# CS535c Fall 2004: Homework 2 clarifications

Kevin Murphy

24 Sep 04

## 1 Q1

Hint: build the minimal Imap in the usual way (ch 3, p97) by finding, for node $X_i$, the minimal subset $U$ of $\{X_1, \ldots, X_{i-1}\}$ s.t., $X_i \perp_G \{X_1, \ldots, X_{i-1}\} \setminus U|U$ where the conditional independencies are with respect to the original graph (the one with the A node). Use the Bayes ball rules to answer these conditional independence queries. You shade $U$ (which will never contain $A$, since $A$ does not exist in the new graph), and see if you can get a ball from $X_i$ to the other (non-parental) ancestors.

## 2 Q2

Hint: See ch 4, p154.

## 3 Q3

Suppose when you send out a 1 bit, it gets transmitted as a +1V signal; and if you send out a 0 bit, it gets transmitted as a -1V signal. The receiver gets this voltage, plus Gaussian noise (zero mean, variance 1); call the received signal for the i'th bit $Y_i$. Suppose you receive $Y_i = -0.1$. So it's a little bit more likely to be a 0 bit than a 1 bit, but it's hard to tell, because it's near the decision boundary for the 2 classes.

You want to calculate the probability that the sent bit was 0 or 1 (ignoring parity checks for now). So we calculate the conditional likelihood of observing $y_i$:

$$\begin{aligned} p(y_i = -0.1|X_i = 0) &= N(-0.1; -1; 1) = 0.2661 \\ p(y_i = -0.1|X_i = 1) &= N(-0.1; +1; 1) = 0.2179 \end{aligned}$$

(You can compute these numbers using the Matlab statistics toolbox function `normpdf`.) We define the local evidence at node $i$ to be the vector $F_i = (0.2661, 0.2179)$. Once we have the $F_i$'s, we no longer need the actual observations, $Y_i$. This is what I mean by saying we have "compiled" the observations into the local potentials.

Note that the conditional likelihoods do not need to sum to one; in fact, $p(y_i|X_i)$ may even exceed 1.0 (since a probability *density* function at a point may be bigger than 1 as long as it integrates to 1 overall : see ch 2, p19. e.g., `normpdf(-1,-1,0.1) = 3.984`). However, for convenience, we will normalize the local evidences:

$$\begin{aligned} f_i(1) &= p(y_i|X_i = 1)/z_i = 0.2661/0.4840 = 0.5498 \\ f_i(2) &= p(y_i|X_i = 0)/z_i = 0.2179/0.4840 = 0.4502 \\ z_i &= p(y_i|X_i = 1) + p(y_i|X_i = 0) = 0.2661 + 0.2179 = 0.4840 \end{aligned}$$

An easy way to do this in Matlab is

```
>> normalize([normpdf(-0.1, -1, 1), normpdf(-0.1, 1, 1)])
ans =
    0.5498    0.4502
```

where I define

```
function [a2, z] = normalize(a)
z = sum(a(:));
a2 = a ./ z;
```

In the homework, I stack all the local evidence vectors up (columwise) into a matrix F, we get

```
F = [0.5 0.5 0.5 0 0;
     0.5 0.5 0.5 1 1];
```

F(:,i) is Matlab notation for the i'th column of matrix F, i.e., the local evidence for node i. This is a potential/ factor that only affects $X_i$.

In this case, the local evidence about $X_1, X_2, X_3$ is completely ambiguous (equivalent to receiving $Y_1 = Y_2 = Y_3 = 0.0$ in the above example: exactly in between -1 and +1); however, the local evidence about the parity bits, $X_4$ and $X_5$, is completely unambiguous: we know that $X_4 = X_5 = 0$ (state 2 = false). This gives us a hard constraint on the possible values of $X_1$ and $X_2$.

When you draw the Markov network, you can ignore the fact that $g$ is an xor (determinsistic) function; just assume generic potentials (see ch 5, p186). But you will need to exploit properties of $g$ when computing the numbers, of course.