VICTORIA UNIVERSITY OF WELLINGTON

Te Whare Wānanga o te Ūpoko o te Ika a Māui

EXAMINATIONS — 2004

END-YEAR

COMP 421

Machine Learning

Time Allowed: 3 Hours

Instructions: There are 5 questions: each question is worth 45 marks.

Answer FOUR (4) questions (180 marks).
If you answer more questions, only your best 4 will be taken.

Use the number of marks as a guide to the amount of time you should spend on each sub-question.

Non-electronic Foreign-English language dictionaries are permitted.

Question 1. Various topics [45 marks]

(a) [10 marks] One approach to finding the best solution to a problem is *exhaustive search*, in which we simply generate and test every possible solution, keeping track of the best one seen so far. Discuss reasons why this is impractical in many situations.

(b) [10 marks] Gradient ascent with line-searching is one method for locating the global optimum of a function in a high-dimensional space. Discuss the issues associated with using this approach, and in what circumstances we might expect it to work well.

(c) [10 marks] Describe TWO ways of handling 'deletions' in Profile hidden Markov models, giving the relative advantages of each.

(d) [15 marks] Answer ONE of the following.

EITHER:
Function approximators (such as neural networks) can be used to represent the Q-values used in reinforcement learning.

1. Outline how this is achieved and how the system can learn from experience.
2. What are the potential benefits, and problems, with this approach?

OR:
The following equation is known as the Bellman equation for value function V of the optimal policy:

$$V^\star_{\mathbf{s}} = \max_a \sum_{s'} P_{\mathbf{s}'|\mathbf{s},a} \ [R_{\mathbf{s}'} \ + \ \gamma V^\star_{\mathbf{s}'}]$$

$\mathbf{s}$ represents the current state, a is an action taken in the current state, and $\mathbf{s}'$ is a state at the next time-step. R is the expected immediate reward. The '$\star$' indicates that this is the *optimal* value function.

1. Give the algorithm known as 'Value Iteration'.
2. Write the equivalent of the Bellman equation but for $Q^\star_{s,a}$ instead of $V^\star_{\mathbf{s}}$.

Question 2. Max likelihood and Bayesian methods [45 marks]

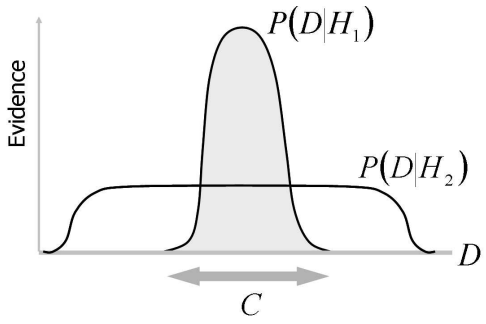
(a) [10 marks] The EM algorithm is a general procedure for finding maximum likelihood parameters θ. It is applicable to models where we have a distribution $P(h|\theta, \mathbf{D})$ over some 'hidden' states h. Explain the two steps in the general algorithm, and why they need to be iterated. If you cannot describe the general form, you might want to refer to either mixtures of Gaussians or HMMs as a specific example.

(b) [10 marks] The following equation gives the log likelihood for a binary (two-class) classification model. The n training examples are indexed by μ and consist of input vectors together with their known classifications Y, which are 0 or 1. The system outputs a probability y that the given input vector should be classified as a 1.

$$\log L \;=\; \sum_{\mu=1}^{n} \; Y^{\mu} \log y^{\mu} \;-\; (1 - Y^{\mu}) \log(1 - y^{\mu})$$

Explain why this is the correct equation for the log likelihood in this situation.

(c) [10 marks] 'Under-fitting' and 'over-fitting' are examples of the problems associated with complexity control in learning systems. The principle known as Ockham's razor loosely suggests that we should use the simplest model capable of accounting for the data. Explain how it is that the Bayesian posterior probability of a model can be said to *automatically* embody Ockham's principle, provided we integrate out the unknown parameters in the model. You *may* find the following diagram useful to refer to in your answer:



(d) [15 marks] Choose ONE of the following.

EITHER:
What are the Cox-Jaynes axioms, and what is their significance for inference in the presence of uncertainty?

OR:
Discuss the following - 'In Bayesian machine learning, hypotheses are compared via their posterior probabilities. But in a fully Bayesian analysis there is no need to choose between competing hypotheses'.

Question 3. Sampling [45 marks]

Suppose we have a probability distribution $p(\mathbf{x})$ defined over the unit hypercube in n dimensions, and we wish to draw samples from $\mathbf{x}$, weighted by their probabilities.

(a) [10 marks] What is the 'curse of dimensionality', and why is it a problem for using Rejection Sampling as an algorithm the above task? You may assume that a simple uniform proposal distribution will be used.

(b) [10 marks] In the Metropolis algorithm we propose a new solution $\mathbf{x}'$ chosen from a proposal distribution $q(\mathbf{x}'|\mathbf{x})$ where $\mathbf{x}$ is the current solution. We then make a jump to this new one with an acceptance probability :

$$a(\mathbf{x}'|\mathbf{x}) = \min\left(1,\ \frac{p(\mathbf{x}')q(\mathbf{x}|\mathbf{x}')}{p(\mathbf{x})q(\mathbf{x}'|\mathbf{x})}\right)$$

Argue that this algorithm will eventually generate samples from the distribution $p(\mathbf{x})$, by showing that the above rule satisfies an appropriate detailed balance equation.

(c) [15 marks] While the curse of dimensionality is much less of an issue for MCMC algorithms, they can still suffer from problems with

- *multi-modal* probability distributions, and
- correlated random walks.

Explain why these present problems for MCMC. (Hint: it *may* help to refer specifically to either Gibbs Sampling or Metropolis-Hastings).

(d) [10 marks] MCMC algorithms only generate samples from the right distribution 'at equilibrium'. Briefly describe the 'Simulated Annealing' algorithm, as a method for speeding up the approach to equilibrium in the Metropolis-Hastings algorithm.

Question 4. Belief nets [45 marks]

(a) [10 marks] Probability propagation is used for making inferences about the values of the unobserved variables in a belief network, given the states of any observed nodes. Instead of just making 'passive' observations, we can actively intervene by forcing one of the variables to take a specific value. However, when running probability propagation, such an intervention should be accompanied by the temporary disconnection of that node from its parents. Explain why this is so.

(b) [5 marks] Assuming that the underlying graph is a polytree (*i.e.* it is singly connected), inferring the probability distribution over values for *all* the unobserved nodes in a belief net takes only twice as long as for any *one* node. Briefly explain why this is so. (Note: you are *not* required to explain the details of the actual messages propagated through the graph).

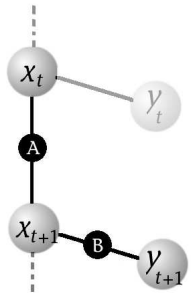
(c) [5 marks] 'Belief nets simplify the full joint by making conditional independence assumptions'. Briefly explain, by giving an example based on the graph corresponding to the following factorization:

$$p(a, b, c) \;=\; p(a)\;\; p(b|a)\;\; p(c|a)$$

(d) [15 marks] In order to run probability propagation on a belief network that is *not* a polytree, it must be converted into one or several such trees. Briefly describe TWO methods for achieving this.

(e) [10 marks] 'Ancestral simulation' is used for estimating the (conditional) probability of an unobserved node in a belief network, given the observed states of some other nodes. Under what circumstances is ancestral simulation a feasible approach to adopt?

Question 5. HMMs and decision networks [45 marks]



The figure on the left shows a generic section of the factor graph of a standard hidden Markov model (HMM), corresponding to one time-step. The variable nodes are the hidden state $\mathbf{x}$ at times t and $t+1$, and observations $\mathbf{y}$ at times t and $t+1$.

(a) [8 marks] Copy the above figure onto a full page, and indicate on it the generic probability distribution represented by each of the messages to and from factors A and B, that would be produced in running probability propagation on this model. Note:

- You *don't* need to give those to and from $\mathbf{y}_t$, and you *don't* need to describe exactly how those messages are generated.
- Your answers should be in the form $P(a, b|c)$, etc, and there should be 8 messages in all.

(b) [4 marks] What is the Viterbi algorithm used for?

HMMs with added 'action' and 'utility' nodes are known as *dynamic decision nets*, and can be used for decision-making in POMDPs (partially observable Markov decision processes). We didn't meet these specifically in lectures, but we did discuss action selection in decision nets, of which these are an example.

(c) [6 marks] Briefly indicate the meanings of the following terms:

1. 'stationary',
2. 'partially observable', and
3. 'Markov'

(Note: it may help to refer to the graph given above as an example.)

(d) [7 marks] By adding the two new types of node (representing actions and utilities), draw a portion of the extended HMM graph as a belief net (*i.e. without factor nodes*) corresponding to one time-step, from t to $t+1$.

(e) [10 marks] Describe an algorithm for deciding between possible actions at time t in the above model.

(f) [10 marks] Discuss such 'dynamic decision nets' in terms of the philosopher Daniel Dennett's notion of 'Popperian creatures'.

****