Logistic Regression

To solve classification problem → Binary classification ←
→ Multiclass classification

Dataset

Independent feature ↓ | Dependent or o/p feature ↓

No.of play hours	Pass/Fail {y}	
9	Fail	0
8	Fail	0
7	Fail	0
6	Fail	0
5	Pass	1
4	Pass	1
1	Pass	1
2	Fail	0

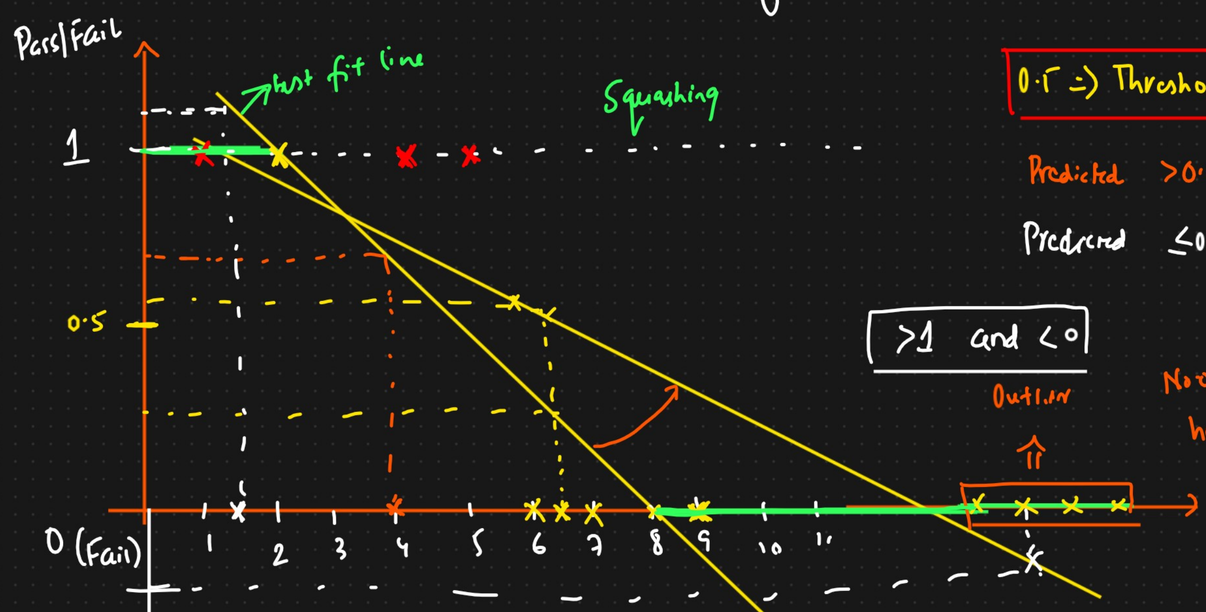
$\hat{y}$ → predicted

TRAIN ↓

No.of play hours → Model → Pass/Fail

↓ Accuracy ↑

Can we solve this classification problem using Regression?



$\boxed{0.5 \Rightarrow \text{Threshold value}}$

Predicted $> 0.5 \Rightarrow$ Pass

Predicted $\leq 0.5 \rightarrow$ Fail

$\boxed{>1 \text{ and } <0}$

Outlier

⇑

No.of play hours

Why we cannot use Linear Regression for Classification

① Best fit line changes because of outliers → prediction goes wrong

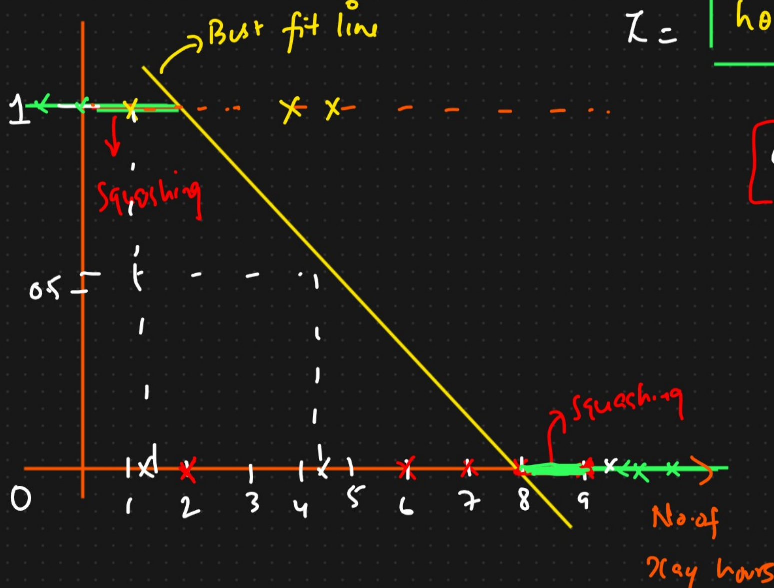
② The outcome comes >1 and <0 also

To solve this problem we use · Logistic Regression

$\Downarrow$

$\boxed{0 \text{ to } 1}$ ⟹ Squashing Technique

How Logistic Regression Solves Classification Problem



$z = \boxed{h_\theta(x) = \theta_0 + \theta_1 x_1}$ → Best fit line

$\Downarrow$

[Sigmoid Activation function]

$\Downarrow$

0 to 1

$$\boxed{\sigma = \frac{1}{1+e^{-z}}} \Rightarrow 0 \text{ to } 1$$

$$h_\theta(x) = \sigma(\underset{\Uparrow z}{\theta_0 + \theta_1 x_1})$$

$$\boxed{h_\theta(x) = \frac{1}{1+e^{-z}}} \Rightarrow z = \theta_0 + \theta_1 x_1 \Rightarrow \boxed{h_\theta(x) = \frac{1}{1+e^{-(\theta_0+\theta_1 x_1)}}}$$

↳ Logistic Regression

Linear Regression Cost function

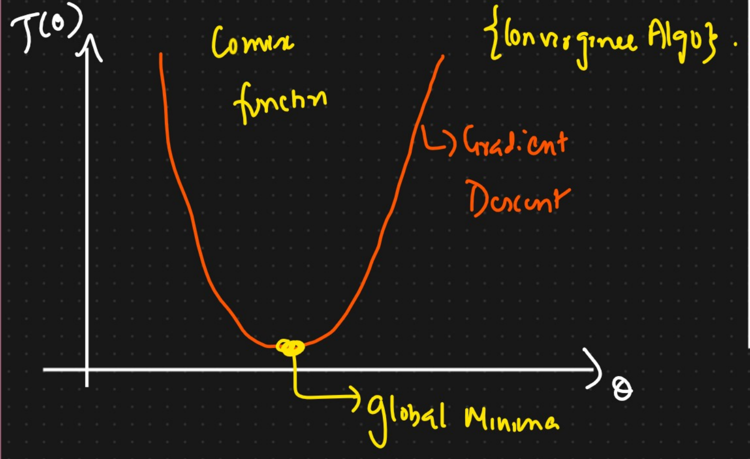
$$J(\theta_0, \theta_1) = \frac{1}{n} \sum_{i=1}^{n} (y_i - h_\theta(x)_i)^2$$

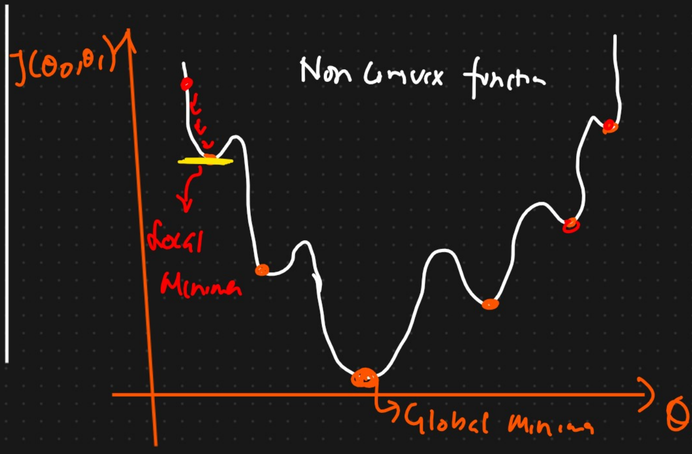
$$h_\theta(x) = \theta_0 + \theta_1 x_1$$

Logistic Regression Cost function

$$J(\theta_0, \theta_1) = \frac{1}{n} \sum_{i=1}^{n} (y_i - h_\theta(x)_i)^2$$

$$h_\theta(x) = \frac{1}{1+e^{-z}} = \frac{1}{1+e^{-(\theta_0+\theta_1 x_1)}}$$





Log Loss

$$J(\theta_0,\theta_1) = \begin{cases} -\log(h_\theta(x)) & \text{if } y=1 \\ -\log(1-h_\theta(x)) & \text{if } y=0 \end{cases}$$

$$h_\theta(x) = \frac{1}{1+e^{-z}} = \frac{1}{1+e^{-(\theta_0+\theta_1 x_1)}}$$

$$\boxed{J(\theta_0,\theta_1) = -y\log(h_\theta(x)) - (1-y)\log(1-h_\theta(x)).} \Rightarrow \text{Convex function}$$

if $y=1$

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

if $y=0 \Rightarrow -\log(1-h_\theta(x))$

Final Aim :

Minimize Cost function $J(\theta_0,\theta_1)$ by changing θ_0 & θ_1

Convergence Algorithm

Repeat

$j = 0, 1$

{

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta} J(\theta_0, \theta_1)$$

}