Machine Learning Final

This TWO-SIDED exam is open book. You may bring in your homework, class notes and text- books to help you. You will have 1 hour and 15 minutes. Write all answers in your blue book. Please make sure YOUR NAME is on each of your blue books. Square brackets [] denote the points for a question.

1. **Reinforcement Learning(RL)**
   Consider the modules approach to RL.
   (a) [5] *BRIEFLY* describe one advantage if the approach over conventional RL
   Defeats the curse of dimensionality by working with small state spaces
   (b) [5] *BRIEFLY* describe one disadvantage if the approach over conventional RL Small state spaces may lead to inconsistent policies
   (c) [5] Could there be a broblem with too many modules? *BRIEFLY* discuss Action selection may be an average instead of a maximum
   (d) [10] Describe one function that chooses a **policy** from a set of modules and defend your choice. Pick action from the module with most expected reward. Avoids averaging disadvantages

2. **Deep Learning**

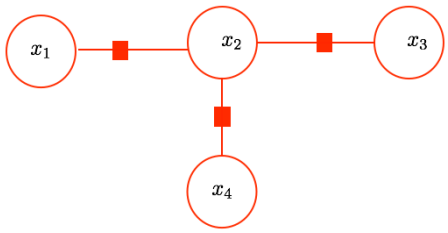
   (a) [15] The early backpropagation networks used a sigmoid activation function $g(u)$ where
   $$g(u) = \frac{1}{1+e^{-u}}$$
   but this had the "vanishing gradient" problem: $g'(u) \to 0$ as $|u|$ becomes large. Show this problem by using the derivative explicitly.
   $g(`u) = \frac{e^{-u}}{(1+e^{-u})^2}$; u large implies $\frac{0}{1}$: u large negative implies $\frac{1}{\infty}$

   (b) [10] Changing the activation to a 'semi-linear' function allowed very deep layers in networks to use backpropagation.
   Using the $H = \frac{1}{2}||\mathbf{x}^K - \mathbf{d}||^2 + \sum_{k=0}^{k=K-1}(\lambda)^{k+1}(-\mathbf{x}^{k+1} + g(W^k\mathbf{x}^k))$, the sigmoid gradient is given by:
   $$\frac{\partial H}{\partial w_{ij}^k} = \lambda_i^{k+1} x_j^k g'(\mathbf{w}_i^k \mathbf{x}^k) \tag{1}$$
   Show how equation (2) is modified when using the new activation function.

   $$\frac{\partial H}{\partial w_{ij}^k} = \lambda_i^{k+1}(x_j^k)^2 \text{ if } x_j > 0 \text{ else } 0$$

3. **Graphical Models**

Consider the following 4-node graph:



where

$$p(x_2) = \sum_{x_1} \sum_{x_3} \sum_{x_4} f_a(x_1, x_2) f_b(x_2, x_3) f_c(x_2, x_4)$$

(a) [10] Show the **extra messages** necessary for $p(x_2)$ when the following subgraph is added to the original.

$$p(x_2) = \sum_{x_5} \sum_{x_1} \sum_{x_3} \sum_{x_4} f_d(x_5, x_1) f_a(x_1, x_2) f_b(x_2, x_3) f_c(x_2, x_4)$$

(b) [15] What is the formula for $p(x_2)$ when x_1 is **observed**?

$$p(x_2) = \sum_{x_3} \sum_{x_4} f_a(\hat{x}_1, x_2) f_b(x_2, x_3) f_c(x_2, x_4)$$

4. **Markov Random Fields**

A **transparent** image is defined as $y(y_1, y_2)$ where y_1 and y_2 are $\in \{+1, -1\}$ How would you set up a Markov Random Field process using an image (x_1, x_2) to de-noise transparent images? Show all notation that you come up with.

[10] What would the energy functions look look like?

[15] What would the network look like?

There are lots of ways to do this. On is to have two independent networks like the one in the Bishop textbook. Another is to couple those two networks with a link that has a energy savings for having them both on.