



Statistical Models in Computational Biology

Jack Kuipers David Dreifuss Xiang Ge Luo Rudolf Schill

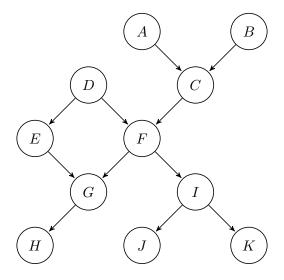
Due 13th of April 2023

Please submit your project with the filename Lastname(s)_Project7.pdf.

Problem 17: Junction Tree Algorithm

(2 points)

Consider the Bayesian network on the variables $U = \{A, \dots, K\}$ given by the graph:



(a) Build the *Junction Tree* of the network.

- (1 point)
- (b) Write the joint probability P(U) in terms of the cluster and separator potentials. (1 point)

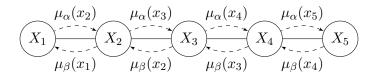


Figure 1: Message passing on the undirected chain.

Problem 18: Benefit of storing messages

(3 points)

Consider message passing on the undirected chain in Figure 1, in which $\mu_{\alpha}(x_n)$ and $\mu_{\beta}(x_n)$ represent the forward and backward messages for $n \in \{2, 3, 4, 5\}$, as seen in the lecture.

- (a) Write the formula for recursively computing the forward and backward messages. (1 point)
- (b) What is the complexity of computing the marginal probability $P(X_4 = 1)$ using message passing? (1 point)

(c) If you store all the messages, what is the complexity of computing all marginal probability distributions X_1 , X_2 , X_3 , X_4 , and X_5 ? How about the general case, with a chain of length N where each node can assume K values? Assume that storing and multiplying is free, so that only summation counts for the complexity. (1 point)

Problem 19(data analysis): Message passing on a chain

(5 points)

We will now use R to perform the message passing shown in Figure 1. Suppose that we have the following (conditional) probability distributions:

(1)
$$p(X_1 = 1) = \frac{1}{3}$$
, $p(X_2 = 1 \mid X_1) = \begin{cases} \frac{4}{5} & \text{if } X_1 = 0 \\ \frac{2}{3} & \text{if } X_1 = 1 \end{cases}$, $p(X_3 = 1 \mid X_2) = \begin{cases} \frac{5}{7} & \text{if } X_2 = 0 \\ \frac{1}{3} & \text{if } X_2 = 1 \end{cases}$

(2)
$$p(X_4 = 1 \mid X_3) = \begin{cases} 3/5 & \text{if } X_3 = 0 \\ 2/5 & \text{if } X_3 = 1 \end{cases}, \quad p(X_5 = 1 \mid X_4) = \begin{cases} 1/2 & \text{if } X_4 = 0 \\ 7/9 & \text{if } X_4 = 1 \end{cases}.$$

Note that these equations fully determine each (conditional) probability distribution, since $X_i \in \{0,1\}$ for $i \in \{1,\ldots,5\}$.

(a) Store clique potentials in an R object

(2 points)

Since all $X_i \in \{0,1\}$, each clique potential $\psi_{n,n+1}$ can be stored as a 2×2 matrix

(3)
$$\Psi_{n,n+1} := \begin{pmatrix} \psi_{n,n+1}(0,0) & \psi_{n,n+1}(0,1) \\ \psi_{n,n+1}(1,0) & \psi_{n,n+1}(1,1) \end{pmatrix}.$$

Each $\psi_{n,n+1}(\cdot,\cdot)$ can be computed using the factorisation from conditional probabilities in equations (1) and (2). Compute and store these clique potentials in a three-dimensional array, such that the third dimension holds the clique matrices in equation (3) for $n = 1, \ldots, 4$.

$$Hint: array(dim = c(2, 2, 4), dimnames = list(c("0", "1"), c("0", "1"), c("Psi12", "Psi23", "Psi34", "Psi45")))$$

(b) Computing forward messages

(1 point)

In the problem *Message passing*, you have written the formula for recursively computing the forward messages $\mu_{\alpha}(x_n)$ for $n \in \{2,3,4,5\}$. How can these be computed for $x_n \in \{0,1\}$ as the matrix product of the vector $(\mu_{\alpha}(X_{n-1}=0),\mu_{\alpha}(X_{n-1}=1))$ multiplied by $\Psi_{n-1,n}$, with $\Psi_{n-1,n}$ as defined in equation (3)? Initialise $\mu_{\alpha}(x_1)=1$ for $x_1 \in \{0,1\}$, and compute the remaining forward messages.

(c) Computing backward messages

(1 point)

Similarly, how can the backward messages $\mu_{\beta}(x_n)$ for $n \in \{4,3,2,1\}$ can be computed for $x_n \in \{0,1\}$ as the matrix product of $\Psi_{n,n+1}$ multiplied by the vector $(\mu_{\beta}(X_{n+1}=0), \mu_{\beta}(X_{n+1}=1))^{\top}$? Initialise $\mu_{\beta}(x_5) = 1$ for $x_5 \in \{0,1\}$, and compute the remaining backward messages.

(d) Compute the marginal probability distribution for each node

(1 point)

Multiply (element-wise) forward and backward messages at each position to obtain the marginal probability distributions for X_1 , X_2 , X_3 , X_4 , and X_5 . What is the normalising constant Z?