Lecture 12
Bayesian Networks

Alice Gao

June 10, 2021

Contents

Learning Goals

By the end of the lecture, you should be able to

- Given a Bayesian network, determine whether an (conditional) independence relationship holds using d-separation.
- Given a joint probability distribution and an order of the variables, construct a Bayesian network that correctly represents the independent relationships among the variables in the distribution.

1 Testing independence using d-separation

Up to now, we haven't had the tools to test whether an independence relationship holds. Let me present a powerful concept called d-separation, which we can use to determine whether an unconditional or conditional independence relationship holds or not.

See below for the definition of d-separation. There are two variables X and Y, and a set of observed variables E.

Definition (D-Separation)**.** A set of variables E **d-separates** variables X and Y if E blocks every un-directed path between X and Y in the network.

To determine whether X and Y are independent given the observed variables E, we can verify whether E d-separates X and Y. If d-separation holds, then the independence relationship holds as well.

Let me clarify a few things regarding the d-separation definition.

First, to verify the definition, we need to consider every un-directed path between X and Y. The word "un-directed" means that we do not care about the direction of the arrows on the path. As long as a series of nodes and edges connect X and Y, we will consider it a path.

Second, there may be multiple paths between X and Y. We need to consider every path and verify that E blocks every path between X and Y.

Third, on each path, there could be multiple nodes between X and Y. The path is blocked if at least one node blocks the path. As soon as we find one node blocking the path, we can move on to a different path. In the worst case, we need to check every node and discover that none of the nodes blocks the path.

Given this definition, our task boils down to the following: pick a path between X and Y and pick a node on the path, determine whether the node blocks the path or not.

This leads to our next question: What does it mean to "block a path"? Let me explain this in three scenarios. Interestingly, these three scenarios correspond to the three key structures that I discussed previously.

In each of the three scenarios, we will look at one path between X and Y and consider one node N on the path. The three scenarios differ by the direction of the two arrows on both sides of N.

1.1 Blocked Path — 3 Scenarios

Scenario one:

The two arrows around N point in the same direction, forming a chain around N. I drew the arrows pointing to the right, but it's fine if they point to the left. In this scenario, if N is observed, then N blocks the path between X and Y.

This rule is similar to the first key structure that is a chain. If we observe whether Alarm is going off, then Burglary and Watson become independent. You can think of observing N as

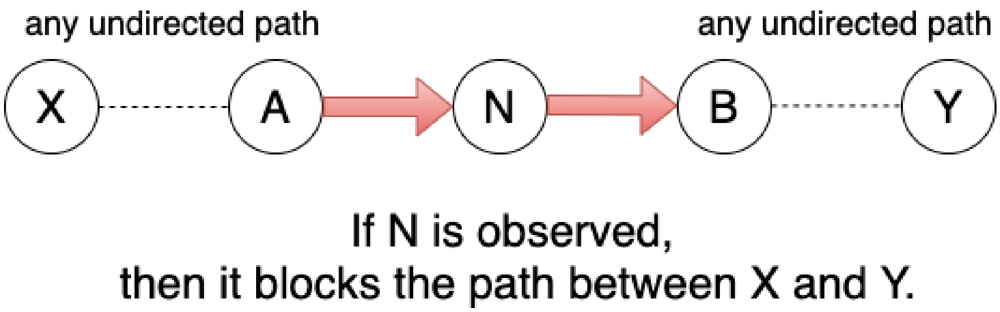


Figure 1: Scenario one

cutting the chain at N.

Scenario two:

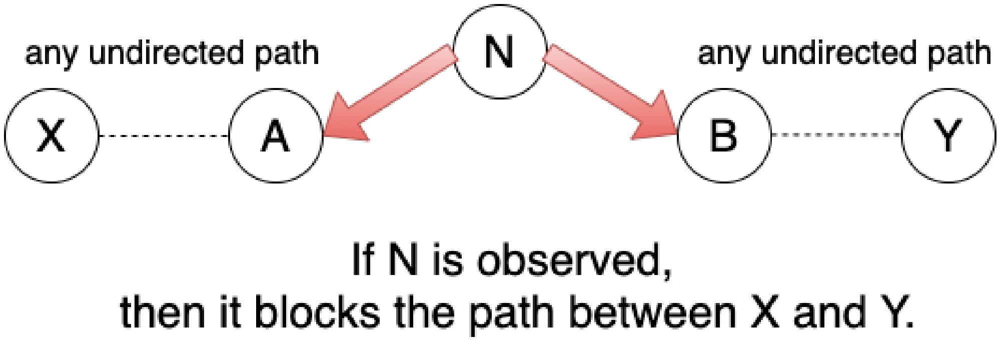


Figure 2: Scenario two

The two arrows around N point away from N, to the two children A and B. If the arrows depict causal relationships, you can think of A and B as unreliable sensors of N. If N is observed, then N blocks the path between X and Y.

This rule is similar to the second key structure. If we observe Alarm, then Watson and Gibbon become independent.

Scenario three:

The two arrows around N point toward N. A and B are both parents of N. If the arrows depict causal relationships, then A and B jointly cause N to happen. The descendants of N are also important in this scenario. The rule says that: If we do not observe N and do not observe any of N's descendants, then the path is blocked.

This rule is similar to the third key structure. If Alarm is not observed, then Burglary and Earthquake are independent. If Alarm is observed, Burglary and Earthquake become dependent.

Note that this rule is the opposite of the first two rules. The first two rules say that N blocks the path if N is observed. This third rule says that N and its descendants block the path if they are not observed.

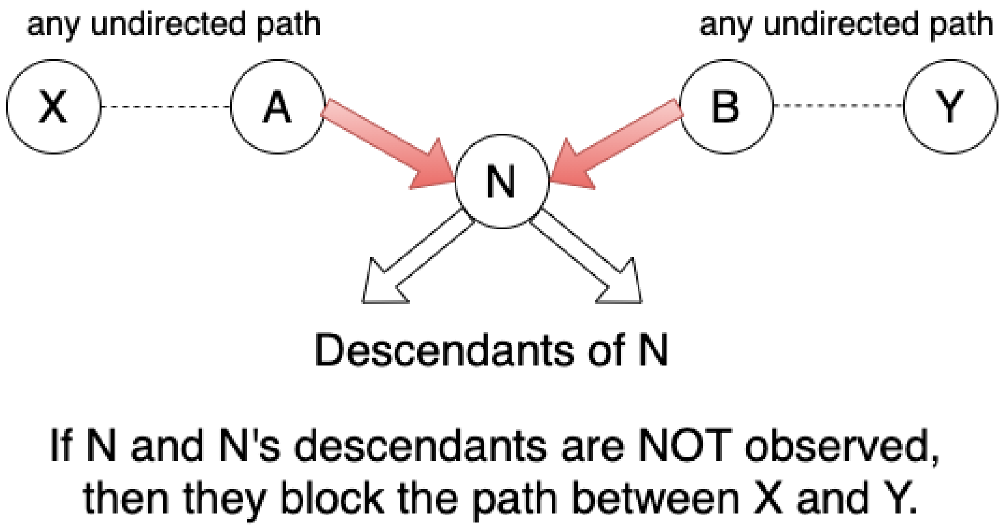
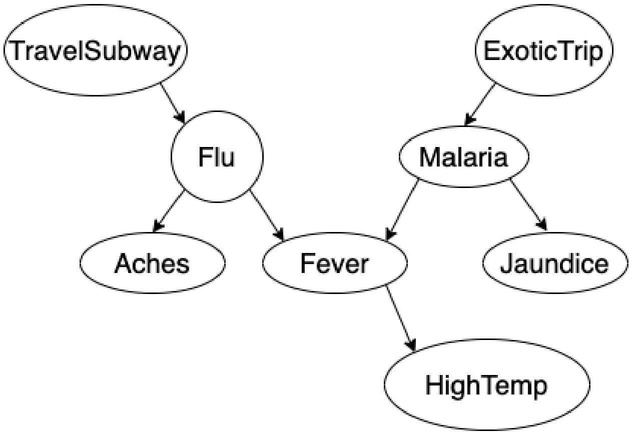

any undirected path
any undirected path
X
A
B
Y
N
Descendants of N
If N and N's descendants are NOT observed, then they block the path between X and Y.

Figure 3: Scenario three

1.2 Examples of Applying D-Separation

To understand d-separation, let's apply it in a few examples. Try to solve each one yourself before reading the discussion. Each example will use the same Bayesian network describing the effects of travelling on the subway or taking an exotic trip. A diagram is provided below:



Problem: **(1a)** Are TravelSubway and HighTemp independent?

Solution: Not independent.

There is only one path to check: TravelSubway — Flu — Fever — HighTemp. Applying rule 1, Flu does not block the path. Applying rule 1 again, Fever does not block the path either. Thus we have an unblocked path between TravelSubway and HighTemp, so the two variables are not independent.

Problem: **(1b)** Are TravelSubway and HighTemp conditionally independent given Flu?

Solution: Independent.

We check the same path as in problem (1a): TravelSubway — Flu — Fever — HighTemp. This time, rule 1 says Flu does block the path, so we can stop checking. (Rule 1 still says Fever does not block the path.) All paths are blocked, so the two variables are conditionally independent given Flu.

Problem: **(1c)** Are TravelSubway and HighTemp conditionally independent given Aches?

Solution: Not independent.

The reasoning is intentionally left as an exercise. Again, we need to check the same path as in problems (1a) and (1b), but you will see Aches does not affect the nodes along the path.

Problem: **(2a)** Are Aches and HighTemp independent?

Solution: Not independent.

There is only one path to check: Aches — Flu — Fever — HighTemp. Applying rule 2, Flu does not block the path. Applying rule 1, Fever does not block the path either. Thus we have an unblocked path between Aches and HighTemp, so the two variables are not independent.

Problem: **(2b)** Are Aches and HighTemp conditionally independent given Flu?

Solution: Independent.

We check the same path as in problem (2a): Aches — Flu — Fever — HighTemp. By rule 2, Flu does block the path, so we can stop checking. (Rule 1 still says Fever does not block the path.) All paths are blocked, so the two variables are conditionally independent given Flu.

Problem: **(3a)** Are Flu and ExoticTrip independent?

Solution: Independent.

The only path to check is ExoticTrip — Malaria — Fever — Flu. By rule 1, Malaria does not block the path. However, rule 3 says Fever does block the path (none of Fever and its descendants are observed). All paths are blocked, so the two variables are independent.

Problem: **(3b)** Are Flu and ExoticTrip conditionally independent given HighTemp?

Solution: Not independent.

We check the same path as in problem (3a): ExoticTrip — Malaria — Fever — Flu. By rule 1, Malaria still does not block the path. This time, rule 3 says Fever does not block the path either (one of Fever's descendants, HighTemp, is observed). Thus we have an unblocked path between Flu and ExoticTrip, so the two variables are not conditionally independent given HighTemp.

2 Constructing Bayesian Networks

For a joint probability distribution, there are multiple correct Bayesian networks. At the start of this document, we presented just one of the many possible correct Bayesian networks for the Holmes scenario.

A Bayesian network is correct if every independence relationship represented by the network is correct. That is, each relationship (unconditional or conditional) exists in the joint distribution.

We prefer one Bayesian over another if the former requires fewer probabilities to define. Generally, the number of probabilities required is positively correlated with the number of directed edges in the network, so we usually aim to reduce the number of edges in the network.

Here is a procedure to construct a correct Bayesian network.

1. Determine the set of variables for the domain.
2. Order the variables $\{X_1, \dots, X_n\}$.
3. For each variable X_i in the ordering:

   3.1 Choose the node's parents.

   Choose the smallest set of parents from $\{X_1, \dots, X_{i-1}\}$ such that given $\text{Parents}(X_i)$, X_i is independent of all the nodes in $\{X_1, \dots, X_{i-1}\} - \text{Parents}(X_i)$. Formally,

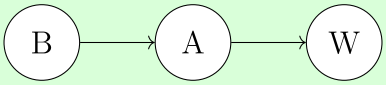
   $$P(X_i|\text{Parents}(X_i)) = P(X_i|X_{i-1} \land \dots \land X_1).$$

   3.2 Create a link from each parent of X_i to the node X_i.

   3.3 Write down the conditional probability table $P(X_i|\text{Parents}(X_i))$.

Let's go through a few examples of applying this procedure. Again, these will correspond to the three keys structures from earlier. Each example will involve recreating a Bayesian network for some variables but with a different ordering. Step 3.3 will not be included because we will revisit it when we discuss learning the probabilities in a Bayesian network based on data, but I will go through choosing parents.

Problem: Consider the Bayesian network:



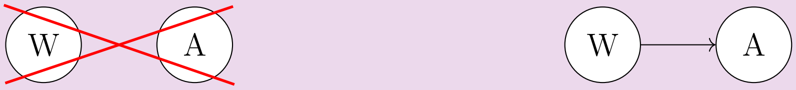
Construct a correct Bayesian network based on the variable ordering: W, A, B.

Solution: Step 1: add W to the network.

Step 2: add A to the network.

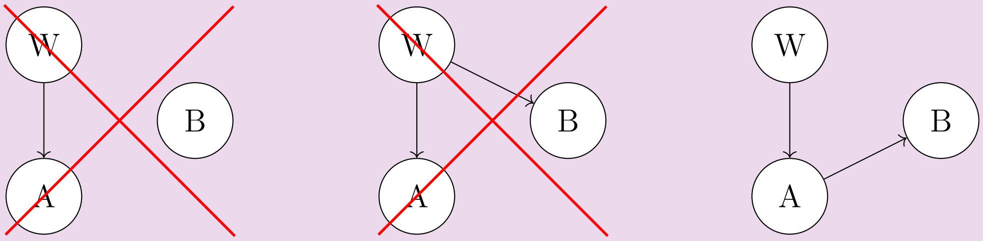
We need to choose the smallest parent set for A such that given A's parent set, A is independent of all other nodes. If A is independent from W, A can simply have no parents. If A is not independent from W, A must have W as a parent.

Note that in the original Bayesian network, A is not independent from W since W is a child of A. Thus we must make W a parent of A here.

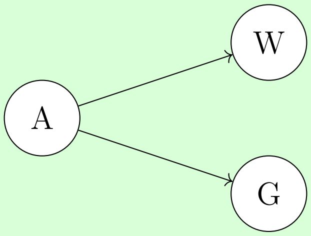


Step 3: add B to the network.

Again, we want to minimize the parent set of B. The parent set could be empty, but that would require B to be independent from A and W. Looking at our original Bayesian network, this does not work. We can try a parent set of size 1 next. We could pick W to be the only parent node of B, but this would require B to be conditionally independent from A given W; again, this does not work. We could pick A to be the only parent node of B, and this would require B to be conditionally independent from W given A. This does work and gives us the final network.



Problem: Consider the Bayesian network:

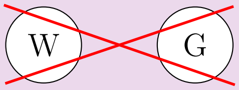


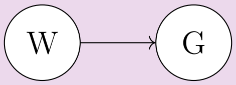
Construct a correct Bayesian network based on the variable ordering: W, G, A.

Solution: Step 1: add W to the network.

Step 2: add G to the network.

We can add G with an empty parent set if G is independent from W. However, this is not the case in the original network (recall the second key structure). Therefore, we need to add G with W as a parent.



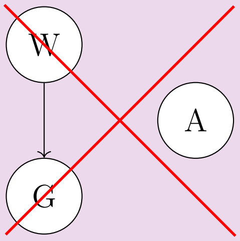


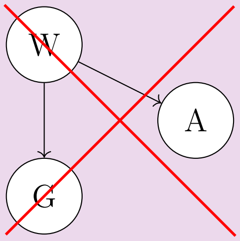
Step 3: add A to the network.

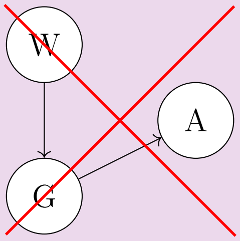
We can add A with no parents if A is independent from W and G. Looking at our original network, though, this is obviously not the case: W and G are children of A.

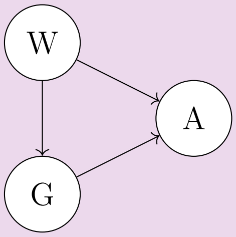
We can try to add A with one parent. To make W the only parent of A, we need A to be conditionally independent from G given W. This is also not the case: G is the child of A. To make G the only parent of A, we need A to be conditionally independent from W given G. Similarly, W is the child of A, so this is not the case either.

Our only option is to make both W and G parents of A.

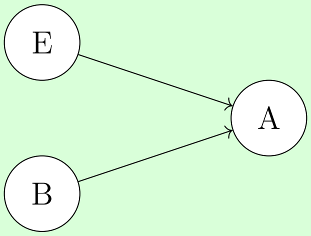








Problem: Consider the Bayesian network:

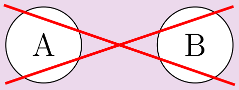


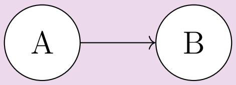
Construct a correct Bayesian network based on the variable ordering: A, B, E.

Solution: Step 1: add A to the network.

Step 2: add B to the network.

We can add B with an empty parent set if B is independent from A, but this is not the case in the original network since A is a child of B. Therefore, we need to add B with A as a parent.



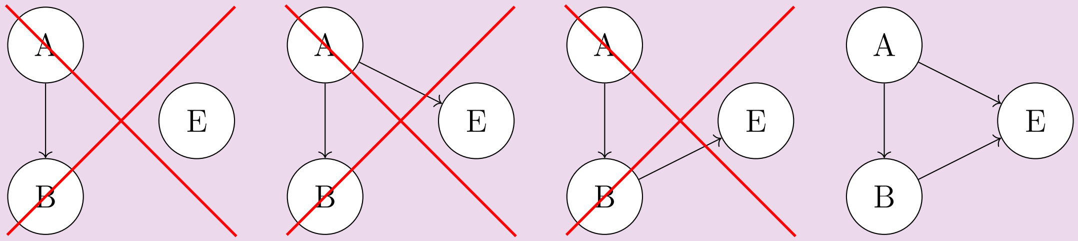


Step 3: add E to the network.

We can add E with no parents if E is independent from A and B. Considering our original network, E and B are indeed independent, but E is a parent of A so E is not independent from A. So, we cannot use an empty parent set.

We can try to add E with one parent. To make A the only parent of E, we need E to be conditionally independent from B given A. By the third key structure (the explaining-away effect), this is not the case. To make B the only parent of E, we need E to be conditionally independent from A given B. A is the child of E, so this is not the case.

Our only option is to make both A and B parents of E.



Notice that an implicit relationship in the original network was that B and E are dependent given A, but this became explicit when we changed the order of the variables.

There are two important points to discuss regarding these examples.

Firstly, in the third example's final answer, you may feel like the links are very unintuitive. For example, why is there a link from Burglary to Earthquake? Burglary is certainly not a cause for Earthquake, so that link doesn't seem to represent a causal relationship. If you're thinking this, you're right—not every link in a Bayesian network represents a causal relationship.

The original Holmes network was constructed by accident in such a way that causes always came before effects, so all the links represented causal relationships. In general, this is not the case. We could reverse all of those links and end up with all links representing some correlation, but not necessarily causation.

Secondly, in the second and third examples, you should have noticed that by changing the order of the variables, we had to increase the number of links in the network. We went from two to three edges in each case, resulting in a more complicated Bayesian network.

Earlier, we mentioned that networks with fewer links are preferred because they are more compact, but how can we come up with these networks? Should we just do this by trial and error, or is there some general rule we can follow instead?

It turns out there is a sort of rule of thumb: you should try to pick a variable ordering that respects the causal relationships. If causes precede effects, we (generally) get a more compact Bayesian network.