# 8 Bayesian Networks

## 8.1 Probability Basics

Before going into the details of Bayesian Network, let us revise the basics of probability. The following are some basic axioms for probability for some given event $A$ and $B$.

1. $0 \leq P(A) \leq 1$
2. $P(A \cup B) = P(A) + P(B) - P(A \cap B)$
3. $P(True) = P(U) = 1$ where $U$ is the universal set
4. $P(False) = P(\phi) = 0$ where $\phi$ is the empty set

**Definition 1** *Independence: Events $A$ and $B$ are independent if $P(A|B) = P(A)$*

**Definition 2** *Conditional Independence: Given event $B$, event $A$ is conditionally independent of event $C$ if $P(A|B \cap C) = P(A|B)$.*

Additionally, we define the notation for events $a$ and $b$, $P(a, b) \equiv P(a \cap b)$.

**Theorem 3** ***Bayes' Rule*** *states that the probability of event $A$, given that event $B$ has occurred, the the probability of event $A$ and $B$, divided by the unconditional probability of event $B$.*

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

*By extension,*

$$P(A \cap B) = P(B) \cdot P(A|B)$$

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(A) \cdot P(B|A)}{P(B)}$$

**Theorem 4** ***Law of Total Probability*** *states that if $B_1, B_2, B_3, \cdots$ serve as a partition of the sample space $S$, then for any event $A$, we have*

$$P(A) = \sum_i P(A \cap B_i) = \sum_i P(A|B_i) \cdot P(B_i)$$

## 8.2 Partially Observable Environments

Suppose there are two coins in a jar: $C_{50}$ and $C_{90}$. The subscripts for each of the coin represents the bias of the coin. In particular, we have the following,

1. Coin $C_{50}$ where the probability of heads given that this coin is flipped is $P(H|C_{50}) = 50\%$
2. Coin $C_{90}$ where the probability of heads given that this coin is flipped is $P(H|C_{90}) = 90\%$

A man walks to the jar, and picks a coin from the jar, and tosses the coin repeatedly 6 times. The goal of the agent is to make an educated guess of which of coin the man had picked. The agent is unable to identify whether the coin is $C_{50}$ or $C_{90}$ through its sensors; however, he/she is able to observe the results for each of the man's toss.

Prior to the toss, with no further information, it is equally likely to the agent that either coin be picked. Thus, the unconditional probability of the man picking $C_{50}$ or $C_{90}$ are 50% in both cases.

However, with the information observed from the man's toss of the coin, a more accurate judgment can be made. Let's say the agent observed the following results of all 6 coin tosses $\langle HTTHTH \rangle$. The agent can use Bayes' rule to update its probability estimates of the coin being $C_{50}$ or $C_{90}$ condition on this new information. At this point, the agent needs to models its beliefs.

| Before Toss | After Toss |
|---|---|
| $P(C_{50}) = 0.5$ | $P(C_{50}\vert\text{Toss}) =?$ |
| $P(C_{90}) = 0.5$ | $P(C_{90}\vert\text{Toss}) =?$ |

To calculate the $P(C_{50}|\text{Toss})$ and $P(C_{90}|\text{Toss})$, we can use Bayes's rule and law of total probability. Therefore, by applying above two rules, we get $P(C_{50}|\text{Toss}) > P(C_{90}|\text{Toss})$.

$$\begin{aligned} P(C_{50}|\text{Toss}) &= \frac{P(C_{50} \cap \text{Toss})}{P(\text{Toss})} \\ &= \frac{P(\text{Toss}|C_{50}).P(C_{50})}{P(\text{Toss})} \\ &= \frac{0.5 \times (0.5)^6}{0.5 \times (0.5)^6 + 0.5 \times (0.9)^3 \times (0.1)^3} \\ &= 0.95 \end{aligned}$$

$$\begin{aligned} P(C_{90}|\text{Toss}) &= \frac{P(C_{90} \cap \text{Toss})}{P(\text{Toss})} \\ &= \frac{P(\text{Toss}|C_{90}).P(C_{90})}{P(\text{Toss})} \\ &= \frac{0.5 \times (0.9)^3 \times (0.1)^3}{0.5 \times (0.9)^3 \times (0.1)^3 + 0.5 \times (0.5)^6} \\ &= 0.04 \end{aligned}$$

The method of updating the agent's beliefs on the probability of outcomes is summaries by Figure 1, where initially there are two contending models: $M_1$ : Coin $C_{50}$ is chosen and $M_2$ : Coin $C_{90}$ is chosen. Through some partial observations of the coin toss, the agent is able to use conditional probabilities to determine which model to chose.

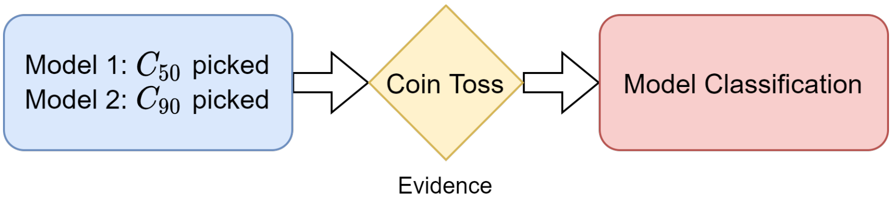


Figure 1: Model classification with conditional probabilities

In fact, this can be generalized to a case of $n$ coins. However, as $n$ grows large, it can be increasingly cumbersome to compute the conditional probabilities. For example, let's consider the case of having 100 coin $C_1, C_2, C_3, \cdots, C_{100}$ where the subscripts indicate the percentage point probabilities of each coin landing a heads when tossed. Suppose you observe $k$ tosses, denoted $Toss$. Then the following have to be computed.

$$\begin{aligned} P(C_1|Toss) &= \frac{P(C_1)P(Toss|C_1)}{P(Toss)} \\ &\vdots \\ P(C_{100}|Toss) &= \frac{P(C_{100})P(Toss|C_{100})}{P(Toss)} \end{aligned}$$

Where each $P(Toss)$ is computed as,

$$P(Toss) = \sum_{i=1}^{100} \frac{1}{100} P(Toss|C_i)$$

Computing $P(Toss)$ is cumbersome and also unnecessary for the agent to find which coin is most likely to be picked. Because the denominator in conditional probability of $P(C_i|Toss)$ have the same $P(Toss)$ for all the coins $C_i$; therefore, we only need to compute the numerator for comparison. In other-words,$\forall i, j$

$$P(C_i|Toss) > P(C_j|Toss) \Leftrightarrow P(C_i)P(Toss|C_i) > P(C_j)P(Toss|C_j)$$

Thus we only need to find the coin with the highest $P(C_i)P(Toss|C_i)$ values for that coin to be most likely to be picked.

***Now, the important question is how to store this conditional probabilities and use them in a efficient manner to chose a model***

Consider a situation where a researcher is interested in predicting how likely a graduate is to do well in a job interview. Some factors to consider could be the graduate's grades or even whether the graduate has to pay a road toll (ERP) on the way to the interview.

The events that have to be tracked are as follows.

1. Grades ($G$), where $G$ means that the graduate's grades are high, and $\bar{G}$ means the graduate's grades are low.

2. Having to pay Electronic Road Pricing ($E$) where $E$ means that a graduate was charged for ERP, and $\bar{E}$ means the graduate was not charged.

3. Job Interview ($I$), where $I$ means that the graduate's interview went well, and $\bar{I}$ means the graduate's interview went awry.

Let's assume that the researcher had collected data from a sample of 600 students, and the joint frequency for every combination of $G$, $E$ and $I$ had been tallied. A naive way to store the values will be a simple table as shown below.

| $G$ | $E$ | $I$ | frequency | $P(C, G, I)$ |
|---|---|---|---|---|
| T | T | T | 160 | 160/600 |
| T | T | F | 60 | 60/600 |
| T | F | T | 240 | 240/600 |
| T | F | F | 40 | 40/600 |
| F | T | T | 10 | 10/600 |
| F | T | F | 60 | 60/100 |
| F | F | T | 10 | 10/600 |
| F | F | F | 20 | 20/600 |

Table 1: The tabulated joint probabilities for combination of events

From the above table 1, we can compute both conditional and unconditional probabilities. For example, the probability of $P(G)$ is the sum of all the joint probabilities in the table where the value of $G$ is true (T).

$$\begin{aligned} P(G) &= P(G, E, I \cup G, E, \bar{I} \cup G, \bar{E}, I \cup G, \bar{E}, \bar{I}) \\ &= P(G, E, I) + P(G, E, \bar{I}) + P(G, \bar{E}, I) + P(G, \bar{E}, \bar{I}) \\ &= \frac{160 + 60 + 240 + 40}{600} \\ &= \frac{500}{600} \end{aligned}$$

Similarly, the conditional probability of $G$ given $E$ can be computed as the sum of the joint probabilities of where both $G$ and $E$ in the row are true divided by the sum of joint probabilities where $E$ is true.

$$\begin{aligned} P(G|E) &= \frac{P(G \cap E)}{P(E)} \\ &= \frac{P(G, E, I) + P(G, E, \bar{I})}{P(G, E, I) + P(G, E, \bar{I}) + P(\bar{G}, E, I) + P(\bar{G}, E, \bar{I})} \\ &= \frac{160 + 60}{160 + 60 + 10 + 60} \\ &= \frac{220}{290} \end{aligned}$$

**Space and Computational Complexity:** The one major drawback is that the tables can be huge when there are more variables. This means that the method of storing the joint probabilities in a tabular form is not scalable to larger problems. If there are 5 variables, then there will need to be $2^5 = 32$ entries. However, to be precise, as the sum of probabilities is 1, we can exclude the last entry (and compute it by subtracting all the other probabilities from 1).

In particular, if there are $n$ variables with two values (true or false), then the space and computational is $O(2^n)$.

## 8.3 Representing Information as Directed Acyclic Graph: Bayesian network

Suppose there are 5 variables:

1. Grades(G)
2. ERP(E)
3. Interview (I)
4. Job Offer (J)
5. Driver's Mood(D)

We can represent the causal information of these variables in the form of a Directed Acyclic Graph (DAG). For example, we know that Grades and having to pay ERP are causal factors of Interview performance. Interview performance in turn directly affect the Job offer. However, we know that having to pay ERP can causally affect the Driver's mood, but the Driver's mood cannot causally affect any other variables. We that we form the graph as shown in Figure 2.

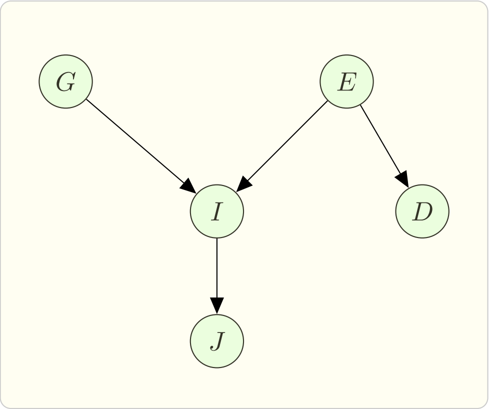


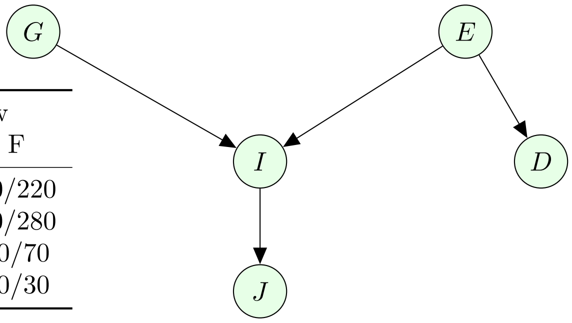
Figure 2: Bayesian Network

In Figure 2, each variable is represented by a node. The causal relation is represented by a directed edge from one variable to another. The network is capturing the fact that how well the interview happens is determined by grades and ERP.

Next, we want to store the conditional probabilities associated with each node in the form of a table. **Given parents, the probability of a node is independent of its non-descendants**, i.e. each node in a Bayesian network is independent of all non-descendants node, given its parents. As per Figure 2, $P(I|G,E,D) = P(I|G,E)$.

| Grades | |
|---|---|
| T | F |
| 500/600 | 100/600 |

| ERP | |
|---|---|
| T | F |
| 290/600 | 310/600 |



| | | Interview | |
|---|---|---|---|
| Grades | ERP | T | F |
| T | T | 160/220 | 60/220 |
| T | F | 240/280 | 40/280 |
| F | T | 10/70 | 60/70 |
| F | F | 10/30 | 20/30 |

| | Driver | |
|---|---|---|
| ERP | T | F |
| T | 230/300 | 70/300 |
| F | 120/300 | 180/300 |

| | Job | |
|---|---|---|
| Interview | T | F |
| T | 380/450 | 70/450 |
| F | 5/100 | 95/100 |

Figure 3: Conditional Probability Table

The Figure 3 represents Bayesian Networks or Inference Networks. Each table of the Figure 3 is called the conditional probability table(CPTs). For Tables associated with nodes(Grades, ERP) that have no causal factors, the conditional probabilities is the same as the unconditional probability of the event.

**How did we come up CPT from Table 1:** We processed the Table 1 in row by row manner to fill up the CPTs.

| $G$ | $E$ | $I$ | Total | $P(I\|G,E)$ |
|---|---|---|---|---|
| $T$ | $T$ | 160 | $160+60=220$ | $\frac{\#I}{\#\text{total}}=\frac{160}{220}$ |
| $T$ | $F$ | 240 | $240+40=280$ | $\frac{\#I}{\#\text{total}}=\frac{240}{280}$ |
| $F$ | $T$ | 10 | $10+60=70$ | $\frac{\#I}{\#\text{total}}=\frac{10}{70}$ |
| $F$ | $F$ | 10 | $10+20=30$ | $\frac{\#I}{\#\text{total}}=\frac{10}{30}$ |

### Space Complexity of Bayesian Network

As discussed earlier, tabular representation requires 32 entries for 5 variables, now with Bayesian representation, it requires 10 rows. Assuming that in the DAG, the maximum number of parents is some positive integer $q$, then the number of entries required in the worse case is $n \cdot 2^q$ with Bayesian network. This makes the space complexity $O(n2^q)$, which is a stark improvement from $2^n$.

### Computing Probabilities in Bayesian Network

Now that we have the network, we want to be able to compute the joint probabilities of multiple variables. We can use the Chain Rule formula to break the computation into multiplication of a series of conditional probabilities which are stored in the Bayesian Network.

**Theorem 5** *<u>Chain Rule:</u> The joint probability of events $x_1, x_2, \cdots, x_n$ is the following,*

$$P(x_1, x_2, \cdots x_n) = P(x_1|x_2, x_3, \cdots x_n)P(x_2|x_3, x_4, \cdots x_n)\cdots P(x_{n-1}|x_n)P(x_n)$$

In the case of our example, one way the joint probability of $G, I, J, E, D$ can be written as is,

$$P(G, I, J, E, D) = P(G|I, J, E, D) \cdot P(I|J, E, D) \cdot P(J|E, D) \cdot P(E|D) \cdot P(D)$$

However, we encounter a problem with this particular ordered application of chain rule. None of the conditional probabilities in the expression above are readily extractable from Bayesian Network that was established. Therefore, this is not a workable way of getting the solution.

Instead, we have to re-order the variables and apply the chain rule in a specific order, that will allow us to extract the conditional probabilities that are actually stored in the Bayesian network. One way to find such an order is to first pick a variable that has no children, remove the node, and continue to do so. Remember, Bayesian network is DAG, therefore there must exists at least one node with no children. One such ordering for above example( 2) is $J, I, G, D, E$.

$$\begin{aligned}
P(G, I, J, E, D) &= P(J, I, G, D, E) \\
&= P(J|I, G, D, E) \cdot P(I|G, D, E) \cdot P(G|D, E) \cdot P(D|E) \cdot P(E) \\
&= P(J|I) \cdot P(I|G, E) \cdot P(G) \cdot P(D|E) \cdot P(E) \\
&\text{conditional independence on non-descendants}
\end{aligned}$$

### Model Selection

Suppose you have given two different models of Bayesian network, and an evidence. Now the important question to ask here which of the Bayesian networks would you choose to keep for future? For example, instead of believing that Driver's mood depends on ERP, we may believe that Driver's mood depends on Student's grades. Now, as shown in Figure 4 and 5, we have 2 Bayesian networks. Also, we have 2 evidences:

1. $e_1 : G = T, E = T, D = F, I = T, J = F$
2. $e_2 : G = T, E = F, D = T, I = T, J = F$

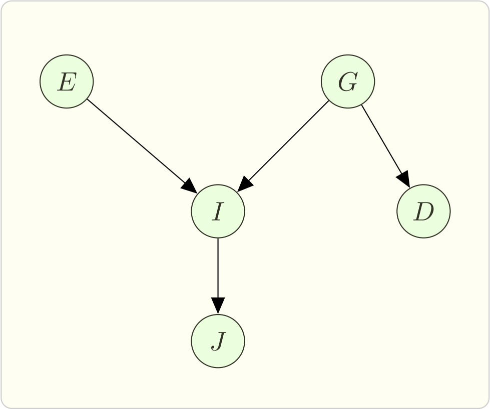


Figure 4: Bayesian Network $M_1$

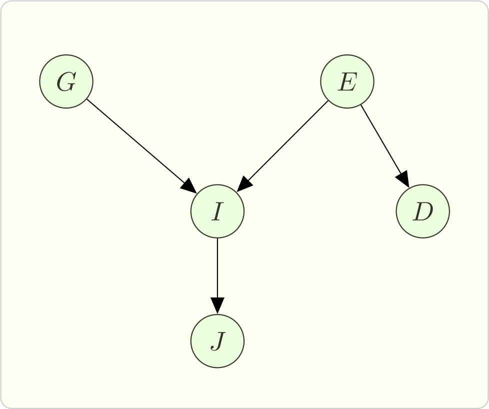


Figure 5: Bayesian Network $M_2$

We want to decide which of the two models is the more likely description of evidences $e_1$ and $e_2$. Thus we need to determine if:

$$P(M_1|\{e_1, e_2\}) \overset{?}{\geq} P(M_2|\{e_1, e_2\})$$
$$\frac{P(M_1|\{e_1, e_2\})}{P(M_2|\{e_1, e_2\})} \overset{?}{\geq} 1$$

As we know:

$$P(M_1|\{e_1, e_2\}) = \frac{P(\{e_1, e_2\}|M_1) \times P(M_1)}{P(\{e_1, e_2\})} \tag{1}$$

$$P(M_2|\{e_1, e_2\}) = \frac{P(\{e_1, e_2\}|M_2) \times P(M_2)}{P(\{e_1, e_2\})} \tag{2}$$

As we are interested in computing the ratio of $P(M_1|\{e_1, e_2\}), P(M_1|\{e_1, e_2\})$, we don't need to compute $P(\{e_1, e_2\})$. We also assume that the data collected are independent of each other. Thus, we have $P(\{e_1, e_2\}|M_1) = P(e_1|M_1)P(e_2|M_1)$, and $P(\{e_1, e_2\}|M_2) = P(e_1|M_2)P(e_2|M_2)$. The final component to consider is $P(M_1)$ and $P(M_2)$. How do we calculate these values?

$P(M_1)$ and $P(M_2)$ reflect the probabilities of each of the models occurring in real life. In fact, it is often difficult to obtain these probabilities, and there are several possible ways to do so:

1. We may choose to start with prior distributions for $P(M_1)$ and $P(M_2)$, based on our own past experiences.

2. We may also base the values of $P(M_1)$ and $P(M_2)$ on historical information from the usage of either of these models.

However, note that it is only necessary for us to figure out the ratio $P(M_1)/P(M_2)$, since we are only trying to compare the magnitude of $P(M_1|\{e_1, e_2\})$ with $P(M_2|\{e_1, e_2\})$.

**Now, the most important question is, How to comeup with these models** We can use any of the local search algorithms to find the good model to work with. We may begin with an arbitrary state, and gradually transition to another state depending on the value assigned to each state, for example suppose we start with model $M_1$(Figure 4), and the neighboring model is model $M_2$(Figure 5), so we transition from $M_1$ to $M_2$ only if given an evidence $e_1$, $P(M_1|\{e_1\}) < P(M_2|\{e_1\})$.

Thus, the algorithm for obtaining an optimal model based on some evidence is as follows:

1. Initialize some random model.

2. At every step, look for a neighbor $N(M)$ of the current model.

3. Compute $\frac{P(\text{evidence}|N(M))}{P(\text{evidence}|M)}$, where $N(M)$ is a neighbor of model $M$.

4. Transition to a new model only if ratio is greater than 1, else terminate with model $M$.