Perceptron

The perceptron algorithm uses a somewhat similar function to logistic regression's sigmoid. The perceptron algorithm uses the unit step function

$$u(z) = \begin{cases} 0 & z < 0 \\ 1 & z \geq 0 \end{cases}$$

and our hypothesis function is defined as $h_\theta(x) = u\left(\theta^T x\right)$. This means that

$$h_\theta(x) = \begin{cases} 0 & \theta^T x < 0 \\ 1 & \theta^T x \geq 0 \end{cases}$$

Like logistic regression, this is used for binary classification. It decides whether or not an input, represented by a vector of numbers, belongs to some specific class. If we look at our hypothesis function, based on an input x, it outputs the hypothesized classification based on $\theta^T x$. The learning algorithm is laid out such that

$$\theta_j = \theta_j + \alpha\left(y^{(i)} - h_\theta\left(x^{(i)}\right)\right) x_j^{(i)}$$

where $y^{(i)}$ represents the classification. If the hypothesis is correct, then $y^{(i)} - h_\theta\left(x^{(i)}\right) = 0$. Otherwise, the result will be -1 or 1.