The sigmoid function, which looks like an *S* curve, rises from 0 and plateaus at 1. From the sigmoid function shown in Figure 20-4, as $\hat{y}$ increases to positive infinity, the sigmoid output gets closer to 1, and as t decreases toward negative infinity, the sigmoid function outputs 0.

Training the Logistic Regression Model

The logistic regression cost function is formally written as

$$Cost\left(h(t),y\right)=\{-\log\left(h(t)\right) if\ y=1 - \log\left(1-h(t)\right) if\ y=0$$

The cost function also known as ***log-loss*** is set up in this form to output the penalty of the algorithm if the model predicts a wrong class. To give more intuition, take, for example, a plot of $-\ log\,(h(t))$ when $y = 1$ in Figure 20-5.

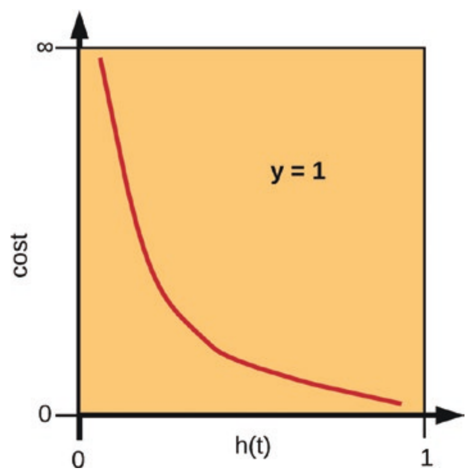

Figure 20-5. *Plot of h(t) when y = 1*

In Figure 20-5, if the algorithm correctly predicts that the target is 1, then the cost tends toward 0. However, if the algorithm $h(t)$ predicts incorrectly the target as 0, then the cost on the model grows exponentially large. The converse is the case with the plot of $-\ log\,(1-h(t))$ when y = 0.

The logistic model is optimized using gradient descent to find the optimal values of the parameter θ that minimizes the cost function to predict the class with the highest probability estimate.

Multi-class Classification/Multinomial Logistic Regression

In multi-class or multinomial logistic regression, the labels of the dataset contain more than 2 classes. The multinomial logistic regression setup (i.e., the cost function and optimization procedure) is structurally similar to logistic regression; the only difference is that the output of logistic regression is 2 classes, while multinomial has greater than 2 classes (see Figure 20-6).

In Figure 20-6, the multi-class logistic regression builds a one-vs.-rest classifier to construct decision boundaries for the different class memberships.

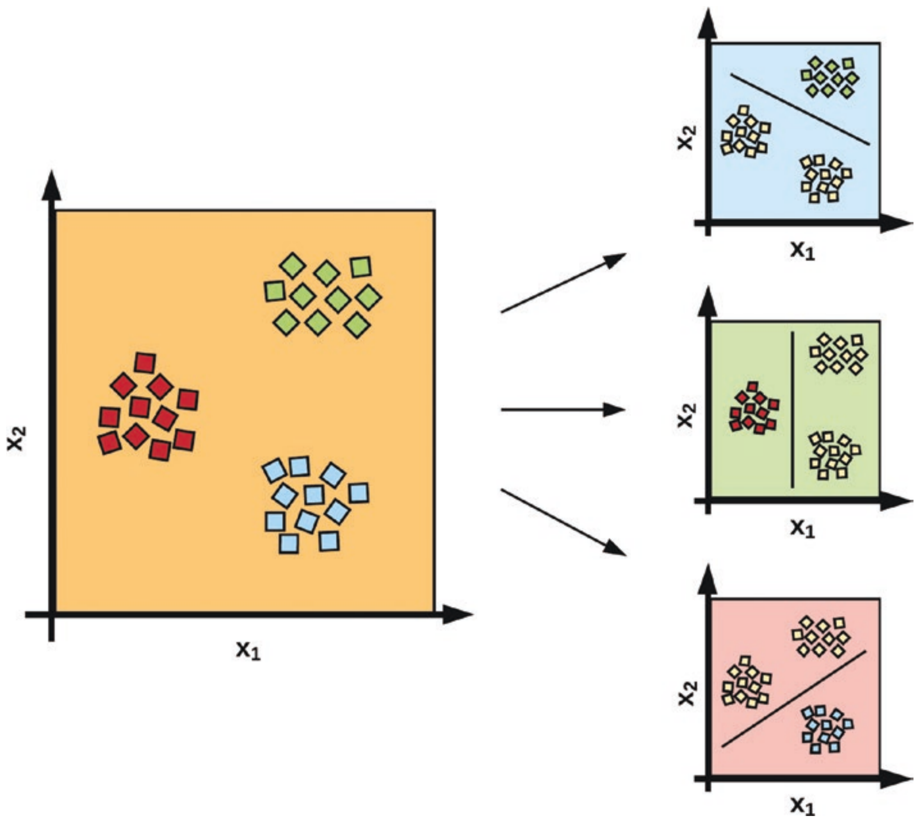

Figure 20-6. *An illustration of multinomial regression*