Course Summary

- Introduction:
  - Basic problems and questions in machine learning.
- Linear Classifiers
  - Naïve Bayes
  - Logistic Regression
  - LMS
- Five Popular Algorithms
  - Decision trees (C4.5)
  - Neural networks (backpropagation)
  - Probabilistic networks (Naïve Bayes; Mixture models)
  - Support Vector Machines (SVMs)
  - Nearest Neighbor Method
- Theories of Learning:
  - PAC, Bayesian, Bias-Variance analysis
- Optimizing Test Set Performance:
  - Overfitting, Penalty methods, Holdout Methods, Ensembles
- Sequential Data
  - Hidden Markov models, Conditional Random Fields; Hidden Markov SVMs

Course Summary

- Goal of Learning
- Loss Functions
- Optimization Algorithms
- Learning Algorithms
- Learning Theory
- Overfitting and the Triple Tradeoff
- Controlling Overfitting
- Sequential Learning
- Statistical Evaluation

Goal of Learning

- Classifier: $\hat{y} = f(\mathbf{x})$ "Do the right thing!"
- Conditional probability estimator: $P(y|\mathbf{x})$
- Joint probability estimator: $P(\mathbf{x},y)$
  - compute conditional probability at classification time

Loss Functions

- Cost matrices and Bayesian decision theory
  - Minimize expected loss
  - Reject option
- Log Likelihood: $\sum_k -I(y=k) \log P(y=k|\mathbf{x},h)$
- 0/1 loss: need to approximate
  - squared error
  - mutual information
  - margin slack (“hinge loss”)

Optimization Algorithms

- None: direct estimation of μ, Σ, P(y), P($\mathbf{x}$ | y)
- Gradient Descent: LMS, logistic regression, neural networks, CRFs
- Greedy Construction: Decision trees
- Boosting
- None: nearest neighbor

Learning Algorithms

- LMS
- Logistic Regression
- Multivariate Gaussian and LDA
- Naïve Bayes (gaussian, discrete, kernel density estimation)
- Decision Trees
- Neural Networks (squared error and softmax)
- k-nearest neighbors
- SVMs (dot product, gaussian, and polynomial kernels)
- HMMs/CRFs/averaged perceptron

The Statistical Problem: Overfitting

- Goal: choose *h* to optimize <u>test set</u> performance
- Triple tradeoff: sample size, test set accuracy, complexity
  - For fixed sample size, there is an accuracy/complexity tradeoff
- Measures of complexity:
  - |H|, VC dimension, log P(h), ||**w**||, number of nodes in tree
- Bias/Variance analysis
  - Bias: systematic error in *h*
  - Variance: high disagreement between different *h*'s
  - test error = Bias^2 + variance + noise (square loss, log loss)
  - test error = Bias + unbiased-variance – biased-variance (0/1 loss)
- Most accurate hypothesis on training data is not usually most accurate on test data
- Most accurate hypothesis on test data may be deliberately wrong (i.e., biased)

Controlling Overfitting

- Penalty Methods
  - Pessimistic pruning of decision trees
  - Weight decay
  - Weight elimination
  - Maximum Margin
- Holdout Methods
  - Early stopping for neural networks
  - Reduce-error pruning
- Combined Methods (use CV to set penalty level)
  - Cost-complexity pruning
  - CV to choose pruning confidence, weight decay level, SVM parameters C and σ
- Ensemble Methods
  - Bagging
  - Boosting

Off-The-Shelf Criteria

Criterion	LMS	Logistic	LDA	Trees	Nets	NNbr	SVM	NB	Boosted Trees
Mixed data	no	no	no	yes	no	no	no	yes	yes
Missing values	no	no	yes	yes	no	some	no	yes	yes
Outliers	no	yes	no	yes	yes	yes	yes	disc	yes
Monotone transforms	no	no	no	yes	some	no	no	disc	yes
Scalability	yes	yes	yes	yes	yes	no	no	yes	yes
Irrelevant inputs	no	no	no	some	no	no	some	some	yes
Linear combinations	yes	yes	yes	no	yes	some	yes	yes	some
Interpretable	yes	yes	yes	yes	no	no	some	yes	no
Accurate	yes	yes	yes	no	yes	no	yes	yes	yes

What We've Skipped

- Unsupervised Learning
  - Given examples $\mathbf{X}_i$
  - Find: $P(\mathbf{X})$
  - Clustering
  - Dimensionality Reduction

What We Skipped (2)

- Reinforcement Learning: Agent interacting with an environment
  - At each time step t
    - Agent perceives current state *s* of environment
    - Agent choose action to perform according to a <u>policy</u>: $a = \pi(s)$
    - Action is executed, environment moves to new state s' and returns reward r
  - Goal: Find π to maximizes long term sum of rewards

What We Skipped (3): Semi-Supervised Learning

- Learning from a mixture of supervised and unsupervised data
- In many applications, unlabeled data is very cheap
  - BodyMedia
  - Task Tracer
  - Natural Language Processing
  - Computer Vision
- How can we use this?

Research Frontier

- More complex data objects
  - sequences, images, networks, relational databases
- More complex runtime tasks
  - planning, scheduling, diagnosis, configuration
- Learning in changing environments
- Learning online
- Combining supervised and unsupervised learning
- Multi-agent reinforcement learning
- Cost-sensitive learning; imbalanced classes
- Learning with prior knowledge