Logistic Regression

⇒ It is an algorithm for performing binary Classification.

Classification

$\boxed{\text{Input}}$ $\boxed{\text{Output}}$ → In binary classification, Output can only take two values. $\{0, 1\}$.

⇒ Set of Labeled Input or Input Output pair

⇒ Objective: Learn the mapping between input & Output.

(i.e. Given Input we want to predict the Output)

⇒ Let Input be represented by a set of n input feature. $(X_1, X_2 \cdots X_n)$ where $X_i \in R$ $i = \{1, \cdots n\}$

⇒ Let Output be represented by Y where $Y \in \{0, 1\}$

⇒ Let us Consider a training set of m training example:

$$\{(X^{(i)}, Y^{(i)}) \mid X^{(i)} \in R^n, Y^{(i)} \in \{0, 1\}, i \in \{1, \cdots m\}\}$$

⇒ In Logistic regression, we are looking for a linear decision boundary that Separates the two lables.

Let $\theta_0 + \theta_1 X_1 + \theta_2 X_2 + \cdots + \theta_n X_n = 0$ the linear decision boundary parametrized by $(\theta_0, \theta_1 \cdots \theta_n)$.

⇒ For simplifying the notation let $x_0 = 1$ be a dummy feature. So decision boundary becomes

$$\Rightarrow \theta_0 x_0 + \theta_1 x_1 + \dots + \theta_n x_n = 0$$

$$\Rightarrow \boxed{\theta^T x = 0}, \text{ where } \theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{bmatrix} \text{ and } x = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_n \end{bmatrix}$$

⇒ x for which $\theta^T x \geq 0$ we will classify it as $y = 1$.

⇒ and for which $\theta^T x < 0$ we will classify it as $y = 0$.

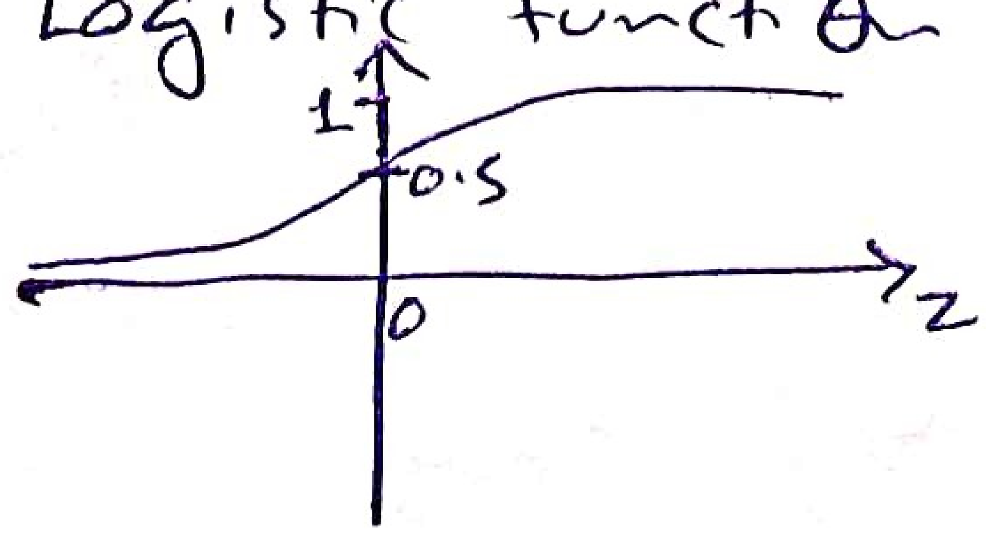
⇒ Given a decision boundary parametized by θ, we ~~are looking~~ want to assign probability to all x.
(i.e. If the input is x, what is the probability that corresponding $Y = 1$)

⇒ So we are looking for a function $h_\theta(x)$ such that,

- $0 \leq h_\theta(x) \leq 1 \quad \forall x \in \mathbb{R}^{n+1}$
- $h_\theta(x) = 0.5 \quad \forall \theta^T x = 0$
- $h_\theta(x) > 0.5 \quad \forall \theta^T x > 0 \quad (h_\theta(x) \uparrow \text{ as } \theta^T x \uparrow)$
- $h_\theta(x) < 0.5 \quad \forall \theta^T x < 0 \quad (h_\theta(x) \downarrow \text{ as } \theta^T x \downarrow)$

⇒ One such function is Logistic function:

$$\boxed{g(z) = \frac{1}{1 + e^{-z}}}$$



⇒ So let $h_\theta(x) = g(\theta^T x) = \dfrac{1}{1 + e^{-\theta^T x}}$

$\Rightarrow$ For estimating θ given the training set, we can maximize the likelihood of θ:

$$\mathcal{L}(\theta) = P(y^{(1)} \ldots y^{(2)} \ldots y^{(m)} \mid x^{(1)} \ldots x^{(m)}; \theta)$$

$$\mathcal{L}(\theta) = \prod_{i=1}^{m} P(y^{(i)} \mid x^{(i)}; \theta)$$

{Assuming all the data in the training set are independent}

$\Rightarrow$ Let $l(\theta) = \log \mathcal{L}(\theta)$ be the log likelihood. Maximizing $\mathcal{L}(\theta)$ is equivalent to maximizing $l(\theta)$ as $\log(x)$ is a monotonic increasing function of x.

$$l(\theta) = \sum_{i=1}^{m} \log P(y^{(i)} \mid x^{(i)}; \theta)$$

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

$$\boxed{l(\theta) = \sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)}))}$$

$\Rightarrow$ To maximize $l(\theta)$ we can use gradient ascent.

$$\boxed{\theta := \theta + \alpha \nabla_\theta l(\theta)}$$

Learning rate

* Calculating the gradient ($\nabla_\theta l(\theta)$)

$$\nabla_\theta l(\theta) = \sum_{i=1}^{m} \left(y^{(i)} \nabla_\theta \log h_\theta(x^{(i)}) \right.$$

$$\left. (1-y^{(i)}) \nabla_\theta \log(1 - h_\theta(x^{(i)})) \right)$$

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

$$\frac{1}{h_\theta(x^{(i)})} * \nabla_\theta h_\theta(x^{(i)})$$

$$\left(1 + e^{-\theta^T x^{(i)}}\right) \times \frac{\theta e^{-\theta^T x^{(i)}}}{(1+e^{-\theta^T x^{(i)}})^2}$$

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

$$\frac{1}{1-h_\theta(x^{(i)})} \times -\nabla_\theta h_\theta(x^{(i)})$$

$$\frac{1}{e^{-\theta^T x^{(i)}} / 1+e^{-\theta^T x^{(i)}}} * \frac{-\theta e^{-\theta^T x^{(i)}}}{(1+e^{-\theta^T x^{(i)}})^2}$$

$$\frac{\cancel{1+e^{-\theta^T x^{(i)}}}}{\cancel{e^{-\theta^T x^{(i)}}}} * \frac{-\theta \cancel{e^{-\theta^T x^{(i)}}}}{(1+e^{-\theta^T x^{(i)}})^{\cancel{2}}}$$

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

$$\nabla_\theta l(\theta) = \sum_{i=1}^{m} \frac{\theta}{1+e^{-\theta^T x^{(i)}}}\left(y^{(i)} e^{-\theta^T x^{(i)}} - 1 + y^{(i)}\right)$$

$$\downarrow$$

$$\frac{\theta}{1+e^{-\theta^T x^{(i)}}}\left(y^{(i)}\left(1+e^{-\theta^T x^{(i)}}\right) - 1\right)$$

$$\nabla_\theta l(\theta) = \sum_{i=1}^{m}\left(y^{(i)} - \frac{1}{1+e^{-\theta^T x^{(i)}}}\right)\theta$$

$$\boxed{\nabla_\theta l(\theta) = \sum_{i=1}^{m}\left(y^{(i)} - h_\theta(x^{(i)})\right)\theta}$$

$\Rightarrow$ So update rule for gradient ascent is:

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