Introducing the Logit or Sigmoid Model

The logistic function, also known as the logit or the sigmoid function, is responsible for constraining the output of the cost function so that it becomes a probability output between 0 and 1. The sigmoid function is formally written as

$$h(t) = \frac{1}{1+e^{-t}}$$

The logistic regression model is formally similar to the linear regression model except that it is acted upon by the sigmoid model. The following is the formal representation:

$$\hat{y} = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_n x_n$$

$$h(\hat{y}) = \frac{1}{1+e^{-\hat{y}}}$$

where $0 \leq h(t) \leq 1$. The sigmoid function is graphically shown in Figure 20-4.

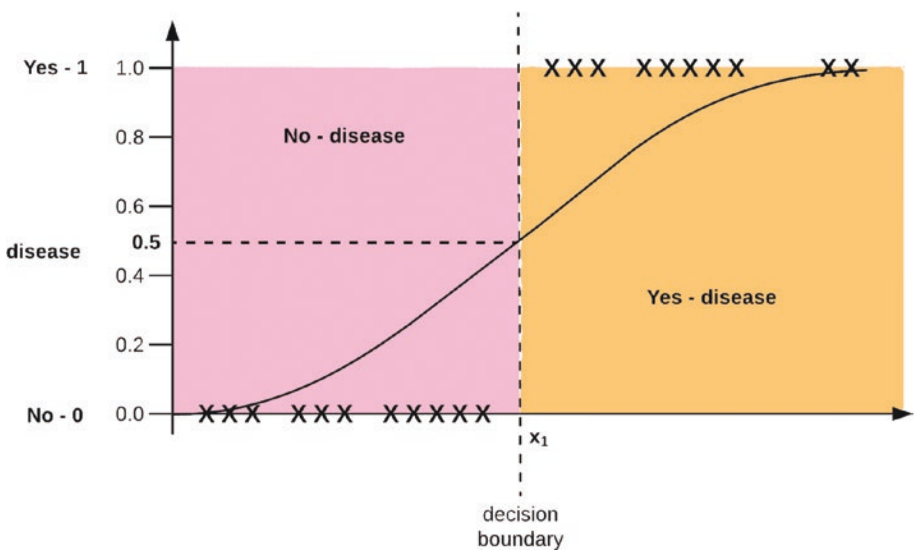

Figure 20-4. *Logistic function*

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.

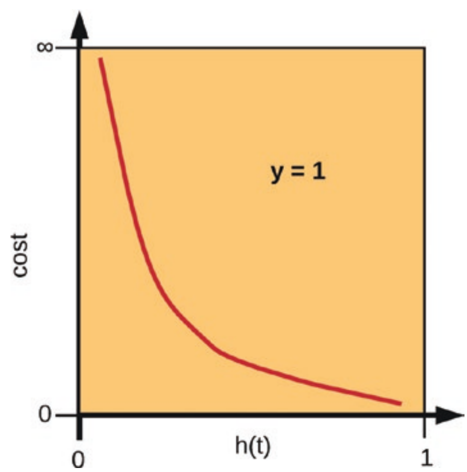

Figure 20-5. *Plot of h(t) when y = 1*

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.