Unit-V - Bayes classification Methods

Bayesian classifiers are statistical classifiers. They can Predict class membership Probabilities such as the Probability that a given tuple belongs to a particular class.

* Bayesiam classification is based on Baye's theorem.
* Bayesian classifiers have high accuracy and speed when applied to larger databases.
* The performance of Baye's classifier algorithm is high when compared to the performance of decision tree and neural networks.
* Naive Bayesian classifiers assume that the effect of an attribute value on a given class is independent of the values of other attributes. This assumption is called class conditional independence. 6
* This assumption to be taken to made the computations simpler. so that this algorithm is called as naive.

→ Bayes theorem:-

Let X be a data sample whose class label is unknown. Let H be some hypothesis, such that the data sample X belongs to a specified class C.

* For classification problems, we want to determine P(H/X), the Probability of the hypothesis H holds given the observed data sample X.
* P(H/X) is the posterior Probability of H conditioned on X. For example, consider a data sample consists of fruits described by their color and shape..
* suppose that X is red and rand and that H is the hypothesis that X is an apple. P(H) is prior probability.
* Similarly P(X/H) is the posterior probability of X conditioned on H. That is ~~the~~ it is the Probability that X is red and rand given that we know that it is true that X is an apple.
* P(X) is prior probability of X.
* Bayes theorem gives Probability of an event based on Prior knowledge of conditions.

$$P(H/X) = \frac{P(X/H) \cdot P(H)}{P(X)} \quad \text{(or)} \quad P(A/B) = \frac{P(B/A) \cdot P(A)}{P(B)}$$

where $P(A|B)$ = Hypothesis
$P(B|A)$ = Likelihood
$P(A)$ = Prior & $P(B)$ = Marginal

Proof :- $P(A|B) = \frac{P(A\cap B)}{P(B)}$ → ①

$\|^{ly}$ $P(B|A) = \frac{P(B\cap A)}{P(A)}$ → ②

from eqⁿ-① $P(A|B).P(B) = P(A\cap B)$ → ③

from eqⁿ-② $P(B|A).P(A) = P(B\cap A)$ → ④

From eqⁿ-③ & eqⁿ-④ we can write

$P(A|B).P(B) = P(A\cap B)$
$P(B|A).P(A) = P(A\cap B)$ ⇒ $P(A|B).P(B) = P(B|A).P(A)$

⇒ $\boxed{P(A|B) = \frac{P(B|A).P(A)}{P(B)}}$

where A is hypothesis
B is given data

$P(A|B)$ = Finding Probability of hypothesis i.e A when the Probability of training examples are given

$P(B|A)$ = Finding Probability of given data provided with the probability of hypothesis that is true.

$P(A)$ = Prob. of hypothesis before observing the given data

$P(B)$ = Probability of given data.

→ Naive Bayesian classification:-

The Naive Bayesian classification works as follows.

Step1- Each data sample is represented by an 'n' dimensional feature vector $X = (x_1, x_2 \ldots x_n)$ [n attributes $A_1, A_2 \ldots A_n$]

Step2. Suppose that there are 'm' classes $C_1, C_2 \ldots C_m$. Given an unknown data sample, X (i.e no class label), the classifier will predict that X belongs the class having the highest posterior probability, conditioned on X. That is, the naive Bayesian classifier assigns an unknown sample X to the class C_i iff $P(C_i|X) > P(C_j|X)$ for $1 \le j \le m, j \ne i$

By Bayes theorem, $P(C_i|X)$ is obtained as

$$\boxed{P(C_i|X) = \frac{P(X|C_i).P(C_i)}{P(X)}}$$

Step 3 – As $P(X)$ is constant for all classes, only $P(X|C_i).P(C_i)$ is maximized.

Step 3.1 – If class prior probabilities are not known then it is commonly assumed that the classes are equally likely, that is $P(C_1) = P(C_2) = \cdots = P(C_m)$. and therefore maximize $P(X|C_i)$.

Step 3.2 – If class prior probabilities are known then use
$P(C_i) = \frac{S_i}{S}$ when S_i = no of samples in class C_i
S = no. of samples.

Step 4 – Given data sets with many attributes, it would be extremely computationally expensive to compute $P(X|C_i)$. In order to reduce computation in evaluating $P(X|C_i)$, the naive assumption of class conditional independence is made. This presumes that the values of the attributes are conditionally independent of one another, given the class label of the sample, that is there are no dependence relationships among that attributes. That is

$$\boxed{P(X|C_i) = \prod_{i=1}^{n} P(X_k|C_i)}$$

The probabilities $P(X_1|C_i)$, $P(X_2|C_i)$ ---- $P(X_n|C_i)$ can be estimated from the training samples, where

a) If A_k is categorical then $P(X_k|C_i) = \frac{S_{ik}}{S_i}$
where S_{ik} = no. of samples of class C_i having value X_k for A_k
S_i = no. of training samples belongs to C_i.

b) If A_k is continuous valued then the

$$P(X_k|C_i) = g(x_i, \mu_{C_i}, \sigma_{C_i}) = \frac{1}{\sqrt{2\pi}\,\sigma_{C_i}} e^{-\frac{(x_k-\mu_{C_i})^2}{2\sigma_{C_i}^2}}$$

where $g(x_i, \mu_{C_i}, \sigma_{C_i})$ is the Gussian density function for attribute A_k.
μ_{C_i} = mean
σ_{C_i} = standard deviation.

Step 5 – In order to classify an unknown sample X, $P(X|C_i).P(C_i)$ is evaluated for each class C_i. Sample X is then assigned to the class C_i iff

$$P(X|C_i).P(C_i) > P(X|C_j).P(C_j) \text{ for } 1 \le j \le m,\ j \ne i$$

Ex.:

Instance	a_1	a_2	a_3	class
1	1	2	1	1
2	0	0	1	1
3	2	1	2	2
4	1	2	1	2
5	0	1	2	1
6	2	2	2	2
7	1	0	1	1
8	2	1	1	
9	0	1	1	

step1 :-

$$P(class = 1) = \frac{4}{7} = 0.571$$

$$P(class = 2) = \frac{3}{7} = 0.428$$

step2. construct tables for all attributes which contains

Attribute	class1	class2	----
value1			
value2			
⋮			

In the given example three (3) tables need to be constructed

attribute (a_1)	class=1	class=2
0	2	0
1	2	1
2	0	2

attribute 3

attribute a_3	class1	class2
0	0	0
1	3	1
2	1	2

a_2	class1	class2
0	2	0
1	1	1
2	1	2

Step3 For test set 2, 1, 1

$$X_1 = \{\overset{a_1}{2}, \overset{a_2}{1}, \overset{a_3}{1}\}$$

step3.1 $P(A_1 = 2 \mid class = 1) = \frac{0}{4} = 0$

$P(A_1 = 2 \mid class = 2) = 2/3 = 0.666$

step3.2 $P(A_2 = 1 \mid class = 1) = 1/4 = 0.25$

$P(A_2 = 1 \mid class = 2) = 1/3 = 0.333$

step 3.3 $P(A_3 = 1 \mid class = 1) = 3/4 = 0.75$

$P(A_3 = 1 \mid class = 2) = 1/3 = 0.333.$

Step 4 – $P(A|c_i) = P(A_1|c_i) \cdot P(A_2|c_i) \cdots P(A_n|c_i)$

$P(A|class=1) = 0 \times 0.25 \times 0.75 = 0$

$P(A|class=2) = 0.666 \times 0.333 \times 0.333 = 0.007$

Step 5 – $P(X_1|C_1) \cdot P(C_1) = 0 \times 0.571 = \boxed{0}$

$P(X_1|C_2) \cdot P(C_2) = 0.007 \times 0.428 = \boxed{0.002996}$

Step 6 – By observing step 5, we can conclude that the given data sample : class label is class 2, because class 2 probability is more.

→ Bayesian Belief Networks :- (Probabilistic Graphical Model – PGM)

Naive Bayesian classifier makes an assumption of class conditional independence that is class label of a sample, the values of the attributes are conditionally independent of one another.

* In naive Bayesian classifier the values of the attribute are conditionally independent of one another.
* Bayesian belief networks specify joint conditional probability distributions. These allow class conditional independencies to be defined between subsets of variables.
* This provides the graphical model of casual relationships, on which learning can be performed.
* These networks are also known as belief networks, Bayesian networks and probabilistic networks.
* A belief network is defined by two components :

i) Direct Acyclic Graph :- In which, each node represents a random variable and each arc represents a probabilistic dependence.

* If an arc is drawn from a node Y to a node Z, then Y is a parent or immediate predecessor of Z, and Z is descendent of Y.
* Each variable is conditionally independent of its non-descendents in the graph, given its parents.
* The variables may be discrete or continuous valued.

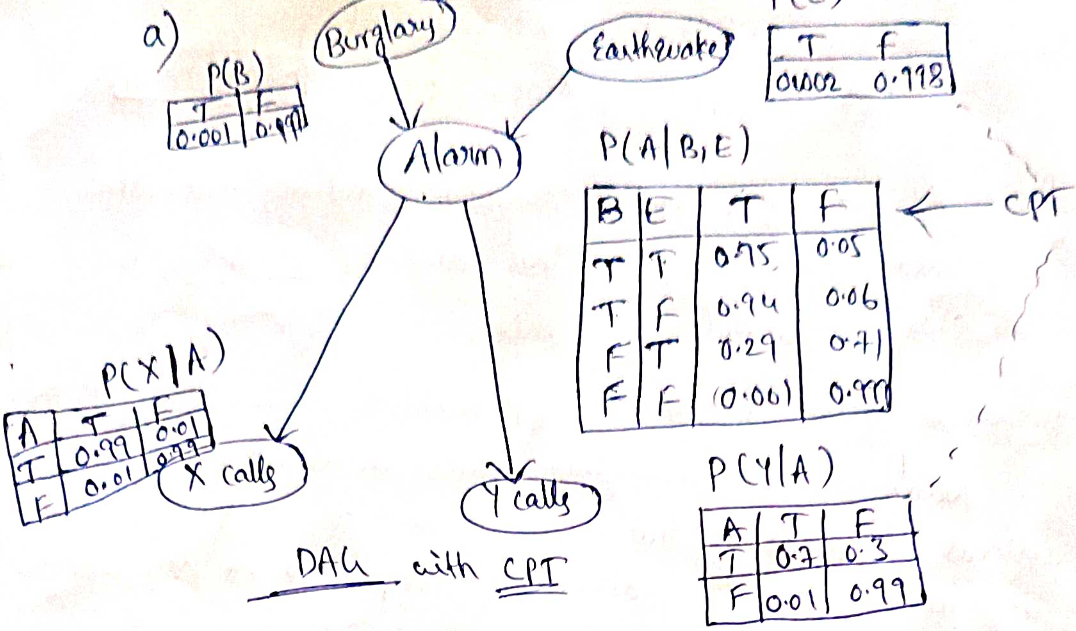
ii) Conditional Probability Table (CPT) :-

The CPT for a variable Z specifies the conditional distribution $P(Z|parent(Z))$

* The joint probability of any tuple $(z_1, z_2 \dots z_n)$ corresponding to the variables & attributes $z_1, z_2, \dots z_n$ is computed by

$$P(z_1, z_2 \dots z_n) = \prod_{i=1}^{n} P(z_i \mid Parent(z_i)).$$

a)



P(B)

T	F
0.001	0.999

P(E)

T	F
0.002	0.998

P(A|B,E)

B	E	T	F
T	T	0.95	0.05
T	F	0.94	0.06
F	T	0.29	0.71
F	F	0.001	0.999

P(X|A)

A	T	F
T	0.99	0.01
F	0.01	0.99

P(Y|A)

A	T	F
T	0.7	0.3
F	0.01	0.99

For example, $P(\overset{T}{X}, \overset{T}{Y}, \overset{T}{A}, \overset{F}{\sim B}, \overset{F}{\sim E}) =$

That means X calls is true, Y calls is True, Alarm has rang, Burglary is False and Earthquake is False then find the probability

$$P(\underset{z_1}{X}, \underset{z_2}{Y}, \underset{z_3}{A}, \underset{z_4}{\sim B}, \underset{z_5}{\sim E}) = \prod_{i=1}^{n} P(z_i \mid Parent(z_i))$$

$$= P(X|A) \cdot P(Y|A) \cdot P(A|\sim B, \sim E) \cdot P(\sim B) \cdot P(\sim E)$$

Substitute values from above CPT

~~= 0.99.~~

$$= 0.99 \times 0.7 \times \overset{0.001}{\cancel{0.94}} \times 0.999 \times 0.998 = \underline{0.691544}$$

→ **Training Bayesian Belief networks:-**

How does a Bayesian belief network learn?

* To learn, no. of scenarios are possible. The network structure may be given in advance & inferred from the data.
* The network variables may be observable & hidden in all & some of training samples. The case of hidden data is also referred to as missing values & incomplete data.
* If the network structure is known and the variables are observable, then training the network is straightforward.
* It consists of CPT entries, as is similarly done when computing the probabilities involved in naive Bayesian classification.

* When the network structure is given and some of the variables are hidden, then a method of gradient descent can be used to train the belief network. The object is to learn the values for the CPT entries.

* Let S be a set of 'S' training samples, $X_1, X_2 \dots X_s$

* Let w_{ijk} is the upper probability CPT entry for the variable $Y_i = y_{ij}$ having the parents $U_i = u_{ik}$

* The weights w is initialised to random probability values. At each iteration of gradient descent updates the weights

Algorithm:-

step1 :- The weights are updated by

$$w_{ijk} \leftarrow w_{ijk} + (l) \frac{\partial \ln P_w(S)}{\partial w_{ijk}}$$

where l is the learning rate, = step size.

step2 -

$$\frac{\partial \ln P_w(S)}{\partial w_{ijk}} = \sum_{d=1}^{s} \frac{P(Y_i = y_{ij}, U_i = u_{ik} | X_d)}{w_{ijk}}$$

where X_d = training sample in S.

P = probability

step3 - Renormalise the weights:-

Because the weights are probability values, they must be in the range 0.0 to 1.0.