UNIT –IV: Syllabus

Classification: Alterative Techniques, Bayes' Theorem, Naïve Bayesian Classification, Bayesian Belief Networks

UNIT-IV

DATA CLASSIFICATION (Alternative Techniques)

Classification is a form of data analysis that extracts models describing important data classes. Such models, called classifiers, predict categorical (discrete, unordered) class labels. For example, we can build a classification model to categorize bank loan applications as either safe or risky. Such analysis can help provide us with a better understanding of the data at large. Many classification methods have been proposed by researchers in machine learning, pattern recognition, and statistics.

Classification: Alternative Techniques:

Bayesian Classification:

- Bayesian classifiers are statistical classifiers.
- They can predict class membership probabilities, such as the probability that a given tuple belongs to a particular class.
- Bayesian classification is based on Bayes' theorem.

Bayes' Theorem:

- Let X be a data tuple. In Bayesian terms, X is considered ― "evidence" and it is described by measurements made on a set of n attributes.
- Let H be some hypothesis, such as that the data tuple X belongs to a specified class C.
- For classification problems, we want to determine P(H|X), the probability that the hypothesis H holds given the ―evidence‖ or observed data tuple X.
- P(H|X) is the posterior probability, or a posteriori probability, of H conditioned on X.
- Bayes' theorem is useful in that it provides a way of calculating the posterior probability, P(H|X), from P(H), P(X|H), and P(X).

$$P\ H\ X = \frac{P\ X\ H\ P(H)}{P(X)}$$

Naïve Bayesian Classification:

The naïve Bayesian classifier, or simple Bayesian classifier, works as follows:

1. Let=be a training set of tuples and their associated class labels. As usual, each tuple is represented by an n-dimensional attribute vector, X = (x1, x2, …,xn), depicting n measurements made on the tuple from n attributes, respectively, A1, A2, …, An.
2. Suppose that there are m classes, C1, C2, …, Cm. Given a tuple, X, the classifier will predict that X belongs to the class having the highest posterior probability, conditioned on X. That is, the naïve Bayesian classifier predicts that tuple X belongs to the class Ci if and only if

$$P\ C_i\ X > P\ C_j\ X\ for\ 1 \leq j \leq m, j \neq i.$$

Thus we maximize$(C_j\ |X)$. The class Ci for which $(C_j\ |X)$. is maximized is called the

maximum posteriori hypothesis. By Bayes' theorem

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

3. As P(X) is constant for all classes, only P(X|Ci)P(Ci) need be maximized. If the class prior probabilities are not known, then it is commonly assumed that the classes are equally likely, that is, P(C1) = P(C2) = …= P(Cm), and we would therefore maximize P(X|Ci). Otherwise, we maximize P(X|Ci)P(Ci).
4. Given data sets with many attributes, it would be extremely computationally expensive to compute P(X|Ci). In order to reduce computation in evaluating P(X|Ci), 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 tuple. Thus,

$$P(X|C_i) = \prod_{k=1}^{n} P(x_k|C_i)$$
$$= P(x_1|C_1) \times P(x_2|C_2) \times \ldots\ldots\ldots \times P(x_n|C_i)$$

5. We can easily estimate the probabilities P(x1|Ci), P(x2|Ci), : : : , P(xn|Ci) from the training tuples.
6. For each attribute, we look at whether the attribute is categorical or continuous-valued. For instance, to compute P(X|Ci), we consider the following:

- If A_k is categorical, then $P(x_k|Ci)$ is the number of tuples of class Ci in=havingthe value x_k for A_k, divided by |Ci ,D| the number of tuples of class Ci in D.
- If A_k is continuous-valued, then we need to do a bit more work, but the calculation is pretty straightforward.

Example:

age	income	student	credit_rating	buys_computer
youth	high	no	fair	no
youth	high	no	excellent	no
middle_aged	high	no	fair	yes
senior	medium	no	fair	yes
senior	low	yes	fair	yes
senior	low	yes	excellent	no
middle_aged	low	yes	excellent	yes
youth	medium	no	fair	no
youth	low	yes	fair	yes
senior	medium	yes	fair	yes
youth	medium	yes	excellent	yes
middle_aged	medium	no	excellent	yes
middle_aged	high	yes	fair	yes
senior	medium	no	excellent	no

We wish to predict the class label of a tuple using naïve Bayesian classification, given the same training data above. The training data were shown above in Table. The data tuples are described by the attributes *age*, *income*, *student*, and *credit rating*. The class label

attribute, *buys computer*, has two distinct values (namely, {yes, no}). Let *C*1 correspond to the class *buys computer=yes* and *C*2 correspond to b*uys computer=no*. The tuple we wish to classify is

X={age= "youth", income= "medium", student= "yes", credit_rating= "fair"}

We need to maximize P(X|Ci)P(Ci), for i=1,2. P(Ci), the prior probability of each class, can be computed based on the training tuples:

P(*buys computer = yes*) = 9/14 = 0.643
P(*buys computer = no*) = 5/14 = 0.357

To compute P(X|Ci), for *i* = 1, 2, we compute the following conditional probabilities:

P(*age = youth* | *buys computer = yes*) = 2/9 = 0.222
P(*income=medium* | *buys computer=yes*) = 4/9 = 0.444
P(*student=yes* | *buys computer=yes*) = 6/9 = 0.667
P(*credit rating=fair* | *buys computer=yes*) = 6/9 = 0.667

P(*age=youth* | *buys computer=no*) = 3/5 = 0.600
P(*income=medium* | *buys computer=no*) = 2/5 = 0.400
P(*student=yes* | *buys computer=no*) = 1/5 = 0.200
P(*credit rating=fair* | *buys computer=no*) = 2/5 = 0.400

Using these probabilities, we obtain

$$P(\boldsymbol{X} \mid buys\ computer=yes) = P(age=youth \mid buys\ computer=yes) \times P(income=medium \mid buys\ computer=yes) \times P(student=yes \mid buys\ computer=yes) \times P(credit\ rating=fair \mid buys\ computer=yes)$$
$$= 0.222 \times 0.444 \times 0.667 \times 0.667 = 0.044.$$

Similarly,

$$P(\boldsymbol{X} \mid buys\ computer=no) = 0.600 \times 0.400 \times 0.200 \times 0.400 = 0.019.$$

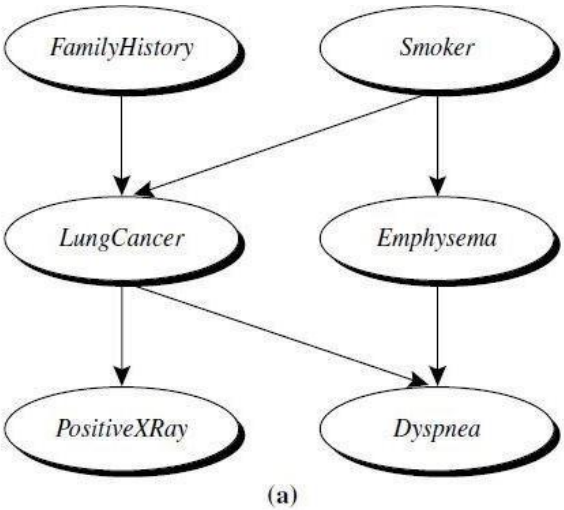
To find the class, C_i, that P(X|Ci)P(Ci), we compute

$$P(\boldsymbol{X} \mid buys\ computer=yes)\ P(buys\ computer=yes) = 0.044 \times 0.643 = 0.028$$
$$P(\boldsymbol{X} \mid buys\ computer=no)\ P(buys\ computer=no) = 0.019 \times 0.357 = 0.007$$

Therefore, the naïve Bayesian classifier predicts *buys computer = yes* for tuple ***X***.

Bayesian Belief Networks

- *Bayesian belief networks*—probabilistic graphical models, which unlike naïve Bayesian classifiers allow the representation of dependencies among subsets of attributes.
- The naïve Bayesian classifier makes the assumption of class conditional independence, that is, given the class label of a tuple, the values of the attributes are assumed to be conditionally independent of one another.
- When the assumption holds true, then the naïve Bayesian classifier is the most accurate in comparison with all other classifiers.
- They provide a graphical model of causal relationships, on which learning can be performed.
- A belief network is defined by two components—a *directed acyclic graph* and a set of *conditional probability tables* (See Figure).
- Each node in the directed acyclic graph represents a random variable. The variables may be discrete- or continuous-valued.
- They may correspond to actual attributes given in the data or to "hidden variables" believed to form a relationship.
- 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 a **descendant** of *Y*.
- *Each variable is conditionally independent of its nondescendants in the graph, given its parents.*



	FH, S	*FH, ~S*	*~FH, S*	*~FH, ~S*
LC	0.8	0.5	0.7	0.1
~LC	0.2	0.5	0.3	0.9

(b)

Simple Bayesian belief network. (a) A proposed causal model, represented by a directed acyclic graph. (b) The conditional probability table for the values of the variable *LungCancer* (*LC*) showing each possible combination of the values of its parent nodes, *FamilyHistory* (*FH*) and *Smoker* (*S*). *Source:* Adapted from Russell, Binder, Koller, and Kanazawa [RBKK95].

For example, having lung cancer is influenced by a person's family history of lung cancer, as well as whether or not the person is a smoker. Note that the variable *PositiveXRay* is independent of whether the patient has a family history of lung cancer or is a smoker, given that we know the patient has lung cancer.

In other words, once we know the outcome of the variable *LungCancer*, then the variables *FamilyHistory* and *Smoker* do not provide any additional information regarding *PositiveXRay*. The arcs also show that the variable *LungCancer* is conditionally independent of *Emphysema*, given its parents, *FamilyHistory* and *Smoker*.

A belief network has one **conditional probability table (CPT)** for each variable. The CPT for a variable *Y* specifies the conditional distribution $P(Y|Parents(Y))$, where $Parents(Y)$ are the parents of *Y*. Figure (b) shows a CPT for the variable *LungCancer*. The conditional probability for each known value of *LungCancer* is given for each possible combination of the values of its parents. For instance, from the upper leftmost and bottom rightmost entries, respectively.