# Learning Bayesian Networks

## Introduction to Bayesian Networks

## Joint Probability Distributions

***Definition 1.1*** Suppose we have sample space containing distinct elements. That is:

.

A function which assigns a real number to each event is called probability function on the set of subsets of if it satisfies the following conditions:

1. for
2. For each event that is not an elementary event,

The pair , will be denoted as ***probability space***.

***Theorem 1.1*** Let , be a probability space. Then

1. .
2. for every .
3. For and such that ,

.

***Definition 1.2*** Let and be events such that . Then the conditional probability of E given F, denoted , is given by:

***Definition 1.3*** Two events are independent if one of the following hold:

1. and
2. or

***Definition 1.4*** Two events and are conditionally independent given if and one of the following holds:

1. and

***Definition 1.8*** Let a set of random variables be specified such that each has a countably infinite space. A function, that assigns a real number to every combination of values of the ’s such that the value of is chosen from the space of , is called joint probability distribution of the random variables in if it satisfies the following conditions:

1. For every combination of values of the ’s:

1. We have:

Suppose we have events such that for and

. Such events are called ***mutually exclusive and exhaustive***.

The law of total probability says that for any event we have:

***Theorem 1.2 (Bayes)*** Given two events and such that and we have:

Furthermore, given mutually exclusive and exhaustive events , , , such that for all , we have for :

***Definition 1.5*** Given a probability space , , a ***random variable*** is a function of . The set of values can assume is called ***the space*** of . A random variable is said to be ***discrete*** if its space is finite or countable.

***Theorem 1.3*** Let a set of random variables V be given and let a joint probability distribution of the variables in be specified according to Definition 1.8. Let be the Cartesian product of the sets of all possible values of the random variables. Assign probabilities to elementary events in as follows:

These assignments result in a probability function on according to ***Definition 1.1***. Furthermore, if we let denote a function (random variable in the classical sense) on this sample space which maps each tuple in to the value of in that tuple, then the joint probability distribution of the ’s is the same as the originally specified joint probability distribution.

### Markov Condition

***Definition***: ***ancestral ordering*** is such ordering for which if Y is descendant of X then Y is on the right of X

***Definition 1.9*** Suppose we have a joint probability distribution of the random variables in some set and a DAG = . We say that satisfies ***the Markov condition*** if for each variable conditionally independent of the set of all of its non-descendants given the set of all its parents. If we denote the sets of parents and non-descendants of by and , respectively, then

When satisfies *the Markov condition*, we say and satisfy Markov condition with each other.

If is a root, then its parent set is empty. So in this case it means that the Markov condition means is independent of - . But implies . We have . So we can rewrite the Markov condition as:

***Theorem 1.4*** If satisfies *the Markov condition*, then P is equal to the product of its conditional distributions of all nodes given values of their parents, whenever these conditional distributions exist.

*Proof*: We prove the case where is discrete. Order the nodes in their ancestral ordering. Let , , … , be the resultant ordering. For a given set of values , , … , let be the subset of these values containing the values of ’s parents. We need to show that whenever for ,

We show this using induction on the variables of the network. Assume for some combination of values of the for

*Induction base* :

Since is empty .

*Induction hypothesis* :

Suppose for this combination of values of the that:

*Induction Step*: Prove assuming that the hypothesis for is true

(1.7)

There are two cases:

*Case 1*: for this combination of values:

(1.8)

Clearly (1.8) implies

Furthermore, due to (1.8) and the induction hypothesis, there is some , where such that . So (1.7) holds.

*Case 2*: For this combination of values:

In this case

The first equality is due to the rule for conditional probability, the second is due to the Markov condition and the third one is due to the induction hypothesis.

***Theorem 1.5*** Let a DAG be given in which each node is a random variable, and let a discrete conditional probability distribution given the values of its parents in be specified. Then the product of these conditional distributions yields a joint probability distribution of the variables and satisfies the Markov condition.

*Proof*: Order the nodes according to the ancestral ordering. Let , , , be the resultant ordering. Next define:

where is the set of parents of in and is the specified conditional probability distribution. First, we show that this does indeed yield joint probability distribution. Clearly, for all values of the variables. Therefore to show that we have joint distribution Definition 1.8 and Theorem 1.3 imply that we only need to show that the sum of , as the variables are ranging through their all possible values, equal to 1. To that end:

.

To be done: *show that the specified conditional distributions are the conditional distributions which they notationally represent in the joint distribution*

Finally, we show the Markov condition is satisfied. To do this, we need to show for that whenever if and then , where is the set of non-descendants of in . Since , we only need to show that . First, for a given , order the nodes so that all and only descendants of precede in the ordering. Note that this ordering depends on whereas the ordering in the first part of the proof does not. Clearly then:

Let

= {, ,, }

In what follows means the sum as the variables in go over all of their possible values. Furthermore, notation such as means the variable has a particular value; notation such as means all variables in the set have particular values; and notation such as means some variables in the set may not have particular values. We have that:

In the second to last step, the sums are each equal to one for the following reason. Each is a sum of a product of conditional probability distributions specified for a DAG. In the case of the numerator, that DAG is the subdigraph of our original digraph , consisting of the variables in , and in the case of the denominator, it is the subdigraph consisting of the variables in . Therefore, the fact that each of those sums equals 1 follows from the first part of the proof. Notice that the theorem requires that the specified conditional distributions be discrete. Often in the case of continuous distributions it still holds. For example it holds for Gaussian distributions. However, it does not hold for all continuous conditional distributions. See [[Dawid and Studeny, 1999]](https://github.com/dimitarpg13/learning_bayesian_networks/blob/main/articles/ConditionalProductsAnAlternativeApproachtoConditionalIndependenceDawid1999.pdf) for example in which no joint distribution having the specialized distributions as conditionals even exist.

## Bayesian Networks

***Definition*** (*Bayesian Network*)

Let be the joint probability distribution of the random variables in some set , and be a DAG. We call a Bayesian network if satisfies the Markov condition. Owning to *Theorem 1.4*, is the product of its conditional distributions in and this is the way is always represented in a Bayesian network. Furthermore, owing to *Theorem 1.5*, if we specify a DAG and any discrete conditional distributions (and many continuous ones), we obtain a Bayesian network. This is the way Bayesian networks are constructed in practice.

Creating Bayesian Network using Causal Edges

**Definition** (*causal DAG*) Given a set of random variables , if for every , in we draw an edge from to if and only if is a direct cause of relative to , we call the resulting DAG a ***causal DAG***.

Ascertaining Causal Influences Using Manipulation

Some of what follows is based on a similar discussion to [[Cooper, 1999]](https://github.com/dimitarpg13/learning_bayesian_networks/blob/main/articles/OverviewoftheRepresentationandDIscoveryofCausalRelationshipsUsingBayesianNetworksCooper1999.pdf).