Review and Exam

The Story So Far…

- Inductive learning / Supervised learning
- Decision trees
- Nearest neighbor (instance-based learning)
- Perceptron
- Linear Regression
- Logistic regression
- Support vector machines (and kernels)
- Neural Networks

ML in a Nutshell

- Tens of thousands of machine learning algorithms
- Hundreds new every year
- Every machine learning algorithm has three components:
  - **Representation**
  - **Evaluation**
  - **Optimization**

Representation

- Decision trees
- Instances
- Linear function (hyperplane)
- Neural networks
- Support vector machines
- (Model ensembles)
- (Sets of rules / Logic programs)
- (Graphical models (Bayes/Markov nets))
- Etc.

Evaluation

- Accuracy
- Precision and recall
- Squared error
- Likelihood
- Posterior probability
- Cost / Utility
- Margin
- Entropy
- KL divergence
- Etc.

Optimization

- Combinatorial optimization
  - E.g.: Greedy search
- Convex optimization
  - E.g.: Gradient descent
- Constrained optimization
  - E.g.: Linear programming

Exam

- Closed book, one page of notes
- 1 hour and 30 minutes to complete
- Conceptual questions, minor calculation
- Covers everything we've discussed in class so far

Inductive Learning

- Definitions
  - Data distribution, loss function, expected loss, training error
  - Training data, test data, validation data (or development data)
  - Overfitting, underfitting
  - Hyperparameters, tuning
  - Inductive bias
- How to define a machine learning problem.

Decision trees

- Representation
  - What functions can a decision tree represent?
  - How does it represent them?
  - How large does a decision tree need to be in order to represent different kinds of functions?
- Learning
  - How do we learn a decision tree?
  - What hyperparameters? How to choose them?
- MIGHT SHOW UP ON TEST:
  - Simple Information Gain, Gain Ratio calculations
  - Inductive Bias in Decision Tree (abstract questions)
- NOT NEEDED:
  - Pruning (Pre-pruning, post pruning)

k-Nearest Neighbor

- Representation
  - What can k-nearest neighbor represent?
  - Distance functions
  - Effect of changing k
- Curse of dimensionality – how distances change in high dimensions (qualitative), why kNN can still work in practice?
- MIGHT SHOW UP ON TEST
  - Simple nearest neighbor calculations for prediction
  - Advantages and disadvantages of kNN
- NOT NEEDED: k-means, kd-trees, weighted k-means

Perceptron

- Representation
  - What functions can a linear model represent?
  - Linear separability
- Learning
  - Perceptron update
  - Convergence (qualitative)
- MIGHT SHOW UP ON TEST:
  - Effect of example order, number of epochs
  - Determine weights for perceptron
- NOT NEEDED:
  - Averaged/Voted perceptron,
  - Analytic geometry, convergence rate or proof, margins
  - Multi-class perceptron

Linear Regression

- Representation
  - Which problem and function linear regression represents?
- Learning
  - Cost function, how to optimize it with gradients?
- MIGHT SHOW UP ON TEST
  - Analytical Solution
  - Relations between Mean Squared Error, Maximum Likelihood Estimation, Maximum A Posterior (MAP)

Logistic Regression

- Representation
  - Which problem and function logistic regression represents?
- Learning
  - Conditional likelihood (basic definition)
  - Logistic function, logistic loss
  - Properties of the loss function
  - Overfitting
- MIGHT SHOW UP ON TEST:
  - For grad students: computing gradients of a loss function, writing gradient descent update rules
- NOT NEEDED: convergence rate, learning rate effect, second-order methods

Linear models

- Regularized learning objective
- Convex and smooth loss functions
- Weight regularization
- Properties of regularizers (p-norms, L1, L2)
- Gradient descent, stochastic gradient descent, mini-batching pros and cons
- Hyper-parameters (learning rate, choice of regularizer, weight of regularizer) and their effect
- Validation, Cross-validation
- NOT NEEDED: Convergence rate of gradient descent, Bayes justification for L1 and L2

Support Vector Machines (SVM)

- Representation
  - Linear SVM, geometric intuition, margins
- Learning
  - The primal and dual optimization problems
  - Support vectors
  - With the kernel trick for nonlinear SVMs, how they affect the prediction?
  - Soft margin SVMs (the primal and dual problems)
  - What hyper-parameters? How they affect the model?
- MIGHT SHOW UP IN TEST
  - Compute typical kernel functions (polynomial, RBF)
  - Identify the predicted class of a given SVM model on an example
  - Multi-class classification
- NOT NEEDED
  - Coordinate ascent, Lagrangian functions, derivation of the dual forms, invent a new kernel

Neural Networks

- Representation
  - Which problems can neural networks represent? With which extensions?
- Learning
  - Gradient descent
  - Activation functions
  - Gradient vanishing
- MIGHT SHOW UP IN TEST
  - Can a neural dataset fit a given dataset?
  - Conceptual questions about activation functions, gradient vanishing
- NOT NEEDED
  - Back-propagation, convergence

Sample questions: Simulate Classifier Behavior

Given a dataset:

- Draw the decision tree that would be generated from it, using information gain to select the splits.
- Show the results of a perceptron update for the first two instances, assuming initial weights of 0, all variables are represented as 0/1 (not -1/+1).
- Draw the decision boundary for an SVM, find the support vectors.
- Determine which model can perfectly fit to the data (zero error on the data).
- Specify parameters for a liner classifier that separates the data.
- Computer cross-validation error for a method

Sample questions: Compare and Contrast Methods

For each problem, which of the classifiers that we have discussed (decision tree, nearest neighbor, logistic regression, perceptron, neural net, SVM) would probably be the best and why?

Problem 1: Recognizing handwritten letters (A-Z) from 32x32 pixel images, given 1 million training examples.

Problem 2: Predicting heart attack probability based on 50 risk factors, each of which has an independent effect on the output. 1000 training examples.

Problem 3: Predicting grade (A,B,C) on an essay test using a bag-of-words model. Only 50 training examples available.

Other Questions

- Predict the label for x in a kernelized SVM given b and α.
- Implication of SVMs with kernels
- How many nodes are requested to represent parity with a decision tree?
- Describe how to properly compare the accuracy of two machine learning methods, each with one parameter to tune, on a dataset of only 1000 examples.
- Write the loss function for a method, derive the update rule based on gradient descent.
- How do you identify overfitting? How do you reduce overfitting in specific models?
- Match the learning curves for different models on training, validation and test data.