15-681 Machine Learning

Fall 1994

Handout # 1: Administrivia

Instructors:

Avrim Blum (Office: Wean 4107, email: avrim@cs.cmu.edu, tel: x8-6452.)
Office hours: Tues 2-3

Tom Mitchell (Office: Wean 5309, email: mitchell@cs.cmu.edu, tel: x8-2611.)
Office hours: Wed 1-2

TA:

Jefferey Shufelt (Office: Wean 4114, email: js@cs.cmu.edu, tel: x8-3075.)
Office hours: Mon 3-4

Course Secretary:

Jean Harpley (Office: Wean 5313, email: jean@a.gp.cs.cmu.edu, tel: x8-3802)

Time: Tuesday and Thursday 10:30-11:50.

Place: Porter A19

Requirements and grading policy:

There will be 6 or 7 homeworks, a midterm, and a final. Some of the homeworks will be written assignments and some will be programming projects. We may combine two of the homeworks into one larger project.

The total value of the homeworks will be 50%, the midterm will be worth 20% and the final, 30%.

OUTLINE

Introduction: An illustrative learning task, and a few approaches to it. What is known from algorithms, theory, experiment, biology, psychology.

Version Spaces: illustration, choosing examples.

Models: PAC model, Mistake bound model. Example algorithms and proofs. General relations. Decision lists.

Decision Tree Learning: Decision trees, ID3, information gain, pruning, query algorithms.

Bias: Minimum description length principle, Occam's razor.

Continuous weightings: Weighted majority algorithm, perceptrons.

Neural Network Learning: multilayer nets, gradient descent.

Overfitting: cross validation, sampling errors, probabilistic inequalities, VC-dimension.

Bayesian Approaches: Expectation Maximization, Hidden Markov Models.

Instance-based techniques: Lazy vs. eager generalization, k nearest neighbors, case-based reasoning.

Genetic Algorithms: Different search methods for induction.

Biological learning systems: experimental results.

Explanation-Based Learning: Using prior knowledge to reduce learning time and sample complexity.

Combined inductive/analytical learning: Symbolic methods, Neural network methods.

Learning Agents: Reinforcement learning, Agent architectures for learning (e.g., Soar), Exploration, Competitive analysis, Learning finite state environments and feature invention.