

length is given the highest priority because it is expected to contain the best set of rules. The total description length for a class is given by $L_{\text{exception}} + g \times L_{\text{model}}$, where $L_{\text{exception}}$ is the number of bits needed to encode the misclassified examples, L_{model} is the number of bits needed to encode the model, and g is a tuning parameter whose default value is 0.5. The tuning parameter depends on the number of redundant attributes present in the model. The value of the tuning parameter is small if the model contains many redundant attributes.

5.1.6 Characteristics of Rule-Based Classifiers

A rule-based classifier has the following characteristics:

- The expressiveness of a rule set is almost equivalent to that of a decision tree because a decision tree can be represented by a set of mutually exclusive and exhaustive rules. Both rule-based and decision tree classifiers create rectilinear partitions of the attribute space and assign a class to each partition. Nevertheless, if the rule-based classifier allows multiple rules to be triggered for a given record, then a more complex decision boundary can be constructed.
- Rule-based classifiers are generally used to produce descriptive models that are easier to interpret, but gives comparable performance to the decision tree classifier.
- The class-based ordering approach adopted by many rule-based classifiers (such as RIPPER) is well suited for handling data sets with imbalanced class distributions.

5.2 Nearest-Neighbor classifiers

The classification framework shown in Figure 4.3 involves a two-step process:

- (1) an inductive step for constructing a classification model from data, and
- (2) a deductive step for applying the model to test examples. Decision tree and rule-based classifiers are examples of **eager learners** because they are designed to learn a model that maps the input attributes to the class label as soon as the training data becomes available. An opposite strategy would be to delay the process of modeling the training data until it is needed to classify the test examples. Techniques that employ this strategy are known as **lazy learners**. An example of a lazy learner is the **Rule classifier**, which memorizes the entire training data and performs classification only if the attributes of a test instance match one of the training examples exactly. An obvious drawback of

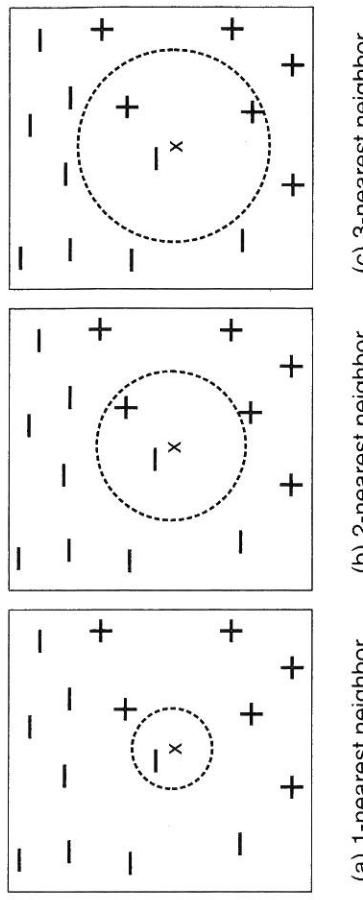


Figure 5.7. The 1-, 2-, and 3-nearest neighbors of an instance.

this approach is that some test records may not be classified because they do not match any training example.

One way to make this approach more flexible is to find all the training examples that are relatively similar to the attributes of the test example. These examples, which are known as **nearest neighbors**, can be used to determine the class label of the test example. The justification for using nearest neighbors is best exemplified by the following saying: *“If it walks like a duck, quacks like a duck, and looks like a duck, then it’s probably a duck.”* A nearest-neighbor classifier represents each example as a data point in a d -dimensional space, where d is the number of attributes. Given a test example, we compute its proximity to the rest of the data points in the training set, using one of the proximity measures described in Section 2.4 on page 65. The k -nearest neighbors of a given example z refer to the k points that are closest to z .

Figure 5.7 illustrates the 1-, 2-, and 3-nearest neighbors of a data point located at the center of each circle. The data point is classified based on the class labels of its neighbors. In the case where the neighbors have more than one label, the data point is assigned to the majority class of its nearest neighbors. In Figure 5.7(a), the 1-nearest neighbor of the data point is a negative example. Therefore the data point is assigned to the negative class. If the number of nearest neighbors is three, as shown in Figure 5.7(c), then the neighborhood contains two positive examples and one negative example. Using the majority voting scheme, the data point is assigned to the positive class. In the case where there is a tie between the classes (see Figure 5.7(b)), we may randomly choose one of them to classify the data point.

The preceding discussion underscores the importance of choosing the right value for k . If k is too small, then the nearest-neighbor classifier may be

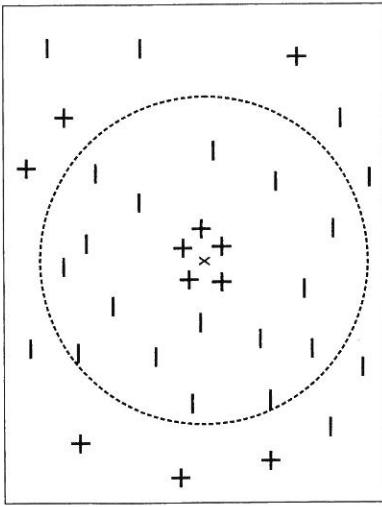


Figure 5.8. k -nearest neighbor classification with large k .

susceptible to overfitting because of noise in the training data. On the other hand, if k is too large, the nearest-neighbor classifier may misclassify the test instance because its list of nearest neighbors may include data points that are located far away from its neighborhood (see Figure 5.8).

5.2.1 Algorithm

A high-level summary of the nearest-neighbor classification method is given in Algorithm 5.2. The algorithm computes the distance (or similarity) between each test example $z = (\mathbf{x}', y')$ and all the training examples $(\mathbf{x}, y) \in D$ to determine its nearest-neighbor list, D_z . Such computation can be costly if the number of training examples is large. However, efficient indexing techniques are available to reduce the amount of computations needed to find the nearest neighbors of a test example.

Algorithm 5.2 The k -nearest neighbor classification algorithm.

- 1: Let k be the number of nearest neighbors and D be the set of training examples.
 - 2: **for** each test example $z = (\mathbf{x}', y')$ **do**
 - 3: Compute $d(\mathbf{x}', \mathbf{x})$, the distance between z and every example, $(\mathbf{x}, y) \in D$.
 - 4: Select $D_z \subseteq D$, the set of k closest training examples to z .
 - 5: $y' = \operatorname{argmax}_v \sum_{(\mathbf{x}_i, y_i) \in D_z} I(v = y_i)$
 - 6: **end for**
-

Once the nearest-neighbor list is obtained, the test example is classified based on the majority class of its nearest neighbors:

$$\text{Majority Voting: } y' = \operatorname{argmax}_v \sum_{(\mathbf{x}_i, y_i) \in D_z} I(v = y_i), \quad (5.7)$$

where v is a class label, y_i is the class label for one of the nearest neighbors, and $I(\cdot)$ is an indicator function that returns the value 1 if its argument is true and 0 otherwise.

In the majority voting approach, every neighbor has the same impact on the classification. This makes the algorithm sensitive to the choice of k , as shown in Figure 5.7. One way to reduce the impact of k is to weight the influence of each nearest neighbor \mathbf{x}_i according to its distance: $w_i = 1/d(\mathbf{x}', \mathbf{x}_i)^2$. As a result, training examples that are located far away from z have a weaker impact on the classification compared to those that are located close to z . Using the distance-weighted voting scheme, the class label can be determined as follows:

$$\text{Distance-Weighted Voting: } y' = \operatorname{argmax}_v \sum_{(\mathbf{x}_i, y_i) \in D_z} w_i \times I(v = y_i). \quad (5.8)$$

5.2.2 Characteristics of Nearest-Neighbor Classifiers

The characteristics of the nearest-neighbor classifier are summarized below:

- Nearest-neighbor classification is part of a more general technique known as instance-based learning, which uses specific training instances to make predictions without having to maintain an abstraction (or model) derived from data. Instance-based learning algorithms require a proximity measure to determine the similarity or distance between instances and a classification function that returns the predicted class of a test instance based on its proximity to other instances.

- Lazy learners such as nearest-neighbor classifiers do not require model building. However, classifying a test example can be quite expensive because we need to compute the proximity values individually between the test and training examples. In contrast, eager learners often spend the bulk of their computing resources for model building. Once a model has been built, classifying a test example is extremely fast.

- Nearest-neighbor classifiers make their predictions based on local information, whereas decision tree and rule-based classifiers attempt to find

a global model that fits the entire input space. Because the classification decisions are made locally, nearest-neighbor classifiers (with small values of k) are quite susceptible to noise.

- Nearest-neighbor classifiers can produce arbitrarily shaped decision boundaries. Such boundaries provide a more flexible model representation compared to decision tree and rule-based classifiers that are often constrained to rectilinear decision boundaries. The decision boundaries of nearest-neighbor classifiers also have high variability because they depend on the composition of training examples. Increasing the number of nearest neighbors may reduce such variability.

• Nearest-neighbor classifiers can produce wrong predictions unless the appropriate proximity measure and data preprocessing steps are taken. For example, suppose we want to classify a group of people based on attributes such as height (measured in meters) and weight (measured in pounds). The height attribute has a low variability, ranging from 1.5 m to 1.85 m, whereas the weight attribute may vary from 90 lb. to 250 lb. If the scale of the attributes are not taken into consideration, the proximity measure may be dominated by differences in the weights of a person.

5.3 Bayesian Classifiers

In many applications the relationship between the attribute set and the class variable is non-deterministic. In other words, the class label of a test record cannot be predicted with certainty even though its attribute set is identical to some of the training examples. This situation may arise because of noisy data or the presence of certain confounding factors that affect classification but are not included in the analysis. For example, consider the task of predicting whether a person is at risk for heart disease based on the person's diet and workout frequency. Although most people who eat healthily and exercise regularly have less chance of developing heart disease, they may still do so because of other factors such as heredity, excessive smoking, and alcohol abuse. Determining whether a person's diet is healthy or the workout frequency is sufficient is also subject to interpretation, which in turn may introduce uncertainties into the learning problem.

This section presents an approach for modeling probabilistic relationships between the attribute set and the class variable. The section begins with an introduction to the **Bayes theorem**, a statistical principle for combining prior

knowledge of the classes with new evidence gathered from data. The use of the Bayes theorem for solving classification problems will be explained, followed by a description of two implementations of Bayesian classifiers: naïve Bayes and the Bayesian belief network.

5.3.1 Bayes Theorem

Consider a football game between two rival teams: Team 0 and Team 1. Suppose Team 0 wins 65% of the time and Team 1 wins the remaining matches. Among the games won by Team 0, only 30% of them come from playing on Team 1's football field. On the other hand, 75% of the victories for Team 1 are obtained while playing at home. If Team 1 is to host the next match between the two teams, which team will most likely emerge as the winner?

This question can be answered by using the well-known Bayes theorem. For completeness, we begin with some basic definitions from probability theory. Readers who are unfamiliar with concepts in probability may refer to Appendix C for a brief review of this topic.

Let X and Y be a pair of random variables. Their joint probability, $P(X = x, Y = y)$, refers to the probability that variable X will take on the value x and variable Y will take on the value y . A conditional probability is the probability that a random variable will take on a particular value given that the outcome for another random variable is known. For example, the conditional probability $P(Y = y|X = x)$ refers to the probability that the variable Y will take on the value y , given that the variable X is observed to have the value x . The joint and conditional probabilities for X and Y are related in the following way:

$$P(X, Y) = P(Y|X) \times P(X) = P(X|Y) \times P(Y). \quad (5.9)$$

Rearranging the last two expressions in Equation 5.9 leads to the following formula, known as the Bayes theorem:

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)}. \quad (5.10)$$

The Bayes theorem can be used to solve the prediction problem stated at the beginning of this section. For notational convenience, let X be the random variable that represents the team hosting the match and Y be the random variable that represents the winner of the match. Both X and Y can be cla

take on values from the set $\{0, 1\}$. We can summarize the information given in the problem as follows:

Probability Team 0 wins is $P(Y = 0) = 0.65$.

Probability Team 1 wins is $P(Y = 1) = 1 - P(Y = 0) = 0.35$.

Probability Team 1 hosted the match it won is $P(X = 1|Y = 1) = 0.75$.

Probability Team 1 hosted the match won by Team 0 is $P(X = 1|Y = 0) = 0.3$.

Our objective is to compute $P(Y = 1|X = 1)$, which is the conditional probability that Team 1 wins the next match it will be hosting, and compares it against $P(Y = 0|X = 1)$. Using the Bayes theorem, we obtain

$$\begin{aligned} P(Y = 1|X = 1) &= \frac{P(X = 1|Y = 1) \times P(Y = 1)}{P(X = 1)} \\ &= \frac{P(X = 1|Y = 1) \times P(Y = 1)}{P(X = 1, Y = 1) + P(X = 1, Y = 0)} \\ &= \frac{P(X = 1|Y = 1) \times P(Y = 1)}{P(X = 1|Y = 1) \times P(Y = 1) + P(X = 1|Y = 0)P(Y = 0)} \\ &= \frac{0.75 \times 0.35}{0.75 \times 0.35 + 0.3 \times 0.65} \\ &= 0.5738, \end{aligned}$$

where the law of total probability (see Equation C.5 on page 722) was applied in the second line. Furthermore, $P(Y = 0|X = 1) = 1 - P(Y = 1|X = 1) = 0.4262$. Since $P(Y = 1|X = 1) > P(Y = 0|X = 1)$, Team 1 has a better chance than Team 0 of winning the next match.

5.3.2 Using the Bayes Theorem for Classification

Before describing how the Bayes theorem can be used for classification, let us formalize the classification problem from a statistical perspective. Let \mathbf{X} denote the attribute set and Y denote the class variable. If the class variable has a non-deterministic relationship with the attributes, then we can treat \mathbf{X} and Y as random variables and capture their relationship probabilistically using $P(Y|\mathbf{X})$. This conditional probability is also known as the **posterior probability** for Y , as opposed to its **prior probability**, $P(Y)$.

During the training phase, we need to learn the posterior probabilities $P(Y|\mathbf{X})$ for every combination of \mathbf{X} and Y based on information gathered from the training data. By knowing these probabilities, a test record \mathbf{X}' can be classified by finding the class Y' that maximizes the posterior probability,

$P(Y'|\mathbf{X})$. To illustrate this approach, consider the task of predicting whether a loan borrower will default on their payments. Figure 5.9 shows a training set with the following attributes: Home Owner, Marital Status, and Annual Income. Loan borrowers who defaulted on their payments are classified as Yes, while those who repaid their loans are classified as No.

| Tid | Home Owner | Marital Status | Annual Income | Defaulted Borrower |
|-----|------------|----------------|---------------|--------------------|
| 1 | Yes | Single | 125K | No |
| 2 | No | Married | 100K | No |
| 3 | No | Single | 70K | No |
| 4 | Yes | Married | 120K | No |
| 5 | No | Divorced | 95K | Yes |
| 6 | No | Married | 60K | No |
| 7 | Yes | Divorced | 220K | No |
| 8 | No | Single | 85K | Yes |
| 9 | No | Married | 75K | No |
| 10 | No | Single | 90K | Yes |

Figure 5.9. Training set for predicting the loan default problem.

Suppose we are given a test record with the following attribute set: $\mathbf{X} = (\text{Home Owner} = \text{No}, \text{Marital Status} = \text{Married}, \text{Annual Income} = \$120\text{K})$. To classify the record, we need to compute the posterior probabilities $P(\text{Yes}|\mathbf{X})$ and $P(\text{No}|\mathbf{X})$ based on information available in the training data. If $P(\text{Yes}|\mathbf{X}) > P(\text{No}|\mathbf{X})$, then the record is classified as Yes; otherwise, it is classified as No. Estimating the posterior probabilities accurately for every possible combination of class label and attribute value is a difficult problem because it requires a very large training set, even for a moderate number of attributes. The Bayes theorem is useful because it allows us to express the posterior probability in terms of the prior probability $P(Y)$, the class-conditional probability $P(\mathbf{X}|Y)$, and the evidence, $P(\mathbf{X})$:

$$P(Y|\mathbf{X}) = \frac{P(\mathbf{X}|Y) \times P(Y)}{P(\mathbf{X})}. \quad (5.11)$$

When comparing the posterior probabilities for different values of Y , the denominator term, $P(\mathbf{X})$, is always constant, and thus, can be ignored. The

prior probability $P(Y)$ can be easily estimated from the training set by computing the fraction of training records that belong to each class. To estimate the class-conditional probabilities $P(\mathbf{X}|Y)$, we present two implementations of Bayesian classification methods: the naïve Bayes classifier and the Bayesian belief network. These implementations are described in Sections 5.3.3 and 5.3.5, respectively.

5.3.3 Naïve Bayes Classifier

A naïve Bayes classifier estimates the class-conditional probability by assuming that the attributes are conditionally independent, given the class label y . The conditional independence assumption can be formally stated as follows:

$$P(\mathbf{X}|Y = y) = \prod_{i=1}^d P(X_i|Y = y), \quad (5.12)$$

where each attribute set $\mathbf{X} = \{X_1, X_2, \dots, X_d\}$ consists of d attributes.

Conditional Independence

Before delving into the details of how a naïve Bayes classifier works, let us examine the notion of conditional independence. Let \mathbf{X} , \mathbf{Y} , and \mathbf{Z} denote three sets of random variables. The variables in \mathbf{X} are said to be conditionally independent of \mathbf{Y} , given \mathbf{Z} , if the following condition holds:

$$P(\mathbf{X}|\mathbf{Y}, \mathbf{Z}) = P(\mathbf{X}|\mathbf{Z}). \quad (5.13)$$

An example of conditional independence is the relationship between a person's arm length and his or her reading skills. One might observe that people with longer arms tend to have higher levels of reading skills. This relationship can be explained by the presence of a confounding factor, which is age. A young child tends to have short arms and lacks the reading skills of an adult. If the age of a person is fixed, then the observed relationship between arm length and reading skills disappears. Thus, we can conclude that arm length and reading skills are conditionally independent when the age variable is fixed.

The conditional independence between \mathbf{X} and \mathbf{Y} can also be written into a form that looks similar to Equation 5.12:

$$\begin{aligned}
 P(\mathbf{X}, \mathbf{Y} | \mathbf{Z}) &= \frac{P(\mathbf{X}, \mathbf{Y}, \mathbf{Z})}{P(\mathbf{Z})} \\
 &= \frac{P(\mathbf{X}, \mathbf{Y}, \mathbf{Z})}{P(\mathbf{Y}, \mathbf{Z})} \times \frac{P(\mathbf{Y}, \mathbf{Z})}{P(\mathbf{Z})} \\
 &= P(\mathbf{X} | \mathbf{Y}, \mathbf{Z}) \times P(\mathbf{Y} | \mathbf{Z}) \\
 &= P(\mathbf{X} | \mathbf{Z}) \times P(\mathbf{Y} | \mathbf{Z}),
 \end{aligned} \tag{5.14}$$

where Equation 5.13 was used to obtain the last line of Equation 5.14.

How a Naïve Bayes Classifier Works

With the conditional independence assumption, instead of computing the class-conditional probability for every combination of \mathbf{X} , we only have to estimate the conditional probability of each X_i , given Y . The latter approach is more practical because it does not require a very large training set to obtain a good estimate of the probability.

To classify a test record, the naïve Bayes classifier computes the posterior probability for each class Y :

$$P(Y | \mathbf{X}) = \frac{P(Y) \prod_{i=1}^d P(X_i | Y)}{P(\mathbf{X})}. \tag{5.15}$$

Since $P(\mathbf{X})$ is fixed for every Y , it is sufficient to choose the class that maximizes the numerator term, $P(Y) \prod_{i=1}^d P(X_i | Y)$. In the next two subsections, we describe several approaches for estimating the conditional probabilities $P(X_i | Y)$ for categorical and continuous attributes.

Estimating Conditional Probabilities for Categorical Attributes

For a categorical attribute X_i , the conditional probability $P(X_i = x_i | Y = y)$ is estimated according to the fraction of training instances in class y that take on a particular attribute value x_i . For example, in the training set given in Figure 5.9, three out of the seven people who repaid their loans also own a home. As a result, the conditional probability for $P(\text{Home Owner}=\text{Yes} | \text{No})$ is equal to $3/7$. Similarly, the conditional probability for defaulted borrowers who are single is given by $P(\text{Marital Status} = \text{Single} | \text{Yes}) = 2/3$.

Estimating Conditional Probabilities for Continuous Attributes

There are two ways to estimate the class-conditional probabilities for continuous attributes in naïve Bayes classifiers:

1. We can discretize each continuous attribute and then replace the continuous attribute value with its corresponding discrete interval. This approach transforms the continuous attributes into ordinal attributes. The conditional probability $P(X_i|Y = y)$ is estimated by computing the fraction of training records belonging to class y that falls within the corresponding interval for X_i . The estimation error depends on the discretization strategy (as described in Section 2.3.6 on page 57), as well as the number of discrete intervals. If the number of intervals is too large, there are too few training records in each interval to provide a reliable estimate for $P(X_i|Y)$. On the other hand, if the number of intervals is too small, then some intervals may aggregate records from different classes and we may miss the correct decision boundary.
2. We can assume a certain form of probability distribution for the continuous variable and estimate the parameters of the distribution using the training data. A Gaussian distribution is usually chosen to represent the class-conditional probability for continuous attributes. The distribution is characterized by two parameters, its mean, μ , and variance, σ^2 . For each class y_j , the class-conditional probability for attribute X_i is

$$P(X_i = x_i|Y = y_j) = \frac{1}{\sqrt{2\pi}\sigma_{ij}} \exp^{-\frac{(x_i - \mu_{ij})^2}{2\sigma_{ij}^2}}. \quad (5.16)$$

The parameter μ_{ij} can be estimated based on the sample mean of X_i (\bar{x}_i) for all training records that belong to the class y_j . Similarly, σ_{ij}^2 can be estimated from the sample variance (s^2) of such training records. For example, consider the annual income attribute shown in Figure 5.9. The sample mean and variance for this attribute with respect to the class No are

$$\begin{aligned} \bar{x} &= \frac{125 + 100 + 70 + \dots + 75}{7} = 110 \\ s^2 &= \frac{(125 - 110)^2 + (100 - 110)^2 + \dots + (75 - 110)^2}{7(6)} = 2975 \\ s &= \sqrt{2975} = 54.54. \end{aligned}$$

Given a test record with taxable income equal to \$120K, we can compute its class-conditional probability as follows:

$$P(\text{Income} = 120 | \text{No}) = \frac{1}{\sqrt{2\pi}(54.54)} \exp^{-\frac{(120 - 110)^2}{2 \times 2975}} = 0.0072.$$

Note that the preceding interpretation of class-conditional probability is somewhat misleading. The right-hand side of Equation 5.16 corresponds to a **probability density function**, $f(X_i; \mu_{ij}, \sigma_{ij})$. Since the function is continuous, the probability that the random variable X_i takes a particular value is zero. Instead, we should compute the conditional probability that X_i lies within some interval, x_i and $x_i + \epsilon$, where ϵ is a small constant:

$$\begin{aligned} P(x_i \leq X_i \leq x_i + \epsilon | Y = y_j) &= \int_{x_i}^{x_i + \epsilon} f(X_i; \mu_{ij}, \sigma_{ij}) dX_i \\ &\approx f(x_i; \mu_{ij}, \sigma_{ij}) \times \epsilon. \end{aligned} \quad (5.17)$$

Since ϵ appears as a constant multiplicative factor for each class, it cancels out when we normalize the posterior probability for $P(Y|\mathbf{X})$. Therefore, we can still apply Equation 5.16 to approximate the class-conditional probability $P(X_i|Y)$.

Example of the Naïve Bayes Classifier

Consider the data set shown in Figure 5.10(a). We can compute the class-conditional probability for each categorical attribute, along with the sample mean and variance for the continuous attribute using the methodology described in the previous subsections. These probabilities are summarized in Figure 5.10(b).

To predict the class label of a test record $\mathbf{X} = (\text{Home Owner}=\text{No}, \text{Marital Status}=\text{Married}, \text{Income}=\$120K)$, we need to compute the posterior probabilities $P(\text{No}|\mathbf{X})$ and $P(\text{Yes}|\mathbf{X})$. Recall from our earlier discussion that these posterior probabilities can be estimated by computing the product between the prior probability $P(Y)$ and the class-conditional probabilities $\prod_i P(X_i|Y)$, which corresponds to the numerator of the right-hand side term in Equation 5.15.

The prior probabilities of each class can be estimated by calculating the fraction of training records that belong to each class. Since there are three records that belong to the class Yes and seven records that belong to the class

| Tid | Home Owner | Marital Status | Annual Income | Defaulted Borrower |
|-----|------------|----------------|---------------|--------------------|
| 1 | Yes | Single | 125K | No |
| 2 | No | Married | 100K | No |
| 3 | No | Single | 70K | No |
| 4 | Yes | Married | 120K | No |
| 5 | No | Divorced | 95K | Yes |
| 6 | No | Married | 60K | No |
| 7 | Yes | Divorced | 220K | No |
| 8 | No | Single | 85K | Yes |
| 9 | No | Married | 75K | No |
| 10 | No | Single | 90K | Yes |

(a)

Figure 5.10. The naïve Bayes classifier for the loan classification problem.

$P(\text{Home Owner} = \text{Yes} | \text{No}) = 3/7$
 $P(\text{Home Owner} = \text{No} | \text{No}) = 4/7$
 $P(\text{Home Owner} = \text{Yes} | \text{Yes}) = 0$
 $P(\text{Home Owner} = \text{No} | \text{Yes}) = 1$
 $P(\text{Marital Status} = \text{Single} | \text{No}) = 2/7$
 $P(\text{Marital Status} = \text{Divorced} | \text{No}) = 1/7$
 $P(\text{Marital Status} = \text{Married} | \text{No}) = 4/7$
 $P(\text{Marital Status} = \text{Single} | \text{Yes}) = 2/3$
 $P(\text{Marital Status} = \text{Divorced} | \text{Yes}) = 1/3$
 $P(\text{Marital Status} = \text{Married} | \text{Yes}) = 0$

For Annual Income:
If class=No: sample mean=110
sample variance=2975
If class=Yes: sample mean=90
sample variance=25

$$\begin{aligned} P(\mathbf{X}|\text{No}) &= P(\text{Home Owner} = \text{No} | \text{No}) \times P(\text{Status} = \text{Married} | \text{No}) \\ &\quad \times P(\text{Annual Income} = \$120\text{K} | \text{No}) \\ &= 4/7 \times 4/7 \times 0.0072 = 0.0024. \end{aligned}$$

$$\begin{aligned} P(\mathbf{X}|\text{Yes}) &= P(\text{Home Owner} = \text{No} | \text{Yes}) \times P(\text{Status} = \text{Married} | \text{Yes}) \\ &\quad \times P(\text{Annual Income} = \$120\text{K} | \text{Yes}) \\ &= 1 \times 0 \times 1.2 \times 10^{-9} = 0. \end{aligned}$$

Putting them together, the posterior probability for class No is $P(\text{No} | \mathbf{X}) = \alpha \times 7/10 \times 0.0024 = 0.0016\alpha$, where $\alpha = 1/P(\mathbf{X})$ is a constant term. Using a similar approach, we can show that the posterior probability for class Yes is zero because its class-conditional probability is zero. Since $P(\text{No} | \mathbf{X}) > P(\text{Yes} | \mathbf{X})$, the record is classified as No.

when $Y = 0$, but is independent of A when $Y = 1$. For simplicity, assume that the class-conditional probabilities for B are the same as for A . Given a record with attributes $A = 0, B = 0$, we can compute its posterior probabilities as follows:

$$\begin{aligned} P(Y = 0|A = 0, B = 0) &= \frac{P(A = 0|Y = 0)P(B = 0|Y = 0)P(Y = 0)}{P(A = 0, B = 0)} \\ &= \frac{0.16 \times P(Y = 0)}{P(A = 0, B = 0)}. \end{aligned}$$

$$\begin{aligned} P(Y = 1|A = 0, B = 0) &= \frac{P(A = 0|Y = 1)P(B = 0|Y = 1)P(Y = 1)}{P(A = 0, B = 0)} \\ &= \frac{0.36 \times P(Y = 1)}{P(A = 0, B = 0)}. \end{aligned}$$

If $P(Y = 0) = P(Y = 1)$, then the naïve Bayes classifier would assign the record to class 1. However, the truth is,

$$P(A = 0, B = 0|Y = 0) = P(A = 0|Y = 0) = 0.4,$$

because A and B are perfectly correlated when $Y = 0$. As a result, the posterior probability for $Y = 0$ is

$$\begin{aligned} P(Y = 0|A = 0, B = 0) &= \frac{P(A = 0, B = 0|Y = 0)P(Y = 0)}{P(A = 0, B = 0)} \\ &= \frac{0.4 \times P(Y = 0)}{P(A = 0, B = 0)}, \end{aligned}$$

which is larger than that for $Y = 1$. The record should have been classified as class 0.

5.3.4 Bayes Error Rate

Suppose we know the true probability distribution that governs $P(\mathbf{X}|Y)$. The Bayesian classification method allows us to determine the ideal decision boundary for the classification task, as illustrated in the following example.

Example 5.3. Consider the task of identifying alligators and crocodiles based on their respective lengths. The average length of an adult crocodile is about 15 feet, while the average length of an adult alligator is about 12 feet. Assuming

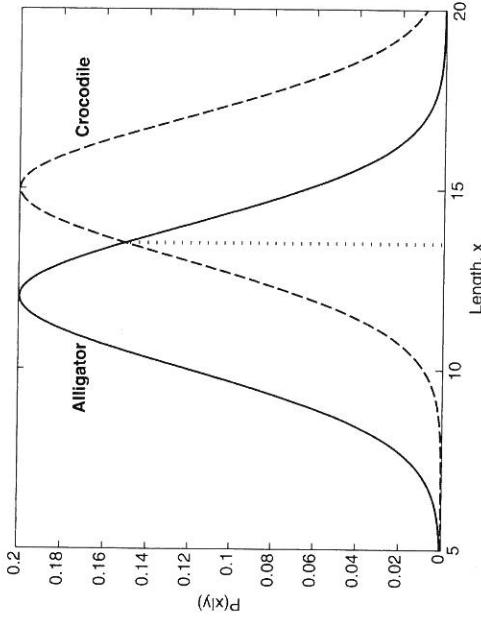


Figure 5.11. Comparing the likelihood functions of a crocodile and an alligator.

that their length x follows a Gaussian distribution with a standard deviation equal to 2 feet, we can express their class-conditional probabilities as follows:

$$P(X|\text{Crocodile}) = \frac{1}{\sqrt{2\pi} \cdot 2} \exp \left[-\frac{1}{2} \left(\frac{X - 15}{2} \right)^2 \right] \quad (5.19)$$

$$P(X|\text{Alligator}) = \frac{1}{\sqrt{2\pi} \cdot 2} \exp \left[-\frac{1}{2} \left(\frac{X - 12}{2} \right)^2 \right] \quad (5.20)$$

Figure 5.11 shows a comparison between the class-conditional probabilities for a crocodile and an alligator. Assuming that their prior probabilities are the same, the ideal decision boundary is located at some length \hat{x} such that

$$P(X = \hat{x}|\text{Crocodile}) = P(X = \hat{x}|\text{Alligator}).$$

Using Equations 5.19 and 5.20, we obtain

$$\left(\frac{\hat{x} - 15}{2} \right)^2 = \left(\frac{\hat{x} - 12}{2} \right)^2,$$

which can be solved to yield $\hat{x} = 13.5$. The decision boundary for this example is located halfway between the two means. ■