Logistic Regression is a classification problem

For example,

Email : Spam / not ?
Online Transaction : Fraud (Yes/No)
Tumor : Malignant / Benign ?

In all of these examples, the prediction variable is y and

Binary Classification Problem ← $y \in \{0, 1\}$ s.t
0: "Negative Class"
1: "Positive Class"

If $y \in \{0, 1, 2, 3, 4\}$, then the problem would be called multi classification.

Hypothesis Representation

We want $0 \le h_\theta(x) \le 1$, where $h_\theta(x)$ is the hypothesis predictor.

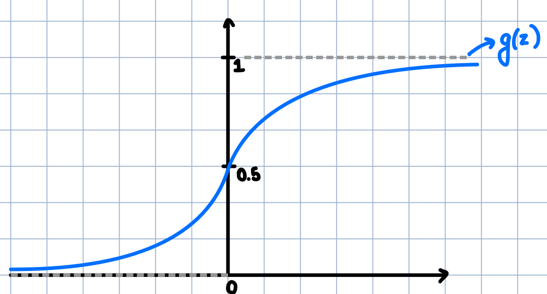
For Linear Regression, $h_\theta(x) = \theta^T x$.

We will modify the above hypothesis as : $g(\theta^T x)$ where

$$g(z) = \frac{1}{1+e^{-z}}, \quad z \in \mathbb{R}.$$

This is called sigmoid function / Logistic fn.

$$\left.\begin{aligned} h_\theta(x) &= \theta^T x \\ g(z) &= \frac{1}{1+e^{-z}} \end{aligned}\right\} \quad h_\theta(x) = \frac{1}{1+e^{-\theta^T x}}$$



Sigmod Function

Interpretation of Hypothesis Output

When $h_\theta(x)$ outputs some number, we are going to treat that number as the estimated prob. that $y=1$ on a new input x.

$h_\theta(x) =$ est. prob. that $y=1$ on input x.

Example:

$$\text{Let } x = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 1 \\ \text{Tumor Size} \end{bmatrix}$$

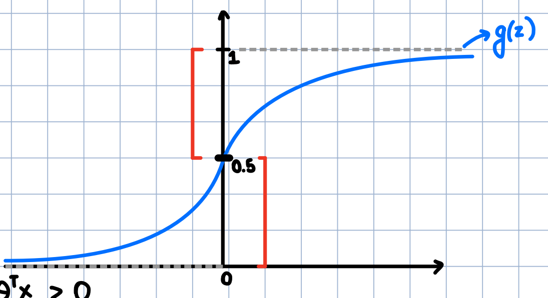
$$\text{and } h_\theta(x) = 0.7$$

Tell patient that 70% chance of tumor being malignant.

Statistically, $h_\theta(x) = P(y=1 \mid x; \theta)$ parameterized by θ.

So, $P(y=0 \mid x;\theta) = 1 - P(y=1 \mid x; \theta)$

Decision Boundary



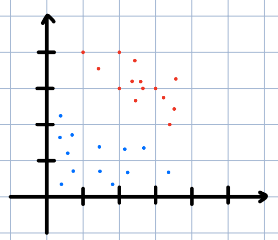
Sigmod Function

From figure, we can see

$$g(z) \geq 0.5 \quad \text{when } z \geq 0$$

Thus $h_\theta(x) = g(\theta^T x) \geq 0.5$ when $\underbrace{\theta^T x}_{z} \geq 0$

Ex 1.



Let our training data look like in the left figure.

Here, hypothesis, $h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$

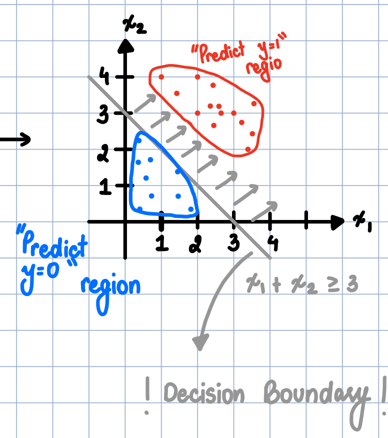
Suppose, we choose $\theta_0 = -3$, $\theta_2 = 1$, $\theta_3 = 1$.

$$\text{Then } \theta = \begin{bmatrix} -3 \\ 1 \\ 1 \end{bmatrix}.$$

Given this choice of Hypothesis parameters, let's figure where $h_\theta(x)$ will predict $y=1$ & 0.

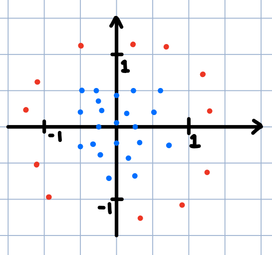
We know for "Predict $y=1$", $\underbrace{-3 + x_1 + x_2}_{\theta^T x} \geq 0$

$\Rightarrow x_1 + x_2 \geq 3$ $\xrightarrow{\text{Plot}}$



* Decision Boundary is a property of the hypothesis & it's parameter $\theta_1, \theta_2, \theta_3$, not of the dataset. If we neglect the dataset nothing changes. *

Ex.2 Nonlinear Decision Boundaries



Let hypothesis,

$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2)$

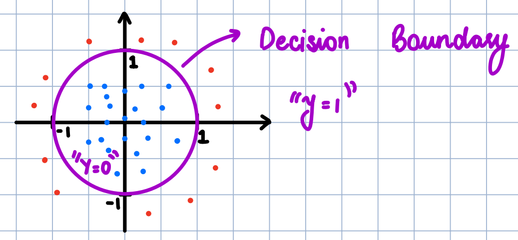
We don't know how to choose θ_i, so let's suppose they are given to us.

$$\theta = \begin{bmatrix} -1 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

This means, predict "$y=1$" if $-1 + x_1^2 + x_2^2 \geq 0$

$\Rightarrow x_1^2 + x_2^2 \geq 1$

Plotting $x_1^2 + x_2^2 \geq 1$.



With complex dataset, complex hypothesis parameters, we will different and more complex plots. (ellipse, or weird shapes)

Cost function

Que : How to fit the parameter θ for logistic Regression ?

Let's define optimization objective of the COST FUNCTION that we will use to fit the parameters.

Given :

Training set : $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots\ldots, (x^{(m)}, y^{(m)})\}$

Each $x \in \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}_{(n+1)\times 1}$ $x_0 = 1$, $y \in \{0, 1\}$ [Y - training set]

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$

To find :

How to choose parameter θ ?

Analysis : (using linear reg cost fn as basis)

Back to linear Reg, we had the cost fn as :-

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

Let $\quad \mathrm{Cost}\left(h_\theta(x^{(i)}), y^{(i)} \right) = \frac{1}{2} \left(h_\theta(x^{(i)}) - y^{(i)} \right)^2$

Then

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} \text{Cost}\left(h_\theta(x^{(i)}),\ y^{(i)}\right)^2$$

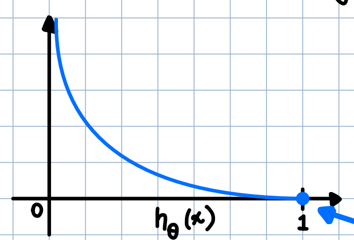
$$= \frac{1}{m} \sum_{i=1}^{m} \text{Cost}\left(h_\theta(x),\ y\right)^2$$

Because of sigmod function $(h_\theta(x))$, the above $J(\theta)$ fn for logistic reg. becomes non - convex.

Cost fn of logistic regression :-

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

Plotting Cost fn for case $y=1$:

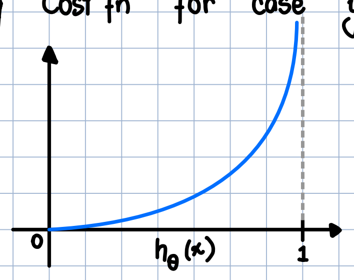


Some properties of the plot:

(*) Cost = 0 if $y=1$, $h_\theta(x) = 1$
But as $h_\theta(x) \to 0$
Cost $\to \infty$

Captures intuition that if $h_\theta(x) = 0$ (Predict $P(y=1|x;\theta)$), but $y=1$ we will penalize learning algorithm by a very large cost.

Plotting Cost fn for case $y=0$:



Some properties of the plot:

(*) $Cost = \infty$ if $y=0$, $h_\theta(x) = 1$]
But as $h_\theta(x) \to 1$
$Cost \to \infty$

Captures intuition that if $h_\theta(x) = 1$ (Predict $P(y=0|x;\theta)$), but $y=0$ we will penalize learning algorithm by a very large cost.

Simplified Cost function and Gradient Descent

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

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

equivalent

Note: $y=0$ or 1 always

Rewriting Cost function,

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

Proof (the two equations are equal):

There are only possibilities of y, 0 or 1.

Let $y=1$. Then

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

Let $y=0$. Then

$$Cost(h_\theta(x), 0) = -\log(1-h_\theta(x))$$

Simple!

$$\therefore J(\theta) = \frac{1}{m}\sum_{i=1}^{m}\left[-y^{(i)}\log(h_\theta(x^{(i)})) - (1-y^{(i)})\log(1-h_\theta(x^{(i)}))\right]$$

In order to fit parameters:

$$\text{minimize } J(\theta) = \min_{\theta} J(\theta)$$

We will use Gradient Descent to minimize $J(\theta)$.

Repeat {

$$\theta_j := \theta_j - \alpha \frac{\partial J(\theta)}{\partial \theta_j}$$

} simultaneously updates all θ_j.

Computing $\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) , y^{(i)} \right) x_j^{(i)}$

Subsitute back in Gradient Descent,

$$\theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right) x_j^{(i)}$$

This looks exactly identical to linear regression!

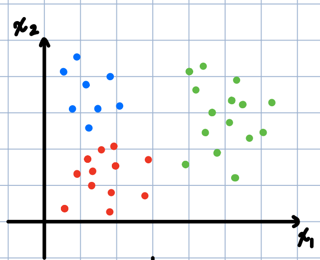
The difference is for linear regression, $h_\theta(x) = \theta^T x$

and for logistic regression, $h_\theta(x) = \frac{1}{1+e^{-\theta^T x}}$.

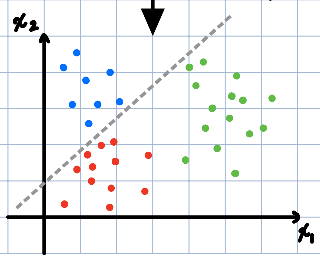
Multiclass Classification: One v/s all

Multiclass classification example:

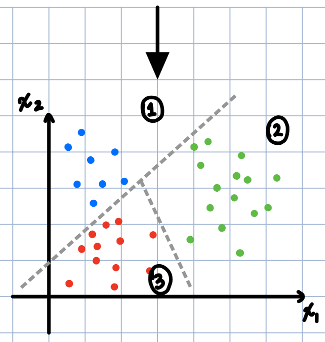
Email foldering/tagging: Work ($y=1$), home ($y=2$), Friends ($y=3$), Hobby ($y=4$)



Take 2 categories as 1 and apply algorithm once



$h_\theta^{(1)}(x)$



$h_\theta^{(2)}(x)$

One v/s All

(*) Train a logistic regression classifier $h_\theta^{(i)}(x)$ for each class i to predict the probability that $y = i$.

(**) On a new input x, to make a prediction, pick the class i that maximizes $\max_i h_\theta^{(i)}(x)$.