of EM Statistical View of EM Recommended: Udacity Clustering		
10 (10/23)	Mid-term review, EM, impossibility, and no-free lunch	Jon Kleinberg's Impossibility Theorem for Clustering Recommended: Udacity Feature Selection		Assignment 2 due (10/22)
11 (10/30)	Feature selection and transformation/dimensionality reduction	ICA Recommended: Udacity Feature Transformation		
12 (11/6)	Markov Decision Processes	Mitchell Ch. 13 Recommended: Udacity Markov Decision Processes		
13 (11/13)	Reinforcement Learning	Mitchell Ch. 13 Reinforcement Learning Survey Recommended: Udacity Reinforcement Learning		Assignment 3 due (unsupervised learning)
14 (11/20)	Reinforcement Learning			
15 (11/27)	Deep Learning and Deep Reinforcement Learning	Deep Learning book Ch. 6 (MLPs) and Ch. 9 (CNNs)		
16 (12/4)	DL and DRL continued, wrapup NOTE: Final exam 12/14	Playing Atari with Deep Reinforcement Learning Human-Level Deep Reinforcement Learning Optional: Mastering the game of Go with deep neural networks and tree search		Assignment 4 due 12/11 (reinforcement learning)

General Information

Note: This course will largely follow that of past courses taught by Charles Isbell. All materials courtesy of the previous course as well (including this syllabus!). Materials will largely follow past years' organization, although towards the end of the course it is more advanced/open-ended and the topics may change (e.g. we will likely not cover game theory).
Machine Learning is a three-credit course on, well, Machine Learning. Machine Learning is that area of Artificial Intelligence that is concerned with computational artifacts that modify and improve their performance through experience. The area is concerned with issues both theoretical and practical. This particular class is a part of a series of classes in the Intelligence thread, and as such takes care to present algorithms and approaches in such a way that grounds them in larger systems. We will cover a variety of topics, including: statistical supervised and unsupervised learning methods, randomized search algorithms, Bayesian learning methods, and reinforcement learning. The course also covers theoretical concepts such as inductive bias, the PAC and Mistake-bound learning frameworks, minimum description length principle, and Ockham's Razor. In order to ground these methods the course includes some programming and involvement in a number of projects.

Objectives

There are four primary objectives for the course:

- To provide a broad survey of approaches and techniques in ML
- To develop a deeper understanding of several major topics in ML
- To develop the design and programming skills that will help you to build intelligent, adaptive artifacts
- To develop the basic skills necessary to pursue research in ML

The last objective is the core one: you should develop enough background that you can pursue any desire you have to learn more about specific techniques in ML, either to pursue ML as a research career, or to apply ML techniques in other research areas in interesting (as opposed to uninteresting) ways.

Prerequisites

The official prerequisite for this course is an introductory course in artificial intelligence. In particular, those of you with experience in a general representational issues in AI, some AI programming, and at least some background (or barring that, willingness to pick up some background) in statistics and information theory should be fine. Any student who did well in an AI course like this one should be fine. You will note that the syllabus for that particular course suggests at least some tentative background in some machine learning techniques as well. Having said all that, the most important prerequisite for enjoying and doing well in this class is your interest in the material. I say that every semester and I know it sounds trite, but it's true. In the end it will be your own motivation to understand the material that gets you through it more than anything else. If you are not sure whether this class is for you, please talk to me.

Resources

- **Readings.** The textbook for the course is *Machine Learning* by Tom Mitchell. We will follow the textbook quite closely for most of the semester, so it is imperative that you have a copy of the book. We will also use supplemental readings as well, but those will be provided for you.
- **Computing.** You will have access to CoC clusters for your programming assignments. You are free to use whatever machines you want to do your work; however, the final result will have to run on the standard CoC boxes. Exactly what this means will be spelled out. This shouldn't be much of a restriction for you.
- **Web.** We will use the class web page to post last minute announcements, so check it early and often.

Statement of Academic honesty

At this point in your academic careers, I feel that it would be impolite to harp on cheating, so I won't. You are all adults, more or less, and are expected to follow the university's code of academic conduct (you know, the honor code). Furthermore, at least some of you are researchers-in-training, and I expect that you understand proper attribution and the importance of intellectual honesty.

Please note that unauthorized use of any previous semester course materials, such as tests, quizzes, homework, projects, lectures, and any other coursework, is prohibited in this course. In particular, you are not allowed to use old exams. Using these materials will be considered a direct violation of academic policy and will be dealt with according to the GT Academic Honor Code. Furthermore, I do not allow copies of my exams out in the ether (so there should not be any out there for you to use anyway). My policy on that is strict. If you violate the policy in any shape, form or fashion you will be dealt with according to the GT Academic Honor Code. I also have several... friends... from Texas who will help me personally deal with you.

Readings and Lectures

My research area is machine learning, and I'm deeply into the area. Given that and my enormous lung capacity, and my tendency to get distracted, it turns out that I can ramble on about the material for days on end; however, that rather misses the point.

Lectures are meant to summarize the readings and stress the important points. You are expected to come to class having already critically read any assigned material. Your active participation in class is crucial in making the course successful. I completely expect to be interrupted throughout a lecture with questions and maybe even your deep insights into the material. This is less about my teaching than about your learning. My role is merely to assist you in the process of learning more about the area.

Grading

Your final grade is divided into three components: assignments, a midterm and a final exam.

- **Assignments.** There will be four graded assignments. They will be about programming and analysis. Generally, they are designed to give you deeper insight into the material and to prepare you for the exams. The programming will be in service of allowing you to run and discuss experiments, do analysis, and so on. In fact, the programming is incidental, as you shall see.
- **Midterm.** There will be a written, closed-book midterm roughly halfway through the term. The exam will be in class.
- **Final Exam.** There will be a written, closed-book final exam at whatever time is scheduled for our class' final exam.

Due Dates

All graded assignments are due by the time and date indicated. I will not accept late assignments or make up exams. You will get zero credit for any late assignment. The only exceptions will require: a **note** from an appropriate authority and **immediate notification** of the problem when it arises. Naturally, your excuse must be acceptable. If a meteor landed on your bed and destroyed your assignment, I need a signed note from the meteor. You should also treat assigned readings as, well, assignments that are due at the beginning of each class.

Numbers

Component	
Assignments	50%
Midterm	25%
Final	25%

Although class participation is not explictly graded, I will use your class participation to determine whether your grade can be lifted in case you are right on the edge of two grades. Participation means attending classes, participating in class discussions, asking relevant questions, volunteering to provide answers to questions, and providing constructive criticism and creative suggestions that improve the course.

Disclaimer

I reserve the right to modify any of these plans as need be during the course of the class; however, I won't do anything capriciously, anything I do change won't be too drastic, and you'll be informed as far in advance as possible.

Reading List

Course Material

Required Text:

- Tom Mitchell, Machine Learning. McGraw-Hill, 1997.

Optional Text:

- Machine Learning: The Art and Science of Algorithms that Make Sense of Data.
- Richard Sutton and Andrew Barto, Reinforcement Learning: An introduction. MIT Press, 1998. (for Reinforcement Learning)

Resources

Software

- WEKA Machine learning software in JAVA that you can use for your projects
- Data Mining with Weka A MOOC Course
- ABAGAIL Machine learning software in JAVA. This is hosted on my github, so you can contribute too
- scikit-learn A popular python library for supervised and unsupervised learning algorithms
- pybrain A popular python library for artifical neural networks
- MATLAB NN Toolbox The toolbox supports supervised learning with feedforward, radial basis, and dynamic networks and unsupervised learning with self-organizing maps and competitive layers.
- Murphy's MDP Toolbox for Matlab
- MATLAB Clustering Package By Frank Dellaert
- ICA Example

Other

- UCI Machine Learning Repository An online repository of data sets that can be used for machine learning experiments.
- Stanford Large Network Dataset Dataset of large social and information networks.
- Vision Benchmark Suite Autonomous car dataset
- Other datasets