Logistic Regression

To solve classification → Binary → Dependent → 2 categories
→ Multiclass classification → More than 2 categories.

Dataset

UPSC Exam

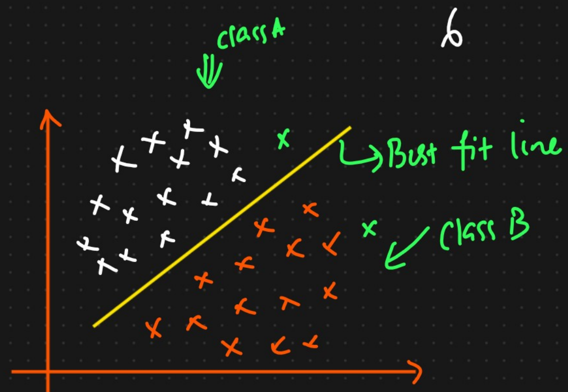
No. of study hours	O/p Pass/Fail
0	0
1	0
2	0
3	0
2	1
4	1
5	1
6	1

TRAIN ↓

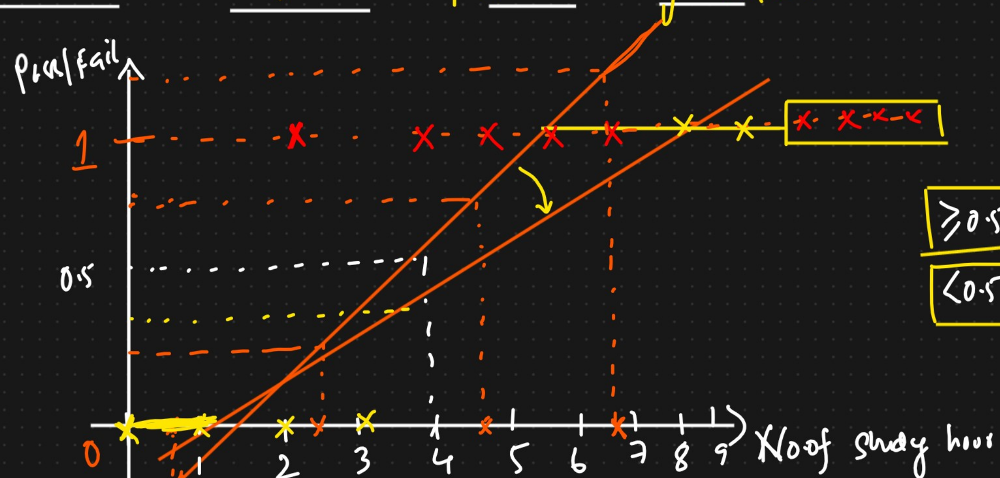
New test data → Model → Pass/Fail

↓

Acc ↑↑



Can we solve this classification problem using Regression?



① Outliers
② >1 and <0

$\boxed{\geq 0.5} \Rightarrow 1$

$\boxed{< 0.5} \Rightarrow 0$

$\boxed{0 \text{ or } 1}$ $\quad > 1 \;\&\; < 0$

$\Downarrow$

Best fit line $\quad h_\theta(x) = \theta_0 + \theta_1 x_1 \Rightarrow z$

$\Downarrow$

Squash the best fit line $\Rightarrow$ Sigmoid fn $\Rightarrow$ 0 to 1 $\Rightarrow \dfrac{1}{1+e^{-z}}$

① How does Logistic Regression solve classification

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

$\Downarrow$

Logistic Regression

Logistic Regression

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

$h_\theta(x)$

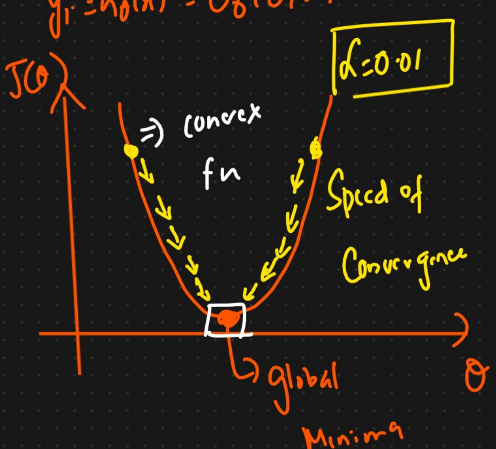
$\Uparrow$

$\hat{y}_1 = \theta_0 + \theta_1 x_1$

Linear Regression Cost fn

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

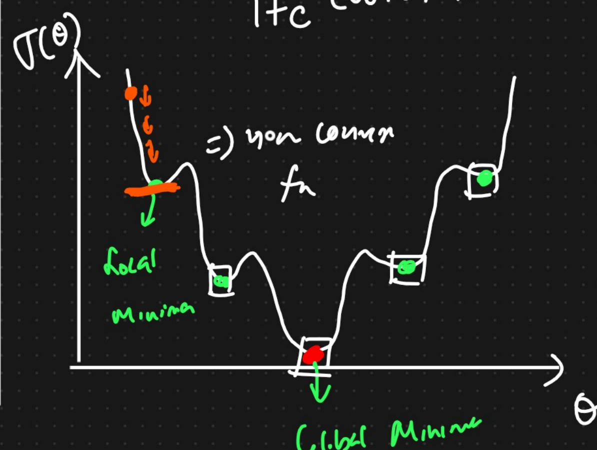
$\hat{y}_i = h_\theta(x) = \theta_0 + \theta_1 x_1$



Logistic Regression Cost fn

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

$$h_\theta(x) = \frac{1}{1+e^{-(\theta_0+\theta_1 x_1)}} \Rightarrow 0 \text{ to } 1$$



Log Loss Cost fn

$$J(\theta_0,\theta_1) = -\underset{\Downarrow}{\overset{\hat{y}_i}{y_i}} \log(h_\theta(x)_i) - (1-y_i)\log(1-h_\theta(x)_i) \Rightarrow \text{Gradient Descent Curve.}$$

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

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

Final Aim : Minimize Cost fn $J(\theta_0,\theta_1)$ by changing θ_0, θ_1

Convergence

Repeat until Convergence

$\{$

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

$\}$

Logistics Regression With Reguralization Parameters

Cost fn

$$J(\theta_0,\theta_1) = -y\log(h_\theta(x)) - (1-y)\log(1-h_\theta(x))$$

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

[Reduce Overfitting]

$$J(\theta_0,\theta_1) = -y\log(h_\theta(x)) - (1-y)\log(1-h_\theta(x)) + L_2 \text{ Reguralization}$$

$$J(\theta_0,\theta_1) = -y\log(h_\theta(x)) - (1-y)\log(1-h_\theta(x)) + L_1 \text{ Reg}$$ [Feature Selection].

$$J(\theta_0,\theta_1) = -y\log(h_\theta(x)) - (1-y)\log(1-h_\theta(x)) + L_2 \text{ Reg} + L_1\text{Reg}.$$

L2 Regularization ⇒ Reduce Overfitting ⇒ Ridge Regression.

$$J(\theta_0,\theta_1) = -y\log(h_\theta(x)) - (1-y)\log(1-h_\theta(x)) + \lambda \sum_{i=1}^{n} (\text{slope})^2$$

L1 Regularization ⇒ feature selection ⇒ Lasso

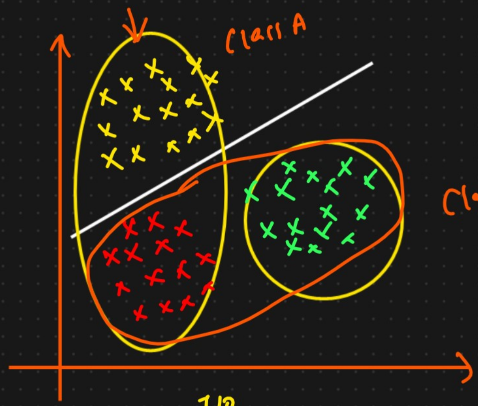
$$J(\theta_0,\theta_1) = -y\log(h_\theta(x)) - (1-y)\log(1-h_\theta(x)) + \lambda \sum_{i=1}^{n} |\text{slope}|$$

Elastic Net

$$J(\theta_0,\theta_1) = -y\log(h_\theta(x)) - (1-y)\log(1-h_\theta(x)) + \lambda_1 \sum_{i=1}^{n} (\text{slope})^2 + \lambda_2 \sum_{i=1}^{n} |\text{slope}|.$$

$$\boxed{C \propto \frac{1}{\lambda}} \qquad \boxed{C}$$

Multiclass Logistic Regression



I/P

f1	f2	f3

Multinomial ⇒ Index of o/p

OVR {One Versus Rest} ⇒ Probabilities of all categories.

Label ⇒

O/P	M1	M2	M3
	A	B	C
A	1	0	0
B	0	1	0
C	0	0	1
C	0	0	1
B	0	1	0

OVR

$$\begin{array}{ccc} M_1 & M_2 & M_3 \end{array} \rightarrow [0.20,\ 0.30,\ 0.50]$$

⇓

C

⇓

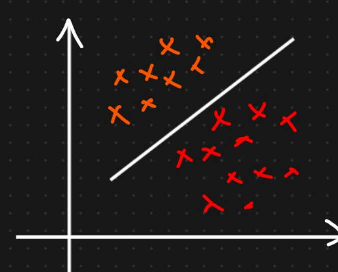
o/p

$\boxed{A \rightarrow 900}$ } Imbalance dataset

$\boxed{B \rightarrow 100}$ ⇒ 1000

{B : 10}

Performance Metrics, Accuracy, Precision, Recall, F-Beta.



① Confusion Matrix

	1	0	Actual
1	2	2	
0	1	1	

Predicted

x_1	x_2	y	$\hat{y}$	
-	-	0	1	→ wrong
		1	1	→ correct
		0	0	
		1	1	
		1	1	
		0	1	
		1	0	

Predicted	1	0	→ Actual
Positive ← 1	TP	FP	
Negative ← 0	FN	TN	

Predicted

$\boxed{\text{FP \& FN Are Errors}}$

$$Acc = \frac{TP + TN}{FP + TP + FN + TN} = \frac{4}{7} = 57.1\%$$

3+2+1+1

DATASET : Imbalance Dataset

1000 datapoints → 900 1's, 100 0's } ⇒ Imbalanced ⇓ Acc = 90%

→ Dumb Model → $\hat{y} \to 1$

	1	0 ← Actual
1	TP	FP
Predicted	FN	TN

③ **Precision** : $\frac{TP}{TP+FP}$ } Out of all the actual values how many are correctly prediction

⇑ FP is Important ↓↓ FP

	1	0
1	TP	FP
0	FN	TN

④ **Recall** : $\frac{TP}{\boxed{TP+FN}}$ } Out of all the predicted value how many are correctly predicted with actual values.

⇓

$\boxed{FN \downarrow\downarrow\downarrow}$ ⇒ Reduce FN

Usecase 1 : Spam classification.

	1	0 / Actual
1	TP	FP↓↓↓
0	FN	TN

Text ⇒ Model ⇒ Spam / Not Spam.

$\boxed{FN \downarrow\downarrow\downarrow}$

Wrong Scenario { Text → Spam, Model → Not a Spam } ⇑ FN

Text ⇒ Spam, Model ⇒ Spam } ⇒ good Scenario. Accurate.

Blunder: { Text → Not a Spam, Model ⇒ Spam }

Text ⇒ Not a Spam, Model ⇒ Not a Spam } ⇒ Accuracy

Usecase : FN is Important

To predict whether a person has diabetes or Not

Actual → Diabetes
Model → Diabetes } Correct ⇒ TP

Actual ⇒ No Diabetes
Model ⇒ Diabetes } FP Wrong Prediction.

Actual → No Diabetes
Model → No Diabetes } Correct ⇒ TN

Actual ⇒ Diabetes
Model ⇒ No Diabetes } FN Blunder

I was Joking

⇓

Gulab Jamoon

Assignment : Tomorrow the Stock Market will crash or Not

Reduce FP or FN

1. Protection of people
2. Protection of Companies.

✓ F-Beta Score