CIS 798 (Topics in Computer Science)
Topics in Intelligent Systems and Machine Learning
Fall, 1999

Midterm Exam (Closed-Book)
Sample Questions

1. **True/False (10 questions, 3 points each)**
   a) **Unsupervised learning means learning with no training examples.**
   False. Unsupervised learning may occur using training examples. The difference between unsupervised learning from examples and supervised learning from examples is the presence, in supervised learning, of a class label (e.g., from the teacher).
   b) **The final version space is guaranteed to contain the target concept, if it belongs to *H* and all examples are consistent.**
   True. See Chapter 2: this is a property of the definition of a version space, not of any particular algorithm.
   c) **The VC dimension of *left-bounded intervals on the real line* is 2.**
   False. The VC dimension of *left-bounded intervals on the real line* is 1. See Lecture 3 for a 2-point set that cannot be shattered by this hypothesis class.
   d) **The VC dimension of *half-spaces in the plane* is 2.**
   False. The VC dimension of *half-spaces in the plane* is 3. The XOR problem in 2D is an example of a 4-point set that cannot be shattered by this hypothesis class.
   e) $\boldsymbol{VC(H) \leq \lg(|H|)}$
   True. The VC dimension of a hypothesis class cannot exceed the logarithm (base 2, for concept learning, i.e., binary classification) of its size. In other words, the number of hypotheses needed to shatter a data set is at least exponential in the number of examples.
   f) **In decision tree learning, perfect purity may not always be achieved at a leaf, even if all attributes are used and the examples are consistent.**
   True. There may not be enough information in the attributes left to choose from to achieve perfect purity (try to find an example for yourself).
   g) **A 3-layer feedforward ANN can represent any function to arbitrarily small error.**
   True. This result was proven by Cybenko.
   h) **Backpropagation of error in feedforward ANNs is not guaranteed to converge to a globally optimal solution.**
   True. Simple backpropagation implements gradient search and only achieves locally optimal solutions.
   i) **If uniform priors are assumed over data, MAP estimation reduces to ML estimation.**
   False. If uniform priors are assumed over hypotheses, MAP estimation reduces to ML estimation.
   j) **Assuming constant time to compute I(X; Y), the Chow-Liu algorithm for learning tree-structured Bayesian networks takes O(n^2) time in the number of variables.**
   False. Assuming constant time to compute I(X; Y), the Chow-Liu algorithm for learning tree-structured Bayesian networks takes O($n^2 \lg n$) time in the number of variables. This is dominated by the heap query (*Delete-Max*) step in Kruskal's maximum weight spanning tree (MWST) algorithm.

2. **Multiple Choice (5 questions, 6 points each)**
   a) **Which of the following is not a global optimization method for general-case machine learning?**
   A) Analytical integration over hypotheses
   B) Gradient descent
   C) Markov chain Monte Carlo (MCMC)
   D) All can be global optimization methods
   E) None are necessarily global optimization methods

Answer: (*B*). Gradient descent is, in general, a local optimization technique.

b) **Which of the following are learning methodologies?**

A) I. Supervised
B) II. Unsupervised
C) III. Reinforcement
D) I and II but not III
E) I, II, and III

Answer: (*E*). Supervised learning, unsupervised learning, and reinforcement learning are all learning methodologies.

c) **Which of the following are measures of expressiveness?**

A) I. VC-dimension of a hypothesis space
B) II. Size of a hypothesis space
C) III. Mistake bounds for a learning algorithm
D) I and II but not III
E) I, II, and III

Answer: (*D*). $VC(H)$ and $|H|$ measure the expressiveness of H relative to an implicit instance space (without specifying a particular concept); mistake bounds measure the complexity of learning a concept with a particular algorithm, relative to an implicit hypothesis space.

d) **What type of anti-overfitting method is pre-pruning?**

A) Prevention
B) Avoidance
C) Identification
D) Recovery
E) None of the above

Answer: (*B*). Pre-pruning avoids overfitting. It halts the training process when imminent overfitting is detected according to some criterion (e.g., decreasing performance on a validation set). It does not prevent overfitting from occurring, nor does it allow overfitting to occur and then recover from it.

e) **What does the $P(D \mid h)$ term in the BIC/MDL criterion correspond to?**

A) Residual error
B) Redundancy error
C) Regression error
D) Data error
E) Model error

Answer: (*A*). $P(D \mid h)$ denotes the (marginal) likelihood of the hypothesis *h,* i.e., the likelihood that it generated the data. We can think of this as a quantity that can be coded by recording *exceptions in D to the predictions of h*. These exceptions represent the residual error of h when it is used as a predictor.

3. Definitions (6 questions, 5 points each)

[Note: *The definitions on the exam will consist of 6 of the 25 below. Entries in* ***bold*** *are examples from the review session.***]**

a) **Learning** – a computer program is said to learn from experience *E* with respect to some class of tasks *T* and performance measure *P*, if its performance at tasks in *T*, as measured by *P*, improves with experience *E*.

b) Performance element / performance system

c) Attribute

d) Instance space

e) Hypothesis space

f) **Version space** – the version space, denoted $VS_{H,D}$, with respect to hypothesis space *H* and training examples *D*, is the subset of hypotheses from *H* consistent with the (labeled) training examples in *D*, *i.e.*, all $h \in H$ such that for all examples *x* in *D*, $h(x) = c(x)$.

g) True error

h) Training error

i) **(Efficiently) PAC learnable** – for a concept class *C* defined over a set of instances *X* of length *n*, and a learner *L* using hypothesis space *H*, *C* is PAC learnable by *L* using *H* if for all

$c \in C$, distributions $\mathcal{D}$ over X, ε such that 0 < ε < ½, and δ such that 0 < ε < ½, learner L will, with probability at least (1 - δ), output a hypothesis $h \in H$ such that $error_{\mathcal{D}}(h) \leq \varepsilon$ (in time that is polynomial in 1/ε, 1/δ, n, and $size(c)$ for efficiently PAC-learnable concepts).

j) Inductive bias
k) Occam's Razor
l) Decision tree – a classifer represented as a tree of arity ≥ 2, whose internal vertices represent tests on attributes, whose edges represent attribute values, and whose leaves represent classifications
m) Overfitting
n) Perceptron
o) Winnow
p) Multi-layer perceptron
q) Recurrent ANN
r) Bayes Optimal Classifier – a classifer that integrates the prediction v of each hypothesis $h \in H$, $P(v \mid h)$, in proportion to its posterior probability given data D, $P(h \mid D)$.
s) Gibbs sampling
t) Naive Bayes assumption
u) Bayesian network
v) Confidence interval
w) Type I statistical error – a case where the null hypothesis H_0 is rejected when it is true (*i.e.*, a false negative). (The probability of such an error, called the significance level for simple H_0, is denoted α.)
x) Type II statistical error
y) k-fold cross validation

4. Problem Solving (1 question, 30 points)

Work through the *PlayTennis* example of Section 3.4.2, Mitchell as a typical problem. Figure 3.4 illustrates the kind of work you should show.

Topics for this problem include: version spaces and the candidate elimination algorithm, decision tree induction using *ID3*, training perceptrons and Winnows, and learning simple (Naïve) Bayesian classifiers. The problem will be to work through a single concrete example using a very small sample data set and show what hypothesis is learned.

5. Proofs (2 questions, 15 points each)

a) Show that returning the class probability $\alpha = p/(p + n)$ at a leaf in decision tree induction minimizes the sum of squared errors, $E = p(1-\alpha)^2 + n\alpha^2$.

Solution: First, let α be the value returned at the leaf. Calculate the sum of squared errors (SSE), and its derivative:

$$\frac{dE}{d\alpha} = 2\alpha n - 2p(1-\alpha) = 2\alpha(p+n) - 2p$$

The zeros of the first derivative correspond to relative minima or maxima of E. We know that the zero is a relative min because the second derivative is positive:

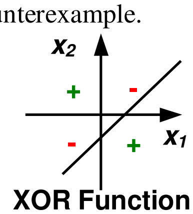
$$\frac{d^2E}{d\alpha^2} = 2(p+n) > 0$$

Therefore, E is minimized when we return $\alpha = p/(p + n)$.

This was an extra credit problem from Homework 1 (Exercise 18.7(b) in Russell and Norvig).

b) Give a counterexample to show that VC(half-planes) < 4.

Solution: XOR in the plane is a counterexample.



XOR Function