

CMPS 142 Midterm Topics (Winter 2010)

The midterm will be in class Thursday Feb. 25. The exam will be closed-book, although students may have one 3x5 card of handwritten notes (both sides). The exam will have space for your answers, so no blue books will be needed.

Here are the topics we have covered in CMPS 142 so far this quarter:

1. What is Machine Learning
2. Classification and Concept learning, batch learning, examples
3. Regression, supervised learning, unsupervised learning (briefly)
4. Hypothesis classes, version space, inductive bias
5. VC (Vapnik-Chervonenkis) dimension and shattering points by hypothesis class
6. Noise and causes (label errors, attribute errors, features or hypothesis class may not fit phenomena exactly)
7. Overfitting and the difference between generalization error and error on the training set; using held-out test sets and cross validation to estimate generalization
8. Nearest Neighbors prediction and kNN algorithm
9. Perceptron algorithm
10. Logistic Regression
11. Artificial Neural Networks and the backpropagation algorithm
12. Decision Trees: Greedy construction of trees, information gain criterion, applying a split criterion (impurity function, e.g. information gain) to select tests at nodes, over-fitting and pruning), converting decision trees into rules
13. Basic probability (sample space, events, random variables, independence, conditional probability, Bayes rule)
14. Estimating probabilities (e.g. the bias of a coin flip): maximum likelihood, priors, maximum a'posteriori, mean a'posteriori, and Laplacian estimates of probabilities
15. Hypotheses as models generating the data, the maximum likelihood hypothesis, the maximum a'posteriori hypothesis, and the mean a'posteriori predictions
16. Asymmetric losses, Bayes' risk, and Bayes' optimal predictions
17. Learning Gaussian distributions – biased and unbiased estimators of the variance
18. Naïve Bayes' algorithm
19. Boosting algorithms and AdaBoost
20. Support vector machines: margin maximization, kernel functions, softmargin (allowing margin errors)