Machine Learning

CSCI 5622, Section 001

Location: Tuesday and Thursday, 9:30-10:45, ECCR 139
Instructor: Professor Michael Mozer
Office: ECOT 741
Office Hours: Tuesday and Thursday, 11:00-12:15
Phone: (303) 492-4103
E-Mail: mozer@colorado.edu
Course URL: http://www.cs.colorado.edu/~mozer/courses/5622

Course Description

The goal of *machine learning* research is to build computer systems that learn from experience and that adapt to their environments. Machine learning systems do not have to be programmed by humans to solve a problem; instead, they essentially *program themselves* based on examples of how they should behave, or based on trial-and-error experience trying to solve the problem. Machine learning systems require *learning algorithms* that specify how the system should change its behavior as a result of experience. Researchers in machine learning develop new learning algorithms, and try to understand which algorithms should be applied in which circumstances.

Machine learning is an exciting interdisciplinary field. Its historical roots are in theoretical computer science, statistics, pattern recognition, and even neuroscience. In the past 15 years, many of these approaches have converged and led to rapid theoretical advances as well as real-world applications.

This course will focus on the methods that have proven valuable and successful in practical applications. The course will also contrast the various methods, with the aim of explaining the situations in which each is most appropriate.

Course Goals

When you have completed this course, you will have an appreciation for the most popular and useful machine learning methods, and should be able to apply these methods to solve learning problems of moderate complexity. You should also be able to read current research papers in machine learning and understand the issues raised by the research.

Prerequisites

You are expected to be a competent programmer, but the language you choose to program in is up to you. C++, Java, LISP, perl, and matlab should all work fine. You should understand basic calculus, linear algebra, and probability and statistics, although we'll do

a quick review of key material in the course. If you have limited background in statistics and linear algebra, expect to put some extra time into the course.

Text

The required text is called *Machine Learning* by Tom Mitchell. Although it is the best and most thorough text in the field, it overlaps only by about 50% with the material I think is important to cover. Consequently, many of the class lectures will be on material not contained in the book. I will hand out supplementary readings for some of the topics not covered in the text. Reading assignments from the text are listed in the syllabus; other reading assignments will be given out as the class date approaches.

The optional text is called *Neural Networks for Pattern Recognition* by Chris Bishop. I have used this text in the past for the neural networks course, and students tended not to like it because of its strong mathematical flavor. However, if you have a background in statistics or want to learn how to think about machine learning from a statistical perspective, it is the book to read.

Handouts

Course handouts will be brought to class *once*. If you miss a class in which a handout is distributed, you may obtain a copy from the “Course work Handout File” located outside the Computer Science Department main office, room ECOT 717, during regular business hours. I will also try to make postscript copies of the handouts available by ftp from the class web site (see URL at the top of this handout).

Course Work

In the past, I have always given exams, even in graduate level courses, if only to force the students to review material. However, the exams were never a great experience, and took time away from more productive and interesting activities. I am going to try an experiment this year and drop the exams in favor of homework assignments.

I believe that students can best learn about machine learning by implementing machine learning algorithms and experimenting with them first hand. I could give assignments covering theoretical issues (e.g., asking you to prove theorems) or I could provide you a simulator (an implementation of a machine learning algorithm, wrapped in a graphical user interface), but to really understand the algorithms, I believe that you have to know them well enough to program them and debug them yourselves. One or more of the homework assignments will involve a competition among participants in the class. I will give everyone the same data for training a machine learning system, and we will determine whose system performs the best on test data.

I expect students to attend all classes. Attendance is important because about half of the material that I will cover is not included in the text. To emphasize the importance of atten-

dance, 10% of your grade will be based on attendance and participation in class. The other 90% of your grade will be based on homework assignments.

Homework Assignments

The homework assignments will generally involve programming to implement machine learning algorithms from scratch, and experimentation to test your algorithm on some data. Programming can be done in any language. You will be asked to summarize your work in brief (2 page) write ups.

Because the homework assignments comprise the bulk of your grade, *collaboration on the assignments in not allowed*. Each student is responsible for his or her own work. Discussion of homework assignments and programs should be limited to clarification of the assignment hand out itself, and should not involve any sharing of pseudocode or code or simulation results. Violation of this policy is grounds for a semester grade of F, in accordance with university regulations.

The tentative schedule of assignments is listed in the syllabus. All assignments will count roughly equally toward the final grade.

The tentative schedule of assignment due dates is given in the class-by-class schedule on the next page. This schedule is subject to change as the course progresses and adapts to your interests.

Lecture Date	*Topics*	*Assignments in/out*
Jan 16	Introduction I • Examples of machine learning systems • Learning paradigms • Classification versus regression	HW1 assigned Ch. 1
Jan 18	NO CLASS	
Jan 23	Introduction II	HW1 due
Jan 25	Simple learning methods • K nearest neighbor • Naive Bayes	Ch 6.2-6.3; 6.9; 8.1-8.2 HW 2 assigned
Jan 30	Overfitting • Model complexity • Bias-variance trade off	
Feb 1	Evaluation • Cross validation • Hypothesis testing	Ch. 5
Feb 6	Decision trees I	Ch. 3 HW2 due; HW3 assigned
Feb 8	Decision trees II	
Feb 13	*catch up day*	
Feb 15	Simple neural networks I • Perceptron algorithm	Ch. 4.1-4.4

Lecture Date	*Topics*	*Assignments in/out*
Feb 20	Simple neural networks II • LMS algorithm	
Feb 22	Multilayer neural networks • Back propagation	HW3 due; HW4 assigned Ch. 4.5-4.7
Feb 27	Neural networks — tricks and variations I	Ch. 4.8
Mar 1	Neural networks — tricks and variations II	Ch 6.5; 8.4; 12.3-12.4
Mar 6	Recurrent neural networks • Application: music composition • Application: time series prediction	
Mar 8	Ensemble techniques I • Mixture of experts • Bagging	reading (to be announced)
Mar 13	Ensemble techniques II • Boosting	
Mar 15	Application: data mining in the real world	HW4 due HW5 assigned
Mar 20	Unsupervised learning I • K means clustering • topographic maps • mixture models / density estimation	Ch. 6.12 reading (to be announced)
Mar 22	Unsupervised learning II • self-supervised back propagation • generative models	
Mar 27	*SPRING BREAK*	
Mar 29	*SPRING BREAK*	
Apr 3	Reinforcement learning I	Ch. 13
Apr 5	Reinforcement learning II	HW 5 due HW6 assigned
Apr 10	Application: adaptive house	
Apr 12	Genetic algorithms	Ch. 9
Apr 17	Hidden Markov models • forward-backward algorithm • application: speech recognition	reading (to be announced)
Apr 19	Bayesian belief networks	Ch 6.11 HW 6 due HW 7 assigned
Apr 24	Models of human learning and adaptation	reading (to be announced)
Apr 26	Advanced topics I • Kernel methods • Support vector machines	reading (to be announced)
May 1	Advanced topics II • Gaussian processes	HW 7 due
May 3	Advanced topics III • Computational learning theory	