

# Introduction to AI - assignment 6

Oded Yechiel

8/1/19

## Contents

<b>1</b>	<b>1</b>
1.1 Construct a Bayes network that is consistent with the above set of independence statements . . .	1
1.2 Is the answer above unique? . . . . .	2
<b>2</b>	<b>2</b>
2.1 Is this network a poly-tree? . . . . .	2
2.2 Is this network (directed-path) singly connected? . . . . .	2
2.3 Determine the truth of the following independence statements (using d-separation) . . . . .	2
2.4 Find $P(E=\text{true}, A=\text{True} \mid B=\text{true})$ . . . . .	3
<b>3</b>	<b>3</b>
3.1 Can this be done without hidden units? . . . . .	3
3.2 Show a network using threshold elements that performs the required computation . . . . .	3

## Exercise 1

$I(\{A, B\}, \{C, D\} \mid \{\})$   
 $I(\{A, B\}, \{C, D, G, H\} \mid \{\})$   
 $I(\{E, F\}, \{C, D, G, H\} \mid \{\})$   
 $I(\{G\}, \{C\} \mid \{D\})$

not  $I(\{A\}, \{B\} \mid \{\})$   
not  $I(\{C\}, \{G\} \mid \{\})$   
not  $I(\{E\}, \{F\} \mid \{\})$   
not  $I(\{C\}, \{G\} \mid \{D, H\})$

### 1.1 Construct a Bayes network that is consistent with the above set of independence statements

Since A and B are closed and are independent from all other nodes, they can be separated from the rest of the tree and just connected between themselves (since they are dependent).

Same goes for nodes E and F.

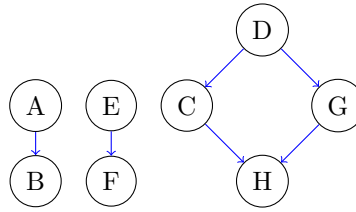


Figure 1: Bayesian Network

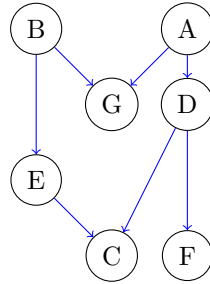


Figure 2: Network tree

Since C and G are dependent unless D is in evidence this suggests that D is a parent of both C and G. In addition, since C and G are dependent in the presence of H this suggests that C and G are parents of H.

Fig. 1 proposes a Bayes Network tree that supports all of the statements in the requirements.

## 1.2 Is the answer above unique?

**No.** As mentioned above H has to be the child of C and G, and D has to be their parents. However, A can be the parent of B, or B can be the parent of A. Same goes for E and F. Thus, there are  $2^2$  combinations, which add up to **4 possible BN that will hold the statements.**

## Exercise 2

### 2.1 Is this network a poly-tree?

### 2.2 Is this network (directed-path) singly connected?

### 2.3 Determine the truth of the following independence statements (using d-separation)

1.  $I(\{D\}, \{E\} \mid \{ \})$  - False
2.  $I(\{D\}, \{E\} \mid \{ A \})$  - True
3.  $I(\{D\}, \{E\} \mid \{A, C\})$  - False
4.  $I(\{B\}, \{F\} \mid \{ G \})$  - True
5.  $I(\{B\}, \{F\} \mid \{ D \})$  - True

6.  $I(\{B\}, \{F\} \mid \{A, C\}) = \text{False}$

## 2.4 Find $P(E=\text{true}, A=\text{True} \mid B=\text{true})$

$P(A=\text{true}) = 0.2$ ,  $P(B=\text{true}) = 0.3$   
 $P(G=\text{true} \mid A=\text{true}, B=\text{true}) = 0.9$ ,  $P(G=\text{true} \mid A=\text{false}, B=\text{true}) = 0.2$   
 $P(G=\text{true} \mid A=\text{true}, B=\text{false}) = 0.5$ ,  $P(G=\text{true} \mid A=\text{false}, B=\text{false}) = 0.1$   
 $P(E=\text{true} \mid B=\text{true}) = 0.8$ ,  $P(E=\text{true} \mid B=\text{false}) = 0.1$   
 $P(C=\text{true} \mid D=\text{true}, E=\text{true}) = 0$

Since G and C are not in evidence and B is in evidence, we can safely say that A and E are statistically independent. Moreover, A and B are also independent since G and C are not in evidence. Hence

$$P(E, A \mid B) = P(E \mid B) * P(A \mid B) = P(E \mid B) * P(A) = 0.8 * 0.2 = 0.16$$

## Exercise 3

### 3.1 Can this be done without hidden units?

**No.** It is impossible with no hidden layers to predict a non-linear function such as modulu. However, for this simple example, a single hidden layer will suffice as shown in the following subsection.

### 3.2 Show a network using threshold elements that performs the required computation

We will use a neural network with 1 hidden layers that is composed of 6 neurons. This means for the input layer (6 neurons), the hidden layer will have 36 weights (6 x 6 matrix - M1) and 6 bias weights that are summed and enter a *tanh* activation function.

The output, a 6x1 vector X1 is then bitwise multiplied by 6 weights (one for each neuron) summed and are added with the bias b2. The output is passed through a sigmoid activation function and the result is rounded to decide if the result is '0' or '1'.

The complete code in Keras that generates the dataset and labels, trains, and provides the weights can be found, along with this report, in the git repository at <https://github.com/odedyec/IntroToAI/blob/master/assignment6/q7.py>

$$M1 = \begin{bmatrix} -2.6035163 & 1.0730004 & 0.73517716 & 0.7892835 & 2.0436563 & -3.0340612 \\ -2.8252845 & 1.1691521 & 1.3764652 & 1.5624126 & -3.209769 & -2.571492 \\ -2.5988533 & 1.1376183 & 0.9316116 & 0.563356 & 2.0437527 & -3.031866 \\ -2.5722306 & 1.2495581 & 0.7956654 & 0.7881598 & 2.0363252 & -3.0355027 \\ -2.603346 & 1.1103705 & 0.60637933 & 0.61684054 & 2.0353322 & -3.025518 \\ -2.5890596 & 1.0401373 & 0.26387808 & 1.0831383 & 2.0149522 & -3.0376844 \end{bmatrix} \quad (1)$$

$$b1 = [8.979317, -5.988329, 0.26482144, -0.250453, -1.1340998, 6.665368]^T \quad (2)$$

$$M2 = \begin{bmatrix} 6.996323 \\ 4.3360047 \\ -0.807298 \\ -0.6610357 \\ -1.4745305 \\ -10.0931015 \end{bmatrix} \quad (3)$$

$$b2 = -2.8664327 \quad (4)$$

$$X1 = \tanh(M1 \times X + b1) \quad (5)$$

$$\hat{y} = \text{round}(\text{sigmoid}(M2 \times X1) + b2) \quad (6)$$