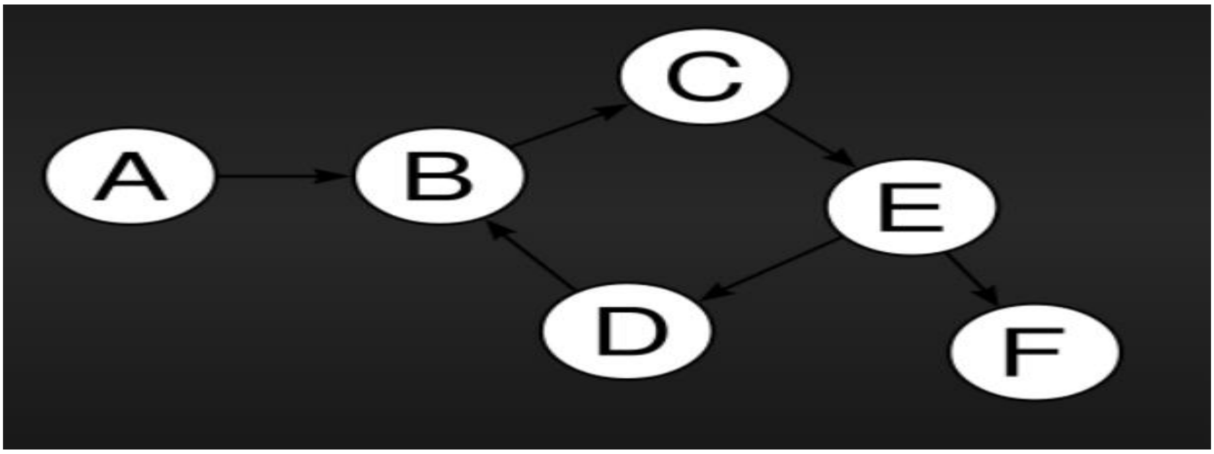
1. **Describe Bayesian network and Bayesian classifiers. Provide and explain one example of applications of the Bayesian network [20 points]**

Bayesian Network and Bayesian Classifiers:

Bayesian Networks are also known as Belief networks or casual networks. Bayesian Networks are used to solve complex problems where we have limited information and resources. These networks are implemented in many advanced technologies in Machine Learning and Artificial Intelligence. Belief Network is a type of Probabilistic Graphical Modelling (PGM) technique. Bayesian Networks are used to compute the uncertainties by using the concept of probability. These networks model the uncertainties or unknowns by using Directed Acyclic Graphs (DAG). Overall, these networks are based on the concept of probability and can be represented using Directed Acyclic Graphs. Directed Acyclic Graph is nothing but a statistical graph that contains a set of nodes and links. The nodes denote random variables and the links denote the relation between the nodes. A simple example of DAG is shown in the below figure where A, B, C, D, E, F are random variables and the links between them are the relations between them.



A is the parent node and B is the child node in the above diagram (if A, B nodes are considered). Similarly C is the child node of B (if B, C nodes are considered). The output of a Directed Acyclic Graph will tell the uncertainty of an event occurring based on the conditional probability distribution of each of the random variable. We need to first understand Joint Probability and Conditional Probability to get a clear idea. Joint Probability is a measure of two events happening at the same time i.e., p (A and B). It is the probability of intersection of A and B. Conditional Probability of an event B is the probability that the event will occur given that an event A has already occurred. P (B|A) is the probability of event B occurring, given that event A occurs. If A and B are independent events P (B|A) = P (B). If A and B are dependent events P (B|A) = P (A and B) / P (A). The probability of a random variable depends on its parent's node. Overall, Bayesian Network is the simplest yet most effective technique applied in predictive modelling and descriptive analysis, diagnostic analysis, prescriptive analysis. Bayesian Networks makes it easy to distinguish correlation from causation as it allows us to see various independent causes at once. All of this is done because Machine learning algorithms do not work on the intuition, they work on the data. Bayesian Networks that model sequence of variables are called Dynamic Bayesian Networks. In numerous applications, the connection between the attribute set and the class variable is non- deterministic. In other words, we can say the class label of a test record can’t be assumed with certainty even though its attribute set is the same as some of the training examples. These circumstances may emerge due to the noisy data or the presence of certain confusing factors that influence classification, but it is not included in the analysis. For example, consider the task of predicting the occurrence of whether an individual is at risk for liver illness based on individuals eating habits and working efficiency. Bayes, who first utilized conditional probability to provide an algorithm that uses evidence to calculate limits on an unknown parameter.

Bayes theorem is expressed mathematically by the following equation that is given below.

$$P(X/Y) = \frac{P(Y/X)P(X)}{P(Y)}$$

Where X and Y are the events and P(Y) not equal to 0. (X/Y) is a **conditional probability** that describes the occurrence of event **X** is given that **Y** is true. It is known as Posterior probability. P(Y/X) is a **conditional probability** that describes the occurrence of event **Y** is given that **X** is true. It is the likelihood. P(X) is prior probability and P(Y) is the marginal probability. These are mostly used for classification. There are different types of Bayesian Classifiers namely, Bayes Optimal Classifier, Gibbs Algorithm, Naïve Bayes Classifier. Bayes optimal classifier is best used when hypothesis space and prior probability are given. But it is time consuming and not a practical application. So there comes Naïve Bayes with an assumption that all the attributes are conditionally independent. The assumptions made by Naive Bayes are not generally correct in real-world situations. In-fact, the independence assumption is never correct but often works well in practice.

This is in brief about Bayesian Networks and Bayesian Classifiers.

Applications of Bayesian Network:

There are many practical applications of Bayesian Network. Few of them are the following:

1. Monty Hall Problem:

   We use Bayesian Network to understand the probability of winning if the participant decides to switch his choice.

2. Gene Regulatory Network: (GNR)

   A GNR contains various DNA segments of a cell that interact with other cell contents through protein and RNA expression products. The predictions of its behaviour can be analysed using Bayesian Networks.

3. Optimized web search:

   With limited information, they can perform huge amount of prediction

4. Image Processing:

   Bayesian Networks use mathematical operations to convert images into digital format. It also allows image enhancement.

5. Spam Filter:

   Google is able to filter the spam emails using Bayesian spam filter, which is the most robust filter

6. Bio monitoring:

   Quantifying the concentration of chemicals couldn't get any easier than with Bayesian Networks. In this, the amount of blood and tissue in humans is measured using indicators.

7. Turbo Code:

   Bayesian Networks are used to create turbo codes that are high-performance forward error correction codes. These are used in 3G and 4G mobile networks.

8. Document Classification:

   These networks are also used for classification of the text documents

9. Disease Diagnostics:

   These networks help in detection and prevention of the diseases.

Solving Monty Hall Problem using Bayes Theorem:

You're on a gameshow called "**Let's Make a Deal**". There are 3 closed doors in front of you. Behind each door is a prize. One door has a **car**, one door has **breath mints**, and one door has a **bar of soap**. You'll get the prize behind the door you pick, but you don't know which prize is behind which door. Obviously you want the car! So you pick **door A**. Before opening **door A**, the host of the show, Monty Hall, now opens **door B,** revealing a bar of soap. He then asks you if you'd like to change your guess. Should you? My gut told me it doesn't matter if I change my guess or not. There are 2 doors so the odds of winning the car with each is 50%. Unfortunately for me, that's 100% wrong.

This is the famous Monty Hall problem. By working through Bayes Theorem, we can calculate the actual odds of winning the car if we stick with **door A**, or switch to **door C**.

Bayes Theorem

Bayes Theorem describes probabilities related to an event, given another event occurs.

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

Bayes Theorem

A = An event.

B = Another event.

P(A|B) = posterior = The probability of an event occurring, given another event occurs

P(B|A) = likelihood = The probability event B occurs, if event A occurs.

P(A)= prior = The probability an event occurs, before you know if the other event occurs.

P(B) = The normalizing constant.

Bayes Theorem + Monty Hall

Note: A, B and C in calculations here are the names of doors, not A and B in Bayes Theorem.

Now let's calculate the components of Bayes Theorem in the context of the Monty Hall problem.

Let's assume **we pick door A**, then **Monty opens door B**. Monty wouldn't open C if the car was behind C so we only need to calculate 2 posteriors:

1. P(door=A | opens =B), the probability A is correct if Monty opened B,
2. P(door=C | opens =B) , the probability C is correct if Monty opened B

Prior: P(A)

The probability of any door being correct before we pick a door is 1/3. Prizes are randomly arranged behind doors and we have no other information. So the **prior**, P(A), of any door being correct is **1/3**.

1. P(door=A), the prior probability that door A contains the car = 1/3

2. P(door=C), the prior probability that door C contains the car = 1/3

Likelihood: P(B|A)

If the car is actually behind door A, then Monty can open door B or C. So the probability of opening either is 50%.

If the car is actually behind door C then monty can only open door B. He cannot open A, the door we picked. He also cannot open door C because it has the car behind it.

1. P(opens =B | door =A), the likelihood Monty opened door B if door A is correct = 1/2
2. P(opens =B | door =C), the likelihood Monty opened door B if door C is correct = 1

Numerator: P(A) x P(B|A)

1. P(door=A) * P(opens =B | door =A), = 1/3 x 1/2 = 1/6
2. P(door=C) * P(opens =B | door =C)= 1/3 x 1 = 1/3

Normalizing Constant: P(B)

In cases where analyzed events cover all possible options and don't overlap, we can take the sum of the numerators.

P(B)= 1/6 + 1/3 = 3/6 = 1/2

Posterior: P(A|B)

Now we just need to do the remaining math.

1. P(door=A | opens =B), = (1/6) / (1/2) = 1/3
2. P(door=C | opens =B)= (1/3) / (1/2) = 2/3

This leaves us with a with a higher probability of winning if we change doors after Monty opens a door. This is in brief about the Monty Hall problem

Source:

https://www.javatpoint.com/data-mining-bayesian-classifiers

https://www.upgrad.com/blog/bayesian-networks/

https://www.youtube.com/watch?v=SkC8S3wuIfg

https://towardsdatascience.com/solving-the-monty-hall-problem-with-bayes-theorem-893289953e16

Class Material