CSG220: Machine Learning
Spring, 2007

Instructor: Professor Ron Williams
Email: rjw@ccs.neu.edu
Office: 334 West Village H, 617-373-8683
Office Hours: Thurs., 3:00-5:00, or by appointment

Course web page: http://www.ccs.neu.edu/home/rjw/csg220

Required Textbook: *Machine Learning*, by Tom M. Mitchell, McGraw-Hill, 1997.

Optional Supplementary Textbook: *Artificial Intelligence: A Modern Approach, 2nd Edition*, by Stuart Russell and Peter Norvig, Prentice-Hall, 2003.

Some Other Useful Books:

- *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, by Ian H. Witten & Eibe Frank, Morgan Kaufmann, 2000.
- *Pattern Classification (Second Edition)*, by Richard O. Duda, Peter E. Hart & David G. Stork, John Wiley & Sons, 2001.
- *An Introduction to Support Vector Machines and Other Kernel-Based Methods*, by Nello Christianini and John Shawe-Taylor, Cambridge University Press, 2000.
- *Reinforcement Learning: An Introduction*, by Richard S. Sutton & Andrew G. Barto, MIT Press, 1998.

Other course materials: Copies of the lecture slides, this syllabus, homework assignments, etc., are all downloadable from the course web site.

Content: Topics to be covered in this course include hypothesis spaces, concept learning, decision trees, neural networks, probabilistic methods, computational learning theory, instance-based learning, support vector machines, boosting, unsupervised learning, reinforcement learning, and genetic algorithms. One objective of this course is to expose you to a wide range of approaches to machine learning and their relationships, and to have you write and/or run programs that implement some of these approaches. You will also learn about many of the issues, both theoretical and practical, that surround this field.

Prerequisites: The only prerequisite for this course is familiarity with undergraduate-level probability theory, together with a reasonable level of general mathematical sophistication. Specific mathematical areas that we will rely on in various parts of this course are probability and statistics, linear algebra, and multidimensional calculus.

Also helpful, but not required, is familiarity with a selected subset of material covered in a basic AI course like CSG120. Furthermore, good programming skill is important, but you have the option of writing programs in any language using any machine you feel comfortable with. However, you are also welcome (indeed, encouraged) to make use of free software provided through various sources, and you will naturally be able to make fullest use of this software if you are familiar with the language it is written in. For example, some of the software that you can freely download and use for this course is written in Common Lisp (including a suite of programs provided by me that is downloadable from the course web site). Other software available from other sources is in Java, and still other software is in C.

Grading: Your overall grade for this course will be based on homework (30%), one examination (30%), given April 12, and a course project (40%).

Homework: Several homework assignments will be distributed and collected every other week throughout the first half of the semester. These will generally consist of a combination of pencil-and-paper questions and some programming and experimental studies. Homework assignments must be turned in by the due date to receive full credit. Homework turned in up to 1 week late will be penalized 20%, and no homework will be accepted beyond 1 week past its due date.

Examination: There will be a 3-hour examination given during the class meeting time on Thursday, April 12.

Course Project: You will be expected to complete a term project for this course. A short (1- or 2-page) project proposal outlining what you intend to do is due Thursday, March 8, and I will provide feedback to you at the following class meeting concerning its sufficiency and appropriateness. At the final class meeting, on April 26, you will turn in a written report and deliver a short presentation to the class on the project. The grade for the project will be based primarily on the written report, but the quality of the presentation may also be taken into account. The main purpose of the presentations, however, is to allow the class to learn about everyone else's projects.

As examples of what you may do for your project, you may:

- apply a machine learning algorithm to an interesting task;
- implement two or more approaches to the same task and compare them;
- experiment with a novel and potentially worthwhile modification of an existing machine learning algorithm;
- experiment with an approach that combines two or more existing machine learning algorithms in a novel and interesting way; or
- perform a relevant theoretical analysis.

More detail on the expected content of project reports will be provided separately.

Course Schedule:

Date	Topic	Mitchell Chapters (or Sections)
Jan. 11	Introduction	1
	Decision Trees	3
Jan. 18	Hypothesis Space Structure	2
	Artificial Neural Networks	4
Jan. 25	Artificial Neural Networks (continued)	
	Probabilistic & Bayesian Methods	6.1-6.10
Feb. 1	Probabilistic & Bayesian Methods (continued)	
Feb. 8	Instance-Based Learning	5
	Avoiding Overfitting	3.7.1, 4.6.5
	Cross-Validation	
Feb. 15	Computational Learning Theory, VC Dimension	7
Feb. 22	Support Vector Machines	
Mar. 1	Ensemble Learning	
	Project Proposals Due	
Mar. 8	Spring Break	
Mar. 15	Markov Decision Processes & Reinforcement Learning	13
Mar. 22	Bayes Nets	6.11
Mar. 29	Gauss Mixture Models	
	EM & k-Means Algorithms	6.12
Apr. 5	Hidden Markov Models	
Apr. 12	*Examination*	
Apr. 19	Genetic Algorithms	9
Apr. 26	Project Reports Due	
	Project Presentations	