

# CS 486/686 Assignment 3 (175 marks in total)

Instructor: Alice Gao

Due Date: 6:00 pm on Thursday, July 11, 2019

## Instructions

- Submit the assignment in the Dropbox labeled Assignment 3 Submissions in the Assignment 3 folder on LEARN. No late assignment will be accepted. This assignment is to be done individually.
- For any programming question, you may use the language of your choice. We highly recommend using Python. If you don't know Python, it may be worthwhile to learn it for this course.
- Lead TAs:
  - Aravind Balakrishnan (a4balakr@uwaterloo.ca)
  - Kaiwen Wu (k77wu@uwaterloo.ca)
  - Ehsan Ganjidoost (eganjido@uwaterloo.ca)

The TAs' office hours will be posted on the course website.

- Submit two files with the following naming conventions. **We will deduct 10 marks if you do not follow these conventions.**
  - **writeup.pdf**
    - \* Include your name, email address and student ID in the writeup file.
    - \* If you hand-write your solutions, make sure your handwriting is legible and take good quality pictures. You may get a mark of 0 if we cannot read your handwriting.
  - **code.zip**
    - \* Include your program, a script to run your program, and a README.txt file with instructions to run the script. You may get a mark of 0 if we cannot run your program.

# Learning goals

## Constructing Bayesian Networks

- Given the relevant probabilities, determine whether two random variables are independent.
- Given the relevant probabilities, determine whether two random variables are conditionally independent given a third random variable.
- Given a scenario with independent assumptions and a given order of the variables, construct a Bayesian network by adding the variables to the network based on the given order.

## Inference in Bayesian Networks

- Define factors. Manipulate factors using the operations restrict, sum out, multiply and normalize.
- Trace the execution of and implement the variable elimination algorithm for calculating a prior or a posterior probability given a Bayesian network.

# 1 The Variable Elimination Algorithm (114 marks)

Your dog Aria has been howling for the last three hours. In the past, Aria may howl when she is genuinely sick, but she also howls sometimes when there is a full moon or when the neighbour's dog is howling. You are trying to figure out if you should take her to the vet.

You will construct a Bayesian network to model the reasons for Aria to howl. Then, you will use the variable elimination algorithm to compute the probability that Aria is genuinely sick and help you decide what to do. You have the following information available to you.

- Aria often howls if she is genuinely sick. However, Aria is a healthy dog and is only sick 5% of the time. If Aria is really sick, then she probably has not eaten much of her dinner and has food left in her bowl. In the past, when Aria is sick then she does not eat her dinner 60% of the time. However, when Aria is healthy she still does not eat her dinner 10% of the time.
- Aria often howls if there is a full moon or your neighbour's dog howls. Your neighbour's dog sometimes howls at the full moon and sometimes howls when your neighbour is away. However, your neighbour's dog is never affected by Aria's howls. Since you live on Earth, there is a full moon once out of every twenty-eight days. You have observed that your neighbour travels three days out of every ten days.
- If there is no full moon and your neighbour is at home then your neighbour's dog never howls. If there is no full moon and your neighbour is away then your neighbour's dog howls half of the time. If there is a full moon and your neighbour is at home then your neighbour's dog howls 40% of the time. Finally, if there is a full moon and your neighbour is away then your neighbour's dog howls 80% of the time.
- If all three triggers are there (Aria sick, full moon, and your neighbour's dog howling), Aria howls 99% of the time. If none of the three triggers are there, Aria will not howl.
- If Aria is sick, she is more likely to howl. If being sick is the only trigger present, Aria will howl half of the time. If Aria is sick and your neighbour's dog is howling then Aria will howl 75% of the time. If Aria is sick and there is a full moon, then Aria will howl 90% of the time.
- If Aria is not sick then she is less likely to howl. The full moon and your neighbour's dog howling will cause Aria to howl 65% of the time. If there is only a full moon then Aria will howl 40% of the time. If there is no full moon, but your neighbour's dog is howling, then Aria will howl 20% of the time.

1. Construct a Bayesian network based on the above information about Aria.

Tip: You should be able to directly take the probabilities given in the text above and insert them into the corresponding conditional probability tables. There is no need to perform any computation to obtain the conditional probabilities.

Your Bayesian network should contain six nodes corresponding to the following binary random variables.

- $AB$ : There is food left in Aria's food bowl.
- $AH$ : Aria howls.
- $AS$ : Aria is sick.
- $M$ : There is a full moon.
- $NA$ : Your neighbour is away.
- $NH$ : Your neighbour's dog howls.

**What to submit:**

- Submit a description of your Bayesian network using the list and table format described below.
- Submit a graphical representation of your Bayesian network. You can draw

**The list and table format of a Bayesian network:** To make marking easier, please use the following format to describe your Bayesian network.

- List the random variables in alphabetical order.
- For each random variable, list its parents in alphabetical order.
- For each random variable, state its conditional probability distribution in a form of a table.

For example, we can describe the Holmes Bayesian network in the following format.

- A  
parents: B, E

conditional probability distribution:

$A$	$B$	$E$	Prob
1	0	0	0.01
1	0	1	0.2
1	1	0	0.95
1	1	1	0.96

- B  
parents: none

conditional probability distribution:

$B$	Prob
1	0.0001

- E

parents: none

conditional probability distribution:

<i>E</i>	Prob
1	0.0003

- G

parents: A

conditional probability distribution:

<i>G</i>	<i>A</i>	Prob
1	0	0.04
1	1	0.4

- R

parents: E

conditional probability distribution:

<i>R</i>	<i>E</i>	Prob
1	0	0.0002
1	1	0.9

- W

parents: A

conditional probability distribution:

<i>W</i>	<i>A</i>	Prob
1	0	0.4
1	1	0.8

### Marking Scheme: (36 marks)

- (2 marks) for the parent nodes of each random variable.
- (4 marks) for the conditional probability distribution for each random variable.

## 2. Implement the variable elimination algorithm.

You may assume that all the random variables are Boolean. You may use the language of your choice, but we will give you tips on the implementation if you use Python.

Please include detailed instructions for the TA to run your program.

Factors are essentially multi-dimensional arrays. If you are using Python, you can consider using pandas or numpy. For numpy, use numpy multi-dimensional arrays as your main data structure. If you are unfamiliar with numpy, go through the following tutorial: <https://docs.scipy.org/doc/numpy-1.15.1/user/quickstart.html>.

Test each function individually using examples from the lecture slides. If you wait till the end to test your entire program, it will be much harder to debug.

Please implement the following functions.

- (a) **restricted\_factor = restrict(factor, variable, value)**: This function restricts the given variable in the given factor to the given value. [Tip: consider using the splicing operations or **take** to implement this function.]

- (b) **product\_factor = multiply(factor1, factor2)**: This function performs point-wise multiplication of the given two factors. [Tip: take advantage of the numpy broadcasting rules to multiply multi-dimensional arrays of different shapes.]
- (c) **result\_factor = sumout(factor, variable)**: This function sums out a variable in a given factor. [Tip: use the sum operation to implement this function.]
- (d) **normalized\_factor = normalize(factor)**: This function normalizes a factor by dividing each entry by the sum of all the entries. This function is useful when we need to convert the factor to a probability distribution (i.e. the sum of the values must be 1.).
- (e) **result\_factor = inference(factor\_list, query\_variables, ordered\_hidden\_var\_list, evidence\_vars)**: This function computes  $Pr(\text{query\_variables}|\text{evidence\_vars})$  by the variable elimination algorithm. The `evidence_vars` parameter is a dictionary where the keys are the variable names and the values are 1/0. This function should do the following.
  - i. Restrict the factors in `factor_list` according to the evidence in `evidence_vars`.
  - ii. Based on the order of the hidden variables in `ordered_hidden_var_list`, eliminate each hidden variable from the product of the factors in `factor_list`.
  - iii. If there are multiple factors left, then multiply them to produce a single factor.
  - iv. If the resulting factor should denote a probability distribution (i.e. the probabilities should sum up to 1), then normalize the resulting factor.

**Marking Scheme:** No marks are awarded for this part. Marks will be awarded for executing the algorithm to compute probabilities in the following questions.

3. The next few questions will ask you to compute a few probabilities using the variable elimination algorithm. For each question, you will need to include a printout of the factors computed at each step of the algorithm. See the specification below for the format of your printout.

You have the option to answer the following questions by executing the variable elimination algorithm by hand rather than implementing it. If you execute the variable elimination algorithm by hand, you will get at most 2/3 of the marks for the correctness of the steps and the final factor. See the marking schemes for more details.

**Factor printout format:**

We will use 1 to represent true and 0 to represent false. Each factor will be represented by a table as shown below. The first column of the table is optional.

The format for printing the result of a **restrict** operation: Suppose that we are restricting a factor to  $W = 1$ .

Restricted to  $W = 1$

	W	A	Prob
0	1	1	0.8
1	0	1	0.2
2	1	0	0.4
3	0	0	0.6

Result:

	A	Prob
0	1	0.8
2	0	0.4

-----

The format for printing the result of a **multiply** operation:

Multiplying:

	E	A	B	Prob
0	1	1	1	0.000288
1	1	0	1	0.000012
2	1	1	0	0.000060
3	1	0	0	0.000240
4	0	1	1	0.949715
5	0	0	1	0.049985
6	0	1	0	0.009997
7	0	0	0	0.989703

	E	Prob
0	1	1.0
2	0	1.0

Result:

	E	A	B	Prob
0	1	1	1	0.000288
1	1	0	1	0.000012
2	1	1	0	0.000060
3	1	0	0	0.000240
4	0	1	1	0.949715
5	0	0	1	0.049985
6	0	1	0	0.009997
7	0	0	0	0.989703

-----

The format for printing the result of a **sum out** operation:

Sumout E from:

	E	A	B	Prob
0	1	1	1	0.000288
1	1	0	1	0.000012
2	1	1	0	0.000060
3	1	0	0	0.000240
4	0	1	1	0.949715
5	0	0	1	0.049985
6	0	1	0	0.009997
7	0	0	0	0.989703

Result:

	A	B	Prob
0	1	1	0.950003
1	0	1	0.049997
2	1	0	0.010057
3	0	0	0.989943

-----

The format for printing the result of a **normalize** operation:

Normalize:

	B	Prob
0	1	0.000030
1	0	0.019055

Result:

	B	Prob
0	1	0.001597
1	0	0.998403

-----

- (a) Compute the prior probability that Aria howls by executing the variable elimination algorithm. Use the lexicographical order when eliminating the hidden variables.

In your writeup.pdf, provide the following.

- The probability you used to answer the question, and
- a printout of the factors computed at each step of the variable elimination algorithm.

**Marking Scheme:** 27 marks

- (2 marks) Your writeup includes the correct expression of the probability used to answer the question.



- (5 marks) The TA can run your program to produce the output printout in your writeup.pdf in under 5 minutes.
- (20 marks) The printout of output has the correct steps and the final factor has the correct probability distribution (up to 3 decimal places).  
(up to 13 marks) You executed the algorithm by hand and the steps and the final factor are correct.

(b) Aria is howling. You look out the window and see that the moon is full. Compute the probability that Aria is sick. Use the lexicographical order when eliminating the hidden variables.

In your writeup.pdf, provide the following.

- The probability you used to answer the question, and
- a printout of the factors computed at each step of the variable elimination algorithm.

**Marking Scheme:** 27 marks

- (2 marks) Your writeup includes the correct expression of the probability used to answer the question.
- (5 marks) The TA can run your program to produce the output printout in your writeup.pdf in under 5 minutes.
- (20 marks) The printout of output has the correct steps and the final factor has the correct probability distribution (up to 3 decimal places).  
(up to 13 marks) You executed the algorithm by hand and the steps and the final factor are correct.

(c) (Aria is howling and the moon is still full.) You walk to the kitchen and see that Aria has not eaten and her food bowl is full. Given this new information, compute the probability that Aria is sick. Use the lexicographical order when eliminating the hidden variables.

In your writeup.pdf, provide the following.

- The probability you used to answer the question, and
- a printout of the factors computed at each step of the variable elimination algorithm.

**Marking Scheme:** 12 marks

- (2 marks) Your writeup includes the correct expression of the probability used to answer the question.
- (10 marks) The printout of output has the correct steps and the final factor has the correct probability distribution (up to 3 decimal places).  
(up to 6 marks) You executed the algorithm by hand and the steps and the final factor are correct.

- (d) (Aria is howling, the moon is full and Aria has not eaten her dinner.) You decide to call your neighbour to see if they are home or not. The phone rings and rings so you conclude that your neighbour is away. Given this information, compute the probability that Aria is sick. Use the lexicographical order when eliminating the hidden variables.

In your writeup.pdf, provide the following.

- The probability you used to answer the question, and
- a printout of the factors computed at each step of the variable elimination algorithm.

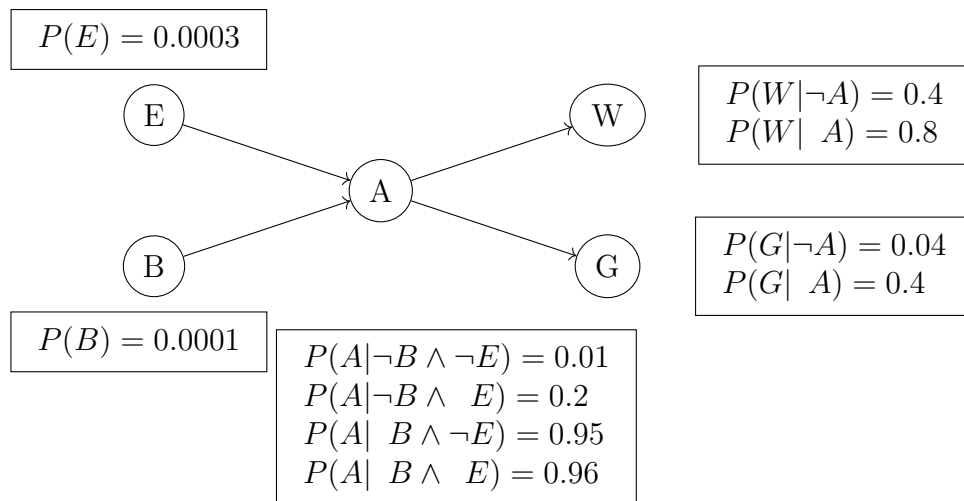
**Marking Scheme: 12 marks**

- (2 marks) Your writeup includes the correct expression of the probability used to answer the question.
- (10 marks) The printout of output has the correct steps and the final factor has the correct probability distribution (up to 3 decimal places).  
(up to 6 marks) You executed the algorithm by hand and the steps and the final factor are correct.

## 2 Constructing a Bayesian Network (61 marks)

Consider the Holmes scenario discussed in class. Let's drop the random variable Radio from the story and keep the other five random variables: Burglary, Earthquake, Alarm, Watson, and Gibbon.

In class, we saw a Bayesian network for the Holmes scenario given below. We can construct the Bayesian network below by adding the variables to the network using the following order: **B, E, A, W, G**. You can assume that the independence assumptions required by the Bayesian network below are exactly the same as the independent assumptions required by the story.



In this question, we will construct another Bayesian network using a different order of the five random variables: **W, G, A, B, E**. The following steps guide you through the process of constructing this Bayesian network.

1.  $W$  is the first random variable in the order. Add  $W$  as the first node in the Bayesian network.

**What to submit:**

- Draw the partial Bayesian network including all the conditional probability distributions.

**Marking Scheme: 2 marks**

- (2 marks) correct conditional probability distribution for  $W$ .

2. Next, we will add  $G$  to the Bayesian network. We need to determine whether  $W$  should be  $G$ 's parent node or not. Answer the following question.

(A) Is  $G$  independent of  $W$ ? If your answer is Yes, then  $G$  should have no parents. If your answer is No, then  $W$  should be  $G$ 's parent.

Add  $G$  to the Bayesian network based on your answer above.

**What to submit:**

- Answer the question and justify your answer.
- Draw the resulting partial Bayesian network including all the conditional probability distributions.

**Marking Scheme: 7 marks**

- (1 mark) correct answer per question.
- (2 marks) correct justification per question.
- (2 marks) correct partial Bayesian network.
- (2 marks) correct conditional probability distribution for  $G$ .

3. Next, we will add  $A$  to the Bayesian network. We need to choose nodes from the set of  $\{ W, G \}$  to be  $A$ 's parent nodes such that  $A$  is independent of the rest of the nodes given its parent nodes. Answer the following questions.

(A) Is  $A$  independent of  $W$  and  $G$ ? If your answer is Yes, then  $A$  should have no parents. If your answer is No, proceed to the next question.

(B) Is  $A$  independent of  $G$  given  $W$ ? If your answer is Yes, then  $W$  should be the only parent of  $A$ . If your answer is No, proceed to the next question.

(C) Is  $A$  independent of  $W$  given  $G$ ? If your answer is Yes, then  $G$  should be the only parent of  $A$ . If your answer is No, then  $W$  and  $G$  should both be parent nodes of  $A$ .

Add  $A$  to the Bayesian network based on your answers above.

**What to submit:**

- Answer each question and justify each answer.
- Draw the resulting partial Bayesian network.

**Marking Scheme: 13 marks**

- (1 mark) correct answer per question.

- (2 marks) correct justification per question.
- (2 marks) correct partial Bayesian network.
- (2 marks) correct conditional probability distribution for  $A$ .

4. Next, we need to add  $B$  to the Bayesian network. Answer the following questions. You only need to justify your answer for the question where your answer is Yes.

- (A) Is  $B$  independent of  $W$ ,  $G$ , and  $A$ ? If your answer is Yes,  $B$  should have no parents. If your answer is No, proceed to the next question.
- (B) Is  $B$  independent of  $G$  and  $A$  given  $W$ ? If your answer is Yes,  $W$  should be the only parent of  $B$ . If your answer is No, proceed to the next question.
- (C) Is  $B$  independent of  $W$  and  $A$  given  $G$ ? If your answer is Yes,  $G$  should be the only parent of  $B$ . If your answer is No, proceed to the next question.
- (D) Is  $B$  independent of  $W$  and  $G$  given  $A$ ? If your answer is Yes,  $A$  should be the only parent of  $B$ . If your answer is No, proceed to the next question.
- (E) Is  $B$  independent of  $G$  given  $A$  and  $W$ ? If your answer is Yes,  $A$  and  $W$  should be the only parent nodes of  $B$ . If your answer is No, proceed to the next question.
- (F) Is  $B$  independent of  $W$  given  $A$  and  $G$ ? If your answer is Yes,  $A$  and  $G$  should be the parent nodes of  $B$ . If your answer is No, proceed to the next question.
- (G) Is  $B$  independent of  $A$  given  $W$  and  $G$ ? If your answer is Yes,  $W$  and  $G$  should be the parent nodes of  $B$ . If your answer is No, all of  $W$ ,  $G$  and  $A$  should be the parent nodes of  $B$ .

Add  $B$  to the Bayesian network based on your answers above.

**What to submit:**

- Answer each question and justify each answer.
- Draw the resulting partial Bayesian network.

**Marking Scheme:** 23 marks

- (1 mark) correct answer per question.
- (2 marks) correct justification per question.
- (2 marks) correct partial Bayesian network.
- (2 marks) correct conditional probability distribution for  $B$ .

5. Finally, we will add  $E$  to the Bayesian network. To save you some work, you only need to answer the following two questions.

- (A) Is  $E$  independent of  $W$ ,  $G$  and  $B$  given  $A$ ? If your answer is Yes,  $A$  should be the only parent of  $E$ . If your answer is No, proceed to the next question.
- (B) Is  $E$  independent of  $W$  and  $G$  given  $A$  and  $B$ ? If your answer is Yes,  $A$  and  $B$  should be the parent nodes of  $E$ . If your answer is No, then all of  $A$ ,  $B$ ,  $W$  and  $G$  should be the parent nodes of  $E$  in the Bayesian network.

Add  $E$  to the Bayesian network based on your answers above.

**What to submit:**

- Answer each question and justify each answer.
- Draw the resulting partial Bayesian network.

**Marking Scheme: 8 marks**

- (1 mark) correct answer per question.
- (2 marks) correct justification per question.
- (2 marks) correct partial Bayesian network.
- (2 marks) correct conditional probability distribution for  $E$ .

Next, please answer a few discussion questions, which ask you to compare the Bayesian network constructed (denoted network B) with the one discussed in class (denoted network A).

- (A) What is the minimum number of probabilities required for network A?

What is the minimum number of probabilities required for network B?

**Marking Scheme: 2 marks**

- 1 mark per question

- (B) Does the network A require independence assumptions that are not required by network B? If yes, list up to two independence assumptions.

Does the network B require independence assumptions that are not required by network A? If yes, list up to two independence assumptions.

**Marking Scheme: 4 marks**

- 2 marks per question

- (C) Out of the two networks, which Bayesian network would you prefer? Justify your answer.

**Marking Scheme: 2 marks**