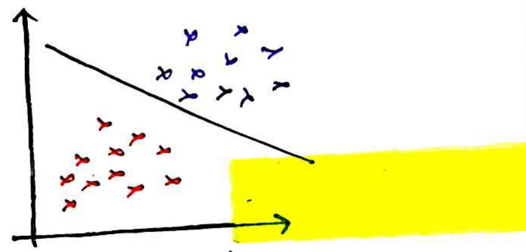
Logistic Regression

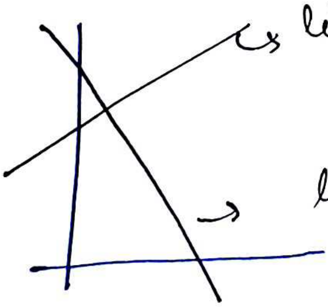
Logistic Regression → Interview
→ Deep learning

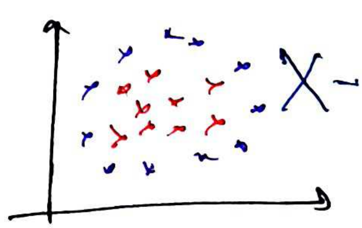
- Classification → categorical → binary 1, 0
  → multi class
- Linear model
  - Linear Regression
  - Logistic Regression → Lines/Planes
    → linear/sort of linear
- multiple Interpretation

cgpa	iq	Placement
-	-	0
-	-	1



gpa | iq | Placement (Y/N)
↓ ↓ ↓





$$\boxed{y = mx + b}$$

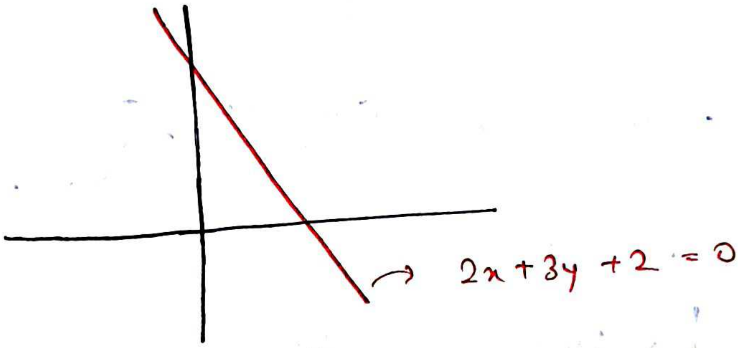
↓

$$\boxed{Ax + By + C = 0}$$ → general eqn of line

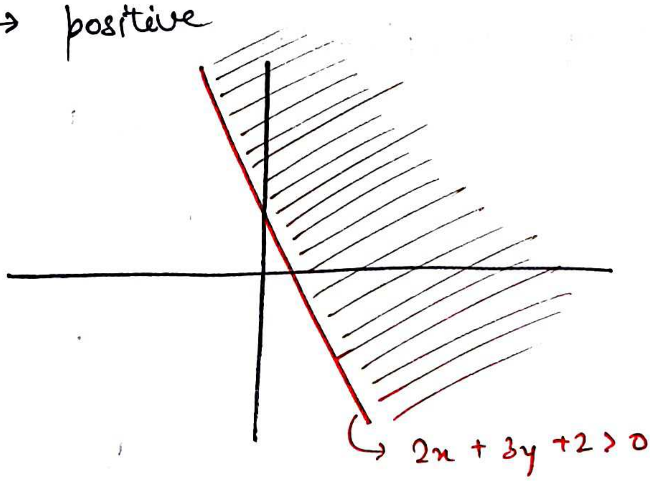
Some basic Geometry

1. Every line has a positive and negative side.

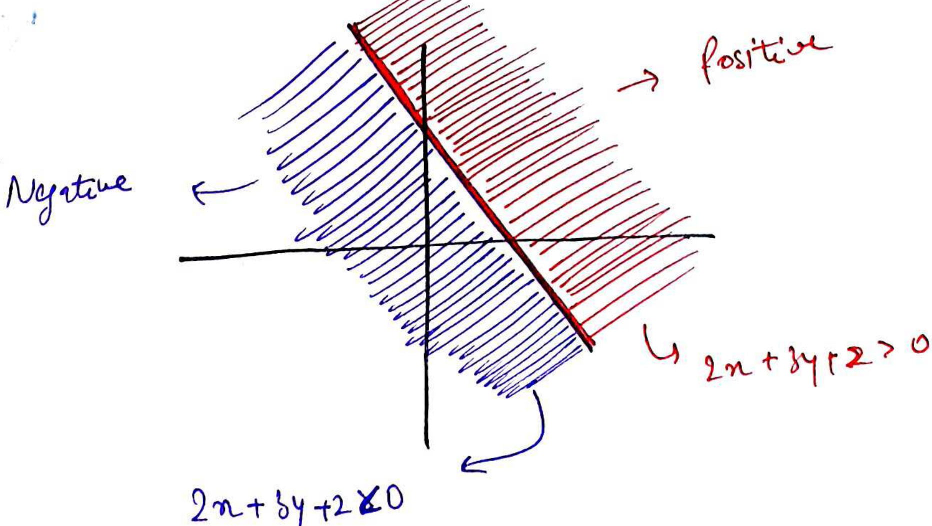
Let assume $2x + 3y + 2 = 0$ is a line



(i) $2x + 3y + 2 > 0$ → positive



(ii) $2x + 3y + 2 < 0$ → negative



2. How to find out if a given point lies on a given line?

Let assume line $4x + 3y + 5 = 0$

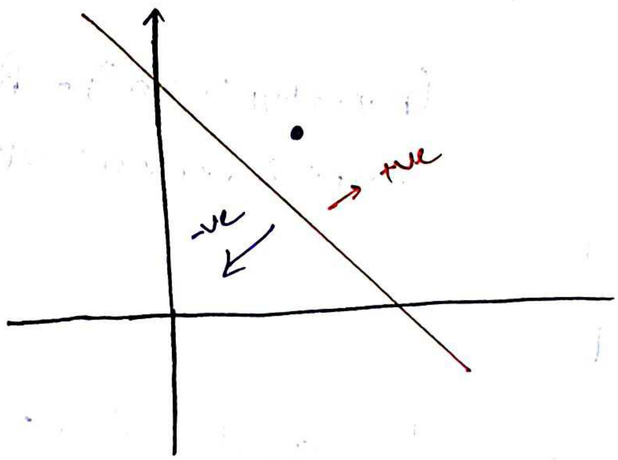
given points $(5, 2)$
x ↵ ↳ y

→ Put given points in assumed line

$\underline{4(5) + 3(2) + 5} = 0$

↳ If this eqn is 0 after solving, then points lies on line. Otherwise not lies.

3. How to find out if a given point is on the positive side of the line or the negative side of the line.

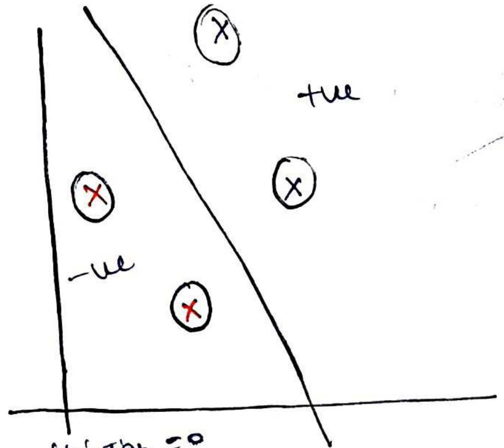


→ If $Ax_1 + by_1 + C > 0 \Rightarrow$ positive region

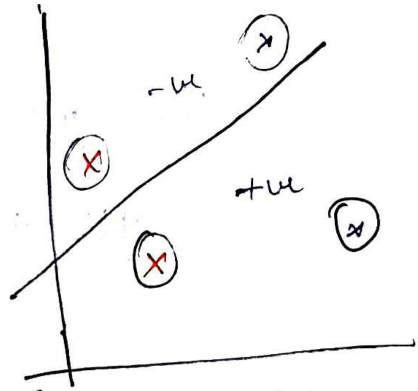
→ If $Ax_1 + by_1 + C < 0 \Rightarrow$ Negitive region

The problem

In classification problem we have to make sure the +ve point land in the +ve region and -ve point lands in the -ve region.



↳ miss clasiffication = 0
model 1
↳ correctly classification



model 2
↳ miss clasification = 2

miss classification → loop (4)

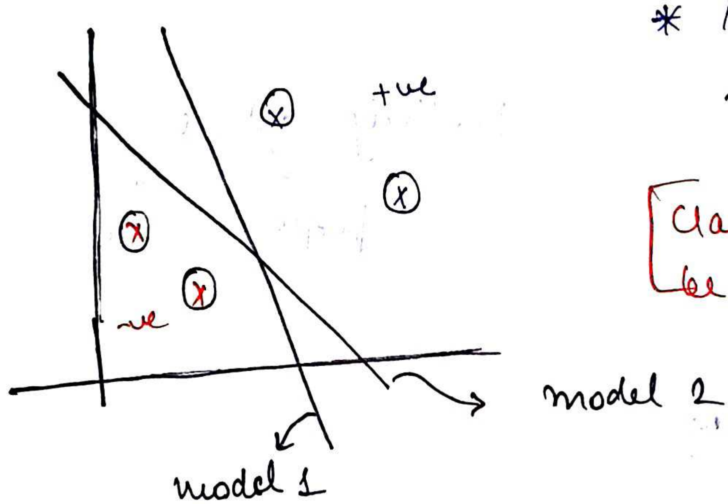
($Ax_1 + By_1 + c > 0$) → Red (negative)
($Ax_1 + By_1 + c < 0$) → Purple (Positive)

model 2
- counter = 0
- loop

if pt → red and $Ax_1 + By_1 + c > 0$
counter + 1 = 1

if pt → green and $Ax_1 + By_1 + c < 0$
counter + 1 = 2

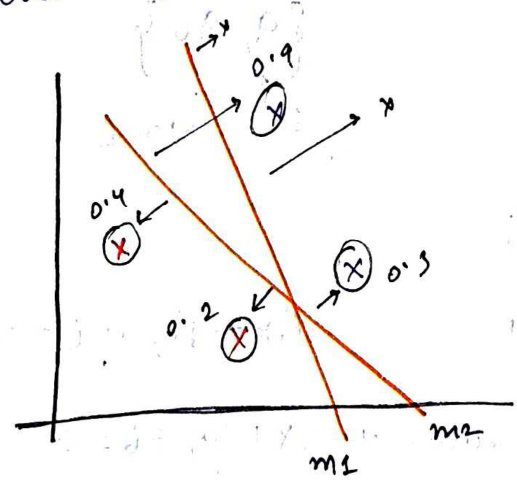
Another problem



* No. missclassified is 0 in both model.

[Classification should not be so black and white]

which model is correct ?

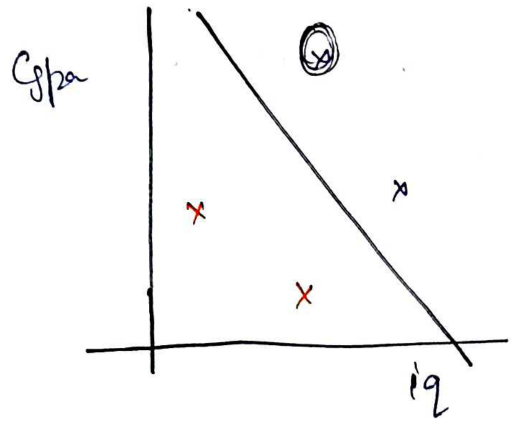


* Find distance between model line and points. Assign distance value as a point value. If distance between points and model line is large that's mean ~~this~~ point are highly positive or negative.

* we want variable not binary number (0,1)

* How algo is works? then we skip part where we classify binary (0,1)

How algo works?



Predicting → 1, 0

1 → purple, 0 → red

iq	cgpa	placed
60	6	0
40	4	0
80	8	1
90	9	1

$\{9, 90\}$ → x_1, x_2

$$Ax + By + C = 0 \rightarrow \beta_0 + \beta_1 X_1 + \beta_2 X_2$$

Let assume → $3x + 5y + 6 = 0$

$$3 \times 9 + 5 \times 90 + 6$$

→ $150 > 0 \rightarrow 1$

$< 0 \rightarrow 0$

$\beta_0 + \beta_1 x_1 + \beta_2 x_2 = Z$

$$\boxed{\begin{array}{ll} Z > 0 & 1 \\ Z < 0 & 0 \end{array}}$$

Step

$$Z \rightarrow \beta_0 + \beta_1 X_1 + \beta_2 X_2$$

$\boxed{\text{Step}(Z)} = (0,1)$ → Skip this step function

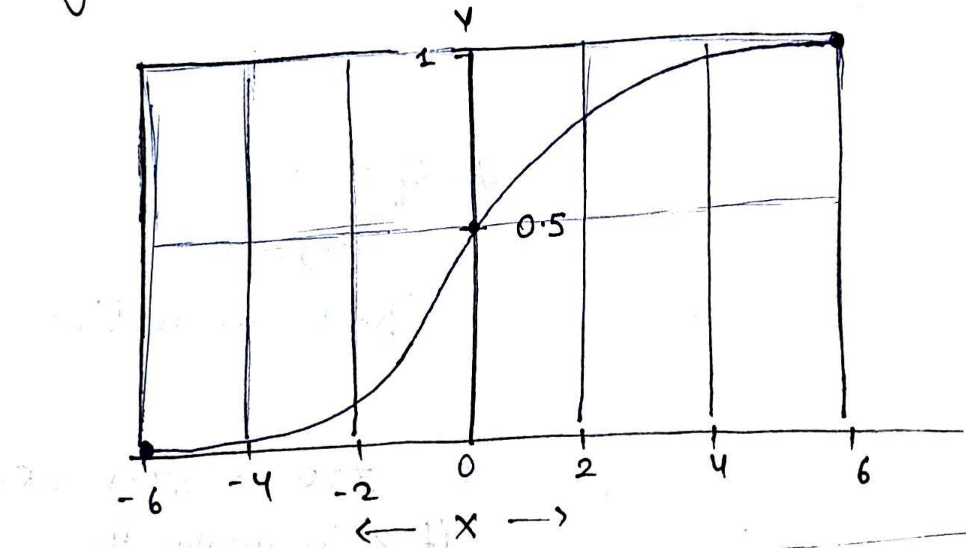
we want $Z\uparrow$ 0.9 this type of value

Z 0.6

$Z\downarrow$ -0.1

Sigmoid function

(use sigmoid insted of step funct)

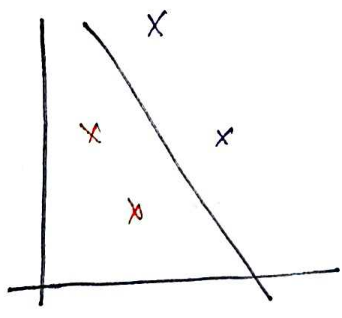


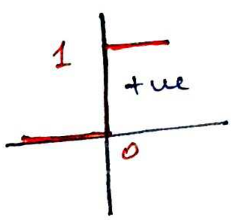
$$\boxed{Y = \frac{1}{1 + e^{-X}}}$$

↳ Sigmoid

* tries to touch 1
* if $x \to \infty$ then $Y = 1$
* if $X \to 0$ then $Y = 0.5$
* if $X \to -\infty$ then $Y = 0$

Step fun.

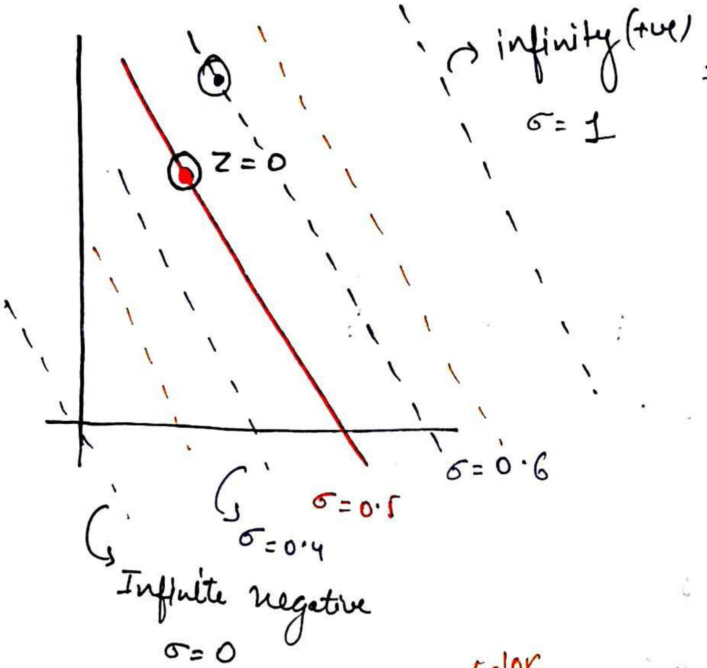




X +ve → Y = 1

X -ve → Y = 0

$\sigma(z) \rightarrow$ Depend on the value of z
$1 - 0$



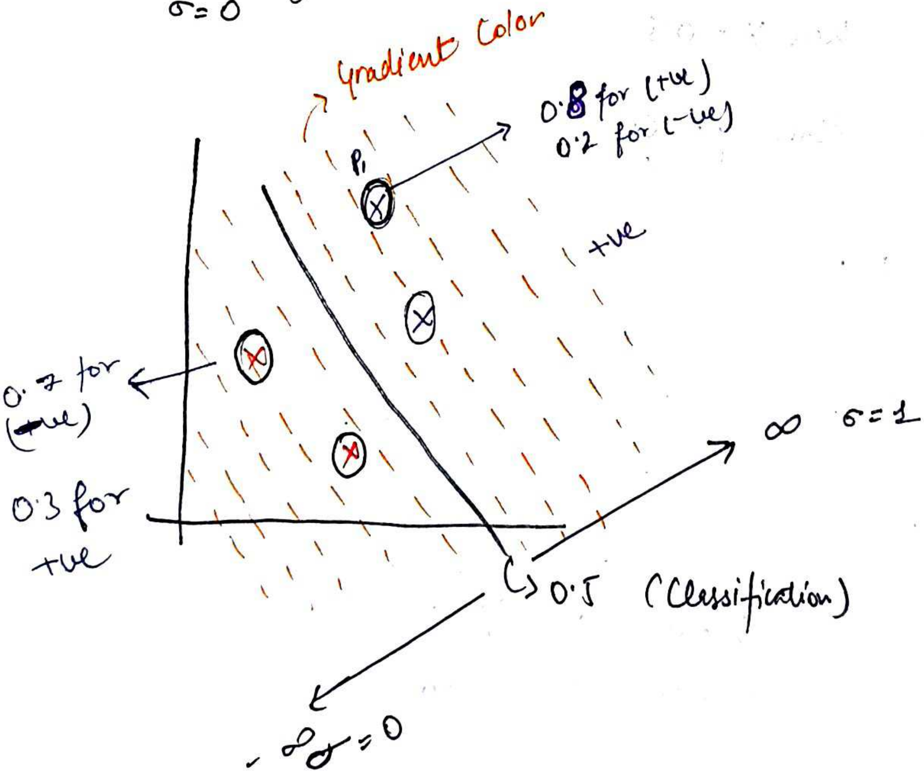
If z = 0

$\sigma(z) = 0.5$

point lies on line

* $z > 0 = \sigma(z) = 0.5\uparrow$

if z is greater than 0 then sigmoid value is more than 0.5.



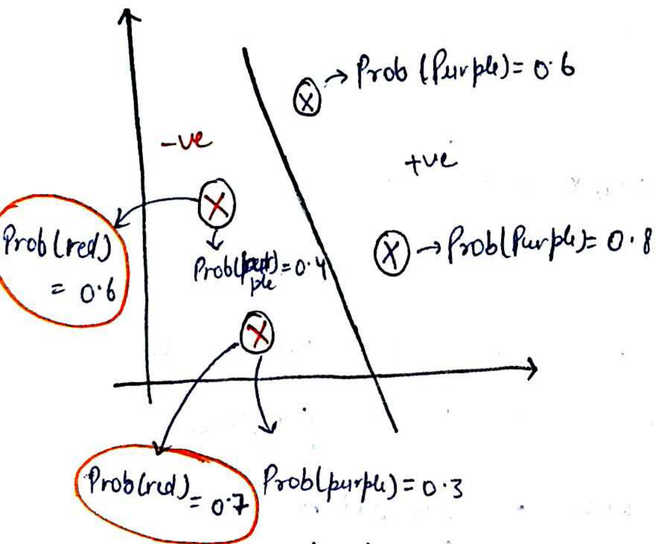
Prabability inter

1. Probability of point(P_1) for +ve = 0.8 (probability)

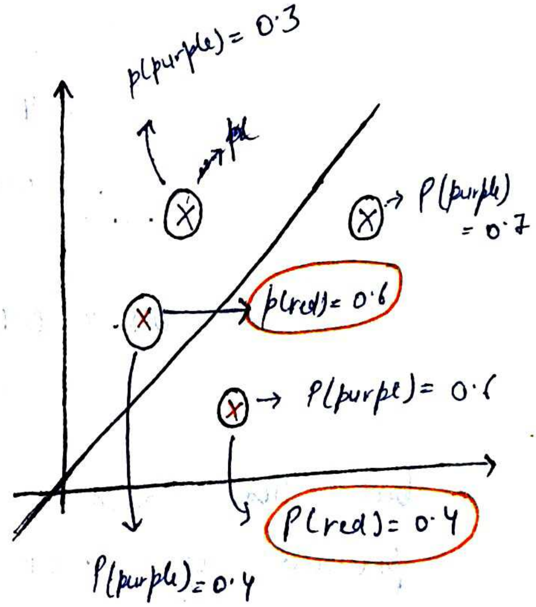
if probability of point (P_1) for +ve = 0.8 then,
probability of point (P_1) for -ve = 0.2

Maximum Likelihood

The likelihood function is the products of the predicted probabilities for the actual class of each observation.



model = 1



model = 2

$M_1 = 0.6 \times 0.8 \times 0.6 \times 0.7 = 0.20$ → maximum

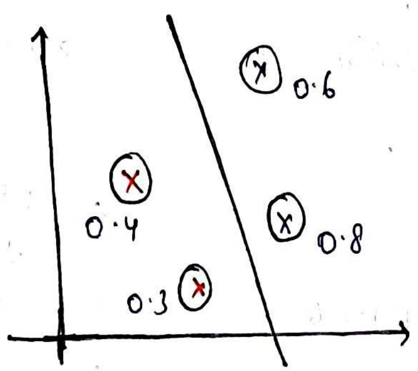
→ model 1

$M_2 = 0.7 \times 0.3 \times 0.6 \times 0.4 = 0.05$

* Maximum product → best line
  → better model
  → logistic Regression line

* Find line → Maximum product

Log Loss



$\boxed{ML = 0.8 \times 0.6 \times 0.6 \times 0.7}$

↳ 4 points

but in dataset, there are 1000+ rows and product of 1000+ row is create problem

Problem : Product 1000+ row = small ≃ 0

↳ underflow

Solution : Use log

$$\log(ML) = \log(0.8 \times 0.6 \times 0.6 \times 0.7)$$

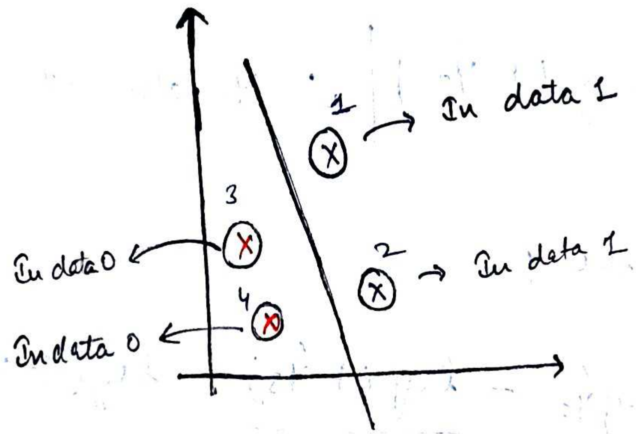
$$= \log(0.8) * \log(0.6) * \log(0.6) * \log(0.7)$$

* log between 0 and 1 number is alway be -ve, so will use $-\log()$

$$= -\log(0.8) * -\log(0.6) - \log(0.6) - \log(0.7)$$

* Answer in positive

* In product of points, we choose maximum number but in log() we choose minimum number answer.



$$-\log(\hat{y}_1) - \log(\hat{y}_2) - \log(\hat{y}_3) - \log(\hat{y}_4)$$ X

This formula is for only prob (purple) but we are using predicted or prob (red).

prob(red) prob(red)

Transform

$\hat{y}_i = \sigma(z) \rightarrow$ Prob $\rightarrow$ getting purple

$i \rightarrow 1, 2, 3 -$

$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2$

$\left.\begin{matrix} y_1 \\ y_2 \end{matrix}\right\}$ getting purple

$$\boxed{-y_1 \log \hat{y}_1 - (1-y_1)\log(1-\hat{y}_1)}$$

example!

for $y_1 = -y_1 \log \hat{y}_1 - (1-y_1)\log(1-\hat{y}_1)$

$= -y_1 \log \hat{y}_1 - \underbrace{(1-1)}_{=0} \log(1-\hat{y}_1)$

$\boxed{y_1 = -y_1 \log \hat{y}_1}$

$\boxed{y_2 = -y_1 \log \hat{y}_1}$ $\therefore$ y_2 also purple and in data is 1 for $y_3 = -0 \log \hat{y}_3 - (1-0) \log(1-\hat{y}_3)$

$$\boxed{y_3 = -(1-0) \log(1-\hat{y}_3)}$$

$$\boxed{y_4 = -(1-0) \log(1-\hat{y}_3)}$$ $\therefore$ y_4 : also red and In data is 0.

Add all the y.

$$\underbrace{-\log \hat{y}_1}_{\text{for positive}} - \log \hat{y}_2 - \underbrace{\log(1-\hat{y}_3)}_{\text{for negative}} - \log(1-\hat{y}_4)$$

formula :-

$$\sum_{i=1}^{n} -y_i \log \hat{y}_i - (1-y_i) \log(1-\hat{y}_i)$$

for Avg :

$$\frac{1}{n} \sum_{i=1}^{n} -y_i \log \hat{y}_i - (1-y_i) \log(1-\hat{y}_i)$$

$$\boxed{L = -\frac{1}{n} \sum_{i=1}^{n} -y_i \log \hat{y}_i - (1-y_i) \log(1-\hat{y}_i)}$$

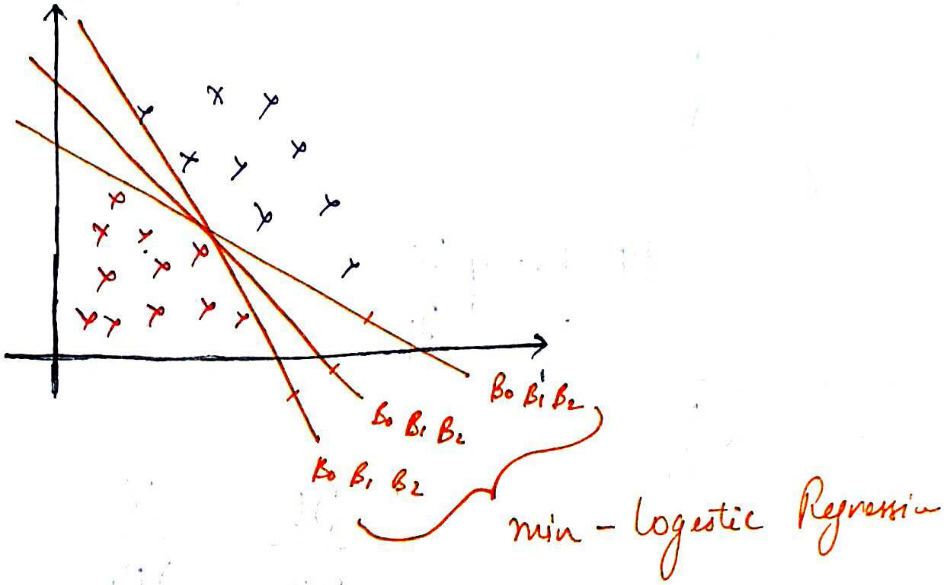
Log function

log Loss Error

Binary Cross entropy

* $\beta_0 \; \beta_1 \; \beta_2 \ldots$ need value to minimize the value of L.

$$\boxed{\hat{y}_i = \sigma(z_i)} \quad \boxed{z_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i}}$$



Using Gradient Descent

$$\boxed{L = -\frac{1}{n}\sum_{i=1}^{n} y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)}$$

X_1	X_2	Y	$\hat{Y}$
28	13	0	0.63
25	12	1	0.37
⋮	⋮	⋮	⋮

$\hat{y}_i = P(\text{green})$

$(1-\hat{y}) = P(\text{red})$

* we cannot find exact value of $\beta_0\ \beta_1\ \beta_2$ like ~~xxxx~~ with the help of use $\rightarrow$ $\beta_0\ \beta_1\ \beta_2$ we we find value but not in Logistic Regression. So, we use gradient descent and estimate value of $\beta_0\ \beta_1\ \beta_2$.

$\beta_0 \; \beta_1 \; \beta_2 \rightarrow$

$$\beta_0 = \beta_0 - \eta \boxed{\frac{\partial L}{\partial \beta_0}}$$

η → learning rate

$\frac{\partial L}{\partial \beta_0}$ → gradient

$$\boxed{\beta_1 = \beta_1 - \eta \frac{\partial L}{\partial \beta_0}}$$

$L \rightarrow \frac{\partial L}{\partial \beta_0}$ differentiate w.r.t β_0

$$\boxed{\beta_2 = \beta_2 - \eta \frac{\partial L}{\partial \beta_0}}$$

$$\frac{\partial L}{\partial \beta_0} \quad \frac{\partial L}{\partial \beta_1} \quad \frac{\partial L}{\partial \beta_2}$$

$$\frac{\partial L}{\partial \beta_1} = -\frac{1}{n} \sum_{i=1}^{n} y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)$$

↳ find differentiate w.r.t β_1

let assume we have 1 point not n.

$$\frac{\partial L}{\partial \beta_1} = \frac{\partial}{\partial \beta_1}\left[-y \log \hat{y} - (1-y)\log(1-y)\right]$$

$$= \frac{-y}{\hat{y}} \sigma(z)\left[1-\sigma(z)\right] x_1$$

$\therefore \; \hat{y} = P(\text{green})$

$\downarrow$

$\sigma(z)$

$\downarrow$

$\beta_0 + \beta_1 x_1 + \beta_2 x_2$

differentiate of sigmoid

↳ sig (1 - sigmoid)

$$= \frac{-y}{\cancel{\hat{y}}} \cancel{\hat{y}}(1-\hat{y}) x_1$$

$$= \boxed{-y(1-\hat{y})x_1}$$

differentiate $-(1-y)\log(1-\hat{y})$ w.r.t β_1

$$= +\frac{(1-y)}{\cancel{(1-\hat{y})}}\,\hat{y}\,\cancel{(1-\hat{y})}\,x_1$$

$$= \boxed{(1-y)\,\hat{y}\,x_1}$$

$$= -y(1-\hat{y})x_1 + (1-y)\hat{y}x_1$$

$$= [-y + \cancel{y\hat{y}} + \hat{y} - \cancel{\hat{y}y}]x_1$$

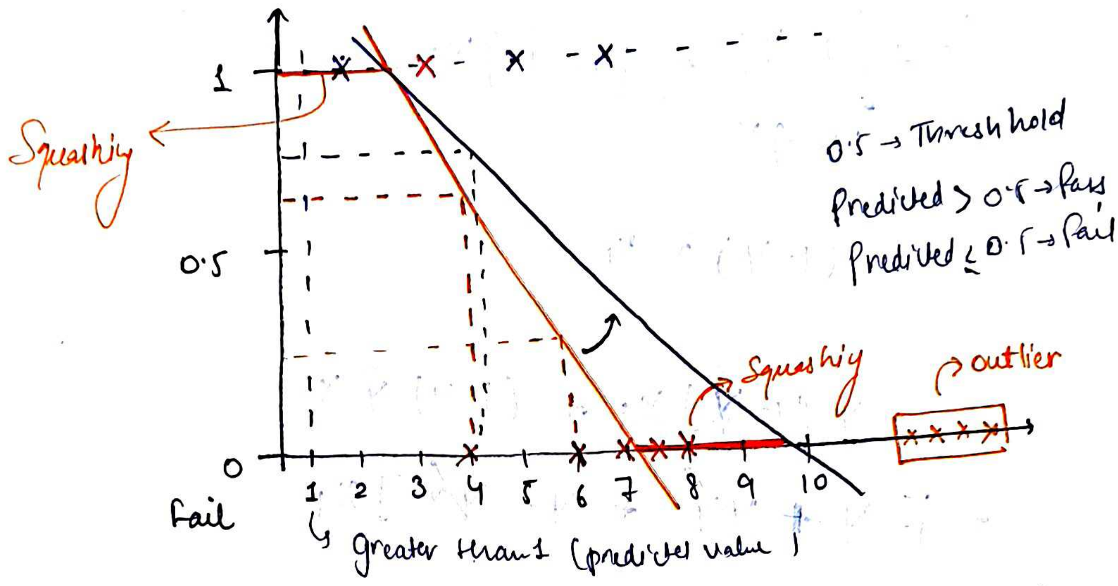
$$= [-y + \hat{y}]x_1$$

$$\boxed{\frac{\partial L}{\partial \beta_1} = [\hat{y}_i - y_i]x_{i1}}$$

$$\frac{\partial L}{\partial \beta_2} = [\hat{y}_i - y_i]x_2$$

$$\boxed{\frac{\partial L}{\partial \beta_0} = [\hat{y}_i - y_i]}$$

Can we solve the classification problem using Regression?



Why we cannot use Linear Regression for Classification?

① Best fit line changes because of outliers and prediction gone wrong

② The output comes greater than 1 and less than 0.

To solve this problem we use Logistic Regression

↳ 0 to 1 → Squashing