Logistic Regression: Binary Classification

Binary Classification, as it's name suggests, solves classification problems with only two classes. For example, trying to determine if email *is spam* or *not spam*. Below describes this more formally using a mathematic expression which says y is an *element* of 0 or 1:

> $y \in \{0, 1\}$

Given the email example, $y = 0$ would mean *no spam* where as $y = 1$ would mean the email *is spam*. But how to determine if y is 0 or 1? By using a *threshold* classifier such that our hypothesis function $h_\theta(x)$ predicts a value between 0 and 1. Such that:

> $0 \leq h_\theta(x) \leq 1$

We predict a y value of 0 or 1 by using the *threshold* of 0.50 which is halfway between 0 and 1:

> If $h_\theta(x) \geq 0.50$; Then predict $y = 1$
>
> If $h_\theta(x) < 0.50$; Then predict $y = 0$

To get our hypothesis function to scale and make predictions in this range we will need to use the *Sigmoid Function*, also called the *Logistic Function*. Recall the hypothesis function for liner regression:

> $h_\theta(x) = \theta^T x$

The *Sigmoid Function* (S) takes our hypothesis (z) as a parameter and returns a value between 0 and 1. For example:

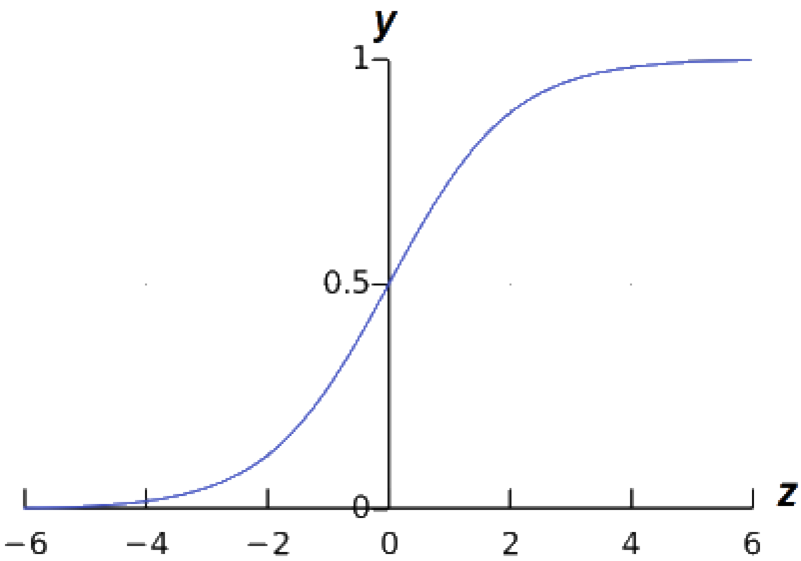
> $z = h_\theta(x) \therefore S(z) = \frac{1}{1+e^{-z}}$

Using the *Sigmoid Function*, our hypothesis changes to the following:

> $h_\theta(x) = S(\theta^T x)$

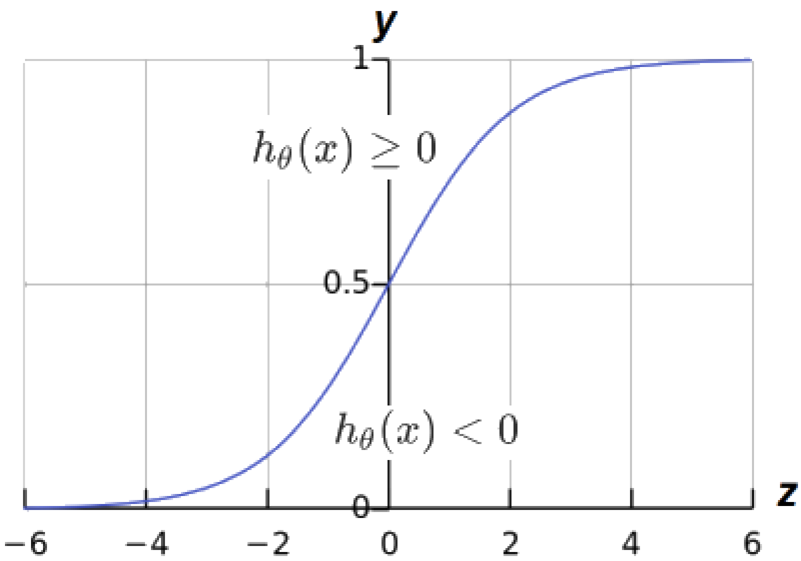
This will effectively scale our hypothesis prediction (z) to a value between 0 and 1. Our hypothesis prediction (z) controls the steepness of the Sigmoid curve. And the numerator, 1 in this case, is the curves maximum value. The irrational number e is also known as *Euler's number*. It is approximately 2.718281, and is the base of the natural logarithm.

Below shows a plot of the *Sigmoid Function* that shows the *threshold* at 0.5 which asymptotes (continually approaches but does not meet) at 1 and 0 with the z axis on the horizontal:



It it worth noting that the prediction returned by the hypothesis function directly corresponds to the *Sigmoid* value, such that:

> $h_\theta(x) \geq 0.5$ when $\theta^T x \geq 0$
>
> $h_\theta(x) < 0.5$ when $\theta^T x < 0$



In interpreting the value returned by the *Sigmoid* function, if it returns $y = .70$ when we say there is a 70% chance that $y = 1$ and therefore, by deduction, a 30% chance that $y = 0$. In a more concrete example, if malignant tumor features lie in the range *above* 0.5:

> $\vec{x} = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 1 \\ tumorSize \end{bmatrix}$

If $h_\theta(x) = 0.7$ then we say there is a 70% chance the tumor might be malignant. Therefore, since $.7 > .5$, we say that $y = 1$. More formally, we write this in mathematical terms where we say, the *probability* that $y = 1$, given feature x, parameterized by θ:

> $h_\theta(x) = P(y = 1|x; \theta)$

Therefore, the probability of $y = 0$ plus the probability of $y = 1$ will always be equal to 1:

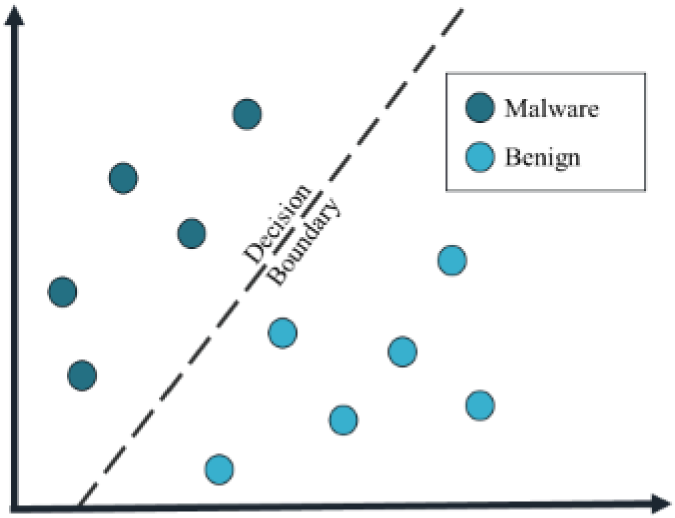
> $P(y = 0|x; \theta) + P(y = 1|x; \theta) = 1$

Likewise, we can find the probability of y being 0 by subtracting the $y = 1$ probability from 1:

> $(y = 0|x; \theta) = 1 - P(y = 1|x; \theta)$

Decision Boundary

The *decision boundary* divides the plotted features from a data set. This boundary is used to determine regions that determine when $y = 0$ or $y = 1$. The decision boundary can be any equation, linear or non-linear, that best segments the data. Below will outline how to find this boundary using the *Sigmoid Function*.



For example, if we have the following hypothesis function:

> $h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$

And our parameters are:

> $$\theta = \begin{bmatrix} -3 \\ 1 \\ 1 \end{bmatrix}$$

Then, when implemented using the *Sigmoid Function* becomes:

> $h_\theta(x) = S(-3 + 1x_1 + 1x_2)$

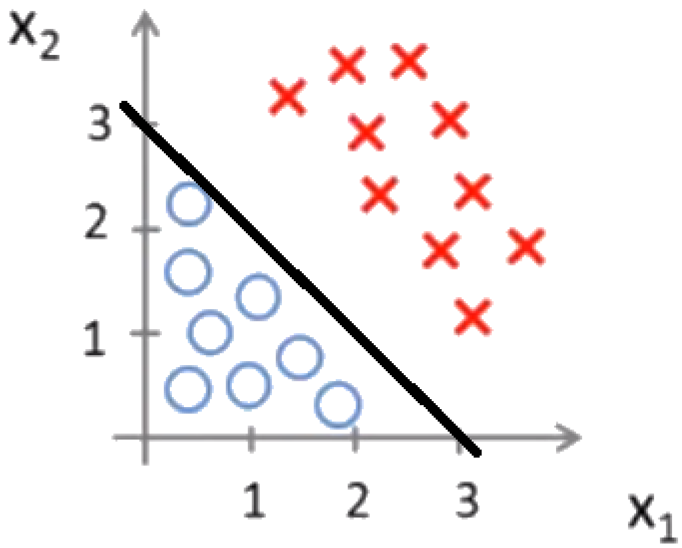
Therefore:

> $y = 1$ if $-3 + 1x_1 + 1x_2 \geq 3$
>
> $y = 0$ if $-3 + 1x_1 + 1x_2 < 3$

Or, rewritten as below where 3, the y-intercept, denotes our *decision boundary*:

> $-3 + x_1 + x_2 \geq 0 \therefore x_1 + x_2 \geq 3$

Remember, that θ_0 is our bias weight with a value of 1 since there are no features associated with it. So, we are basically saying that *our features lie beyond the boundary where x_1 or x_2 is 3*. Plotting the line $x_1 + x_2 = 3$ results in the decision boundary graphed below where both the x_1 axis and x_2 axis are intercepted at 3. Therefore, any value on the right side of the decision boundary, where $x_1 + x_2 \geq 3$, yields a y value of 1. Conversely, $x_1 + x_2 < 3$ will result in a y value of 0.



It is important to note that the equation $x_1 + x_2 = 3$ corresponds to region where $h_\theta(x) = 0.5$. Also, the decision boundary, is a property of the hypothesis and not the data set. In other words, removing all features from the graph does not impact the decision boundary or the hypothesis from determining the predicted value for y.

Logistic Regression Cost Function

Below is the cost function from linear regression. Note the $\frac{1}{2}$ has been moved inside the summation to make things easier to understand for this topic.

> $J(\theta) = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{2}(h_\theta(x)^{(i)} - y^{(1)})^2$

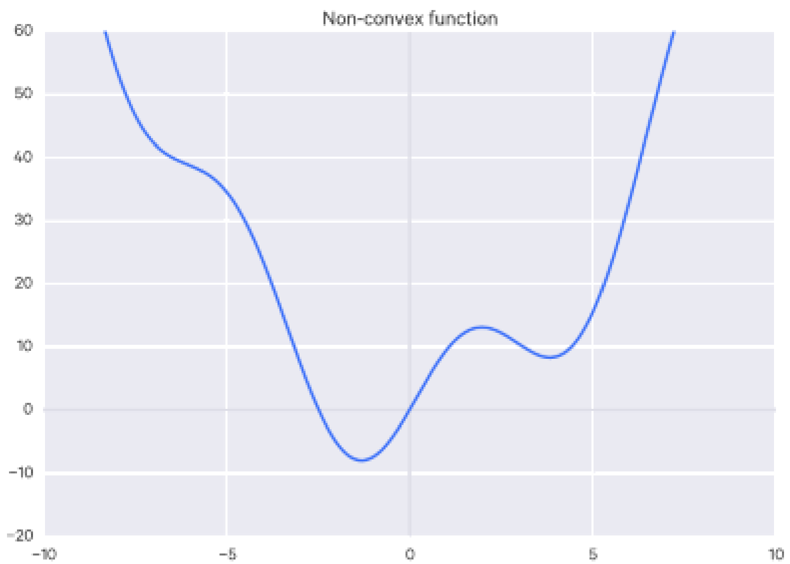
Here, the *cost* is:

> $\frac{1}{2}(h_\theta(x)^{(1)} - y^{(i)})^2$

Therefore, for reasons described later, we can define our cost function as:

> $Cost(h_\theta(x), y) = \frac{1}{2}(h_\theta(x) - y)^2$

When this function is graphed it results in a *non-convex* graph with multiple local minimums:



It is non-liner because of the *Sigmoid Function* $\frac{1}{1+e^{-z}}$. When using *Gradient Decent* using this cost function you are not guaranteed to converge at the *global* minimum. What's needed is a *convex* cost function with a single global minimum. For this to happen, our cost function has to change:

> $$Cost(h_\theta(x), y) = \begin{Bmatrix} -log(h_\theta(x)) & \text{if} \quad y = 1 \\ \text{-}log(1 - h_\theta(x)) & \text{if} \quad y = 0 \end{Bmatrix}$$

Instead of using two conditions, the cost function for a specific feature can be simplified into a single equation which satisfies the above equation:

> $$Cost(h_\theta(x), y) = -y \cdot log(h_\theta(x)) - (1 - y) \cdot (log(1 - h_\theta(x)))$$

In the equation there are two smaller equations, $-y \cdot log(h_\theta(x))$ and $-(1 - y) \cdot (log(1 - h_\theta(x)))$.

- When $y = 1$, the value for the second equation becomes 0 leaving $-y \cdot log(h_\theta(x))$
- When $y = 0$, the value for the first equation becomes 0 leaving $-(1 - y) \cdot (log(1 - h_\theta(x)))$

Taking the Cost function for a feature, we can will implement our cost function across all features using a summation:

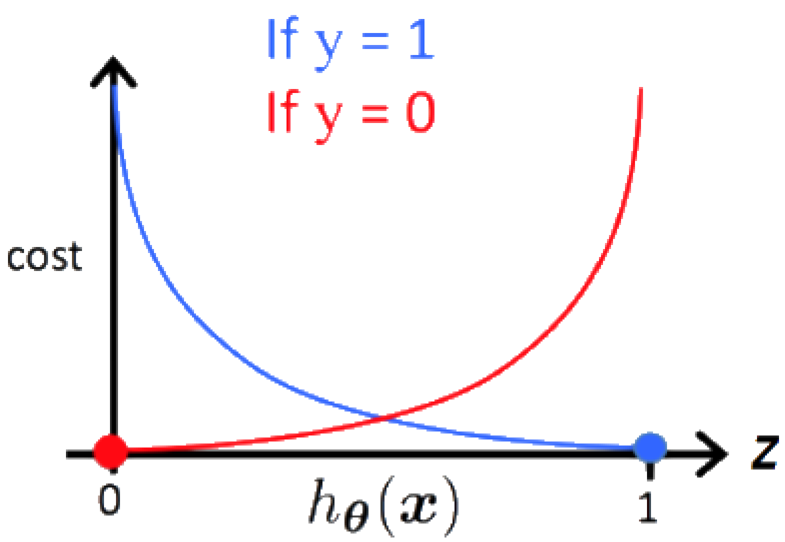
> $$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} Cost(h_\theta(x^{(i)}, y^{(i)})$$

Expanded is:

> $$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} (-y^{(i)} \cdot log(h_\theta(x^{(i)})) - (1 - y^{(i)}) \cdot (log(1 - h_\theta(x^{(i)}))))$$

Note: This function can be derived using the *Principal Maximum Likelihood Estimation*.

When we plot this cost function we get a graph like below. The log functions where $y = 0$ and $y = 1$ actually continue on beyond the z axis but we are only concerned with the range between 0 and 1.



Therefore, for the two *Cost* function parameters $h_\theta(x)$ and y:

> $$\text{Cost} = 0 \text{ if } y = 1 \text{ and } h_\theta(x) = 1$$

Likewise as $h_\theta(x)$ approaches 0, the cost will approach infinity since it asymptotes:

> $$\text{as } h_\theta(x) \to 0 \text{ then } Cost \to \infty$$

The graph above captures the intuition that if $h_\theta(x) = 0$ (Predict $P(y = 1|x; \theta) = 0$), *but* we found out that $y = 1$ in actuality, then we penalize the learning algorithm by a very large cost where $Cost \to \infty$

Gradient Descent

The equation for Gradient Descent for Logistic Regression is the same equation with the exception of the hypothesis with implements the *Sigmoid Function*.

> $\theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta \left(x^{(i)}\right) - y^{(i)}\right) \cdot x_0^{(i)}$ where $h_\theta(x) = \frac{1}{1+e^{-(\theta^T x)}}$