**“What are Bayesian classifiers?”** Bayesian classifiers are statistical classifiers. They can predict class membership probabilities such as the probability that a given tuple belongs to a particular class.

Na¨ıve Bayesian classifiers assume that the effect of an attribute value on a given class is independent of the values of the other attributes. This assumption is called classconditional independence. It is made to simplify the computations involved and, in this sense, is considered “na¨ıve.”

**Bayes’ Theorem**

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. In other words, we are looking for the probability that tuple X belongs to class C, given that we know the attribute description of X.

P(H|X) is the posterior probability, or a posteriori probability, of H conditioned on X

Suppose that H is the hypothesis that our customer will buy a computer. Then P(H|X) reflects the probability that customer X will buy a computer given that we know the customer’s age and income

In contrast, P(H) is the prior probability, or a priori probability, of H.

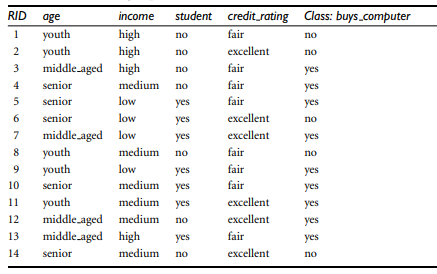
The posterior probability, P(H|X), is based on more information (e.g., customer information) than the prior probability, P(H), which is independent of X

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). Bayes’ theorem is

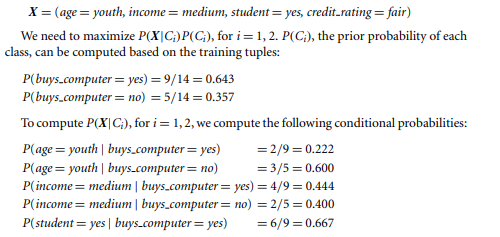


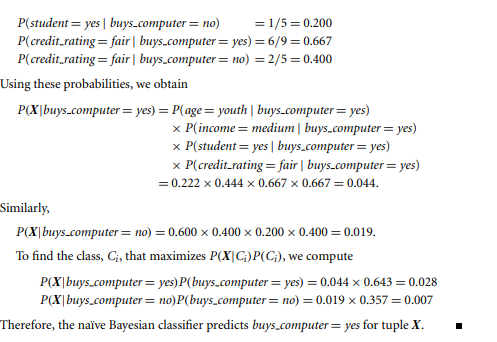
Bayesian classifiers are also useful in that they provide a theoretical justification for other classifiers that do not explicitly use Bayes’ theorem. For example, under certain assumptions, it can be shown that many neural network and curve-fitting algorithms output the maximum posteriori hypothesis, as does the na¨ıve Bayesian classifier

Predicting a class label using na¨ıve Bayesian classification. We wish to predict the class label of a tuple using na¨ıve Bayesian classification



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 C1 correspond to the class buys computer = yes and C2 correspond to buys computer = no. The tuple we wish to classify is





**Rule-Based Classification**

learned model is represented as a set of IF-THEN rules. We first examine how such rules are used for classification. We then study ways, in which they can be generated, either from a decision tree or directly from the training data using a sequential covering algorithm.

Rules are a good way of representing information or bits of knowledge.

A rule-based classifier uses a set of IF-THEN rules for classification.

An IF-THEN rule is an expression of the form IF condition THEN conclusion.

An example is rule R1,

R1: IF age = youth AND student = yes THEN buys computer = yes.

The “IF” part (or left side) of a rule is known as the rule antecedent or precondition. The “THEN” part (or right side) is the rule consequent. In the rule antecedent, the condition consists of one or more attribute tests (e.g., age = youth and student = yes) that are logically ANDed. The rule’s consequent contains a class prediction (in this case, we are predicting whether a customer will buy a computer). R1 can also be written as

