ML #1 LINEAR REGRESSION

1. Simple Linear Regression
2. Multi Linear Regression
3. Polynomial Regression
4. Performance (Accuracy)
5. LASSO | RIDGE | Elastic Net

① SIMPLE LINEAR REGRESSION

- Find Best Fit line in such a way that sum of Error is Minimum

$\boxed{L.R} \rightarrow$ B.F.L with Min(Σ Error)

DATASET

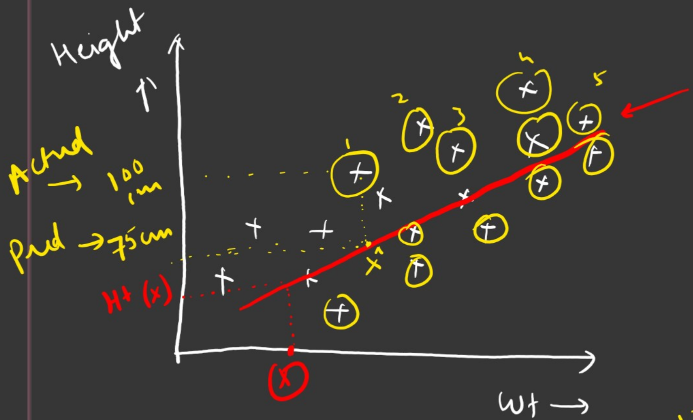
① → Independent	① → dependent
Weight	Height
74	170
80	180
75	175
⋮	⋮

Wt → Ht

1 (Input feature) → 1 (O/P feature)

Wt → $\boxed{\text{Model}}$ → Ht

New Wt → Model → New Ht



Best Fit line (B.F.L)

① ERROR = Actual Point − Predicted Point

= 100 − 75 = 25cm

$$\boxed{\Sigma \text{Error} = \text{Minimum}}$$

Clear the Concepts

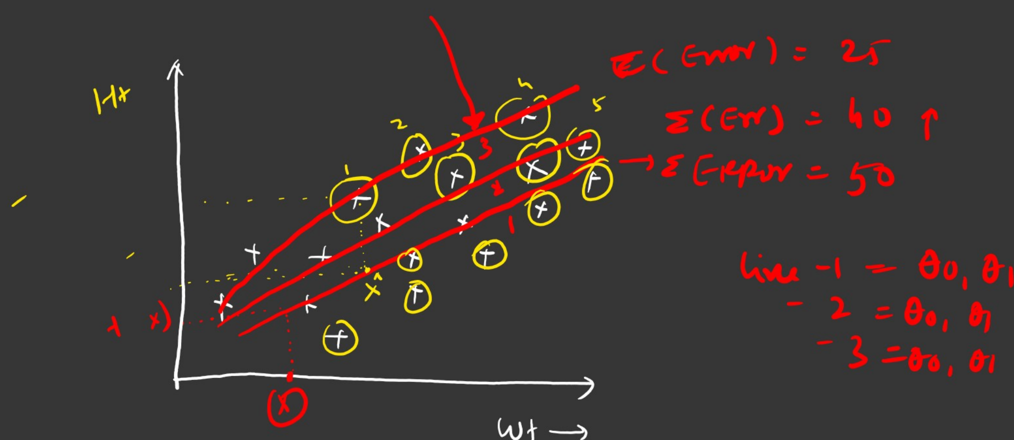
$$\left.\begin{aligned} y &= mx + c \\ h_\theta(x) &= \theta_0 + \theta_1 x \end{aligned}\right\}$$

θ_0 → Intercept

θ_1 → Slope

$\boxed{x = 0 \; ; \; y = ?}$

Unit Movement of x → movement of y.



Constructing B.F.L with diff Value of θ_0, θ_1 to Reduce error → Optimization

COST Function

$$J(\theta_0, \theta_1) = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - h_\theta(x)\right)^2 \rightarrow \text{Mean Squared ERROR}$$

Actual Value

Predicted Value

Final Aim :-

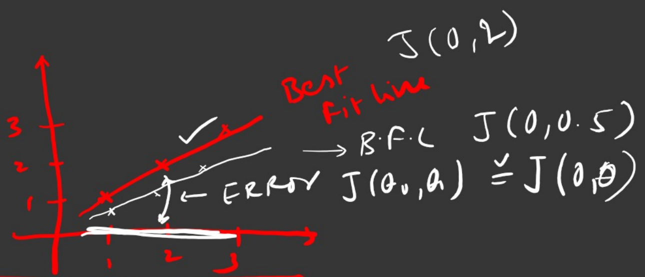
Minimize (Cost Function)

$$\min \left[J(\theta_0, \theta_1) \right] = \frac{1}{n} \sum_{i=1}^{n} (y_i - h_\theta(x))^2 \downarrow\downarrow\downarrow$$

DATASET

Min the $\boxed{\text{Cost Function}}$ → ERROR

x	y
1	1
2	2
3	3



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

n = No. of DATA Point
y_i = Actual Value
$h_\theta(x)$ = Predicted Value

$h_\theta(x) = (\hat{y}_i)$ ← Predicted Value

Lets Assume,

$\theta_0 = 0$

$\theta_1 = 1, 0, 0.5$

$\theta_1 = 1$

$h_\theta(x) = \cancel{\theta_0} + \theta_1(x)$

$x=1$ $h_\theta(1) = 1(1) = 1$

$x=2$ $h_\theta(2) = 1(2) = 2$

$x=3$ $h_\theta(3) = 1(3) = 3$

$\theta_1 = 0$

$x=1$

$h_\theta(1) = 0 + 0(1) = 0$

$h_\theta(2) = 0$

$h_\theta(3) = 0$

$\theta_1 = 0.5$

$x=1$

$h_\theta(1) = 0 + 0.5(1) = 0.5$

$h_\theta(2) = 0 + (0.5)(2) = 1$

$h_\theta(3) = 0 + 0.5(3) = 1.5$

Calculate Cost Function

$\theta_1 = 1$

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

$$= \frac{1}{3} \left[\cancel{(1-1)^2} + \cancel{(2-2)^2} + \cancel{(3-3)^2} \right]$$

$$\boxed{J(\theta_0, \theta_1) = 0}$$

$\theta_1 = 0$

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

$$= \frac{1}{3} \left[(1-0)^2 + (2-0)^2 + (3-0)^2 \right]$$

$$= \frac{1}{3} [1 + 4 + 9] = \frac{14}{3} \cong 4.6$$

$\theta_1 = 0.5$

$y_i = (1, 2, 3)$

$h_\theta(x) = (0.5, 1, 1.5)$

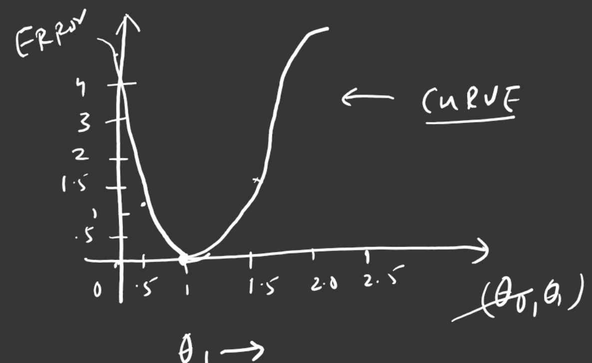
$$J(\theta_0, \theta_1) = \frac{1}{n} \sum_{i=1}^{n} (y_i - h_\theta(x))^2$$

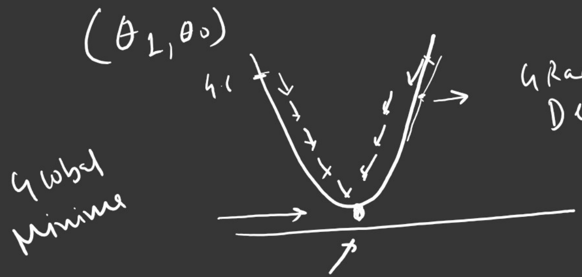
$$= \frac{1}{3} \left[(1-0.5)^2 + (2-1)^2 + (3-1.5)^2 \right]$$

$$= \frac{1}{3} \left[(0.25) + 1 + 2.25 \right]$$

$$= 1.16$$

$J(\theta_0, \theta_1)$	Σ ERROR	
(0, 1) →	0	→ Min Error
(0, 0.5)	1.16	
(0, 0)	4.6	
1.5		





Gradient Descent → to find slope in the curve

Convergence Algorithm → $\boxed{\text{PARABOLA}}$

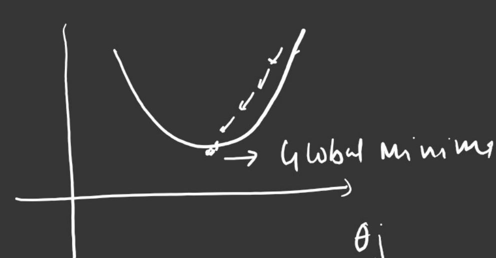
- Optimized the change of θ_j

Repeat until convergence

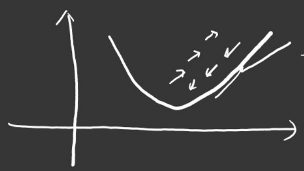
$$\{$$

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

$$\}$$



α = Learning Rate (ML)



→ tangent → slope

+ve ↙ ↘ -ve

$$\theta_j : \theta_j - \alpha \boxed{\frac{\partial}{\partial \theta_j} J(\theta_j)} \rightarrow \text{slope}$$

-0.5 ← ϕ → 0.5 1

↓ ↑

Slope = +ve

$\theta_j(\text{new}) := \theta_j - \alpha(x)$ ↓

Slope : +ve

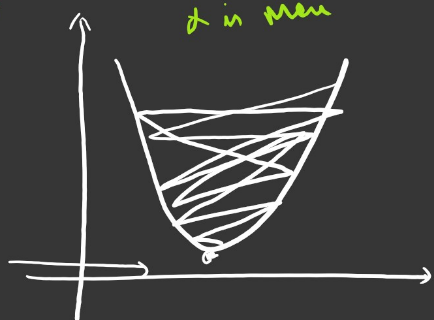
$\theta_j(\text{New})\uparrow = \theta_j - \alpha(-\text{slope})$

$\theta_j + \alpha x$

α → learning Rate → Speed at which convergence happens

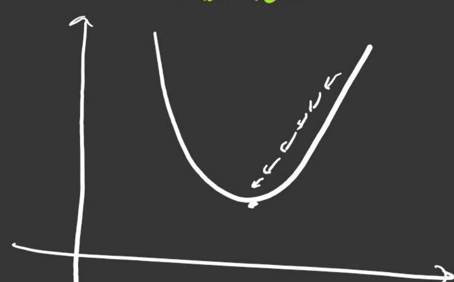
α is more

∅ Random



α is less

∅ Time consuming

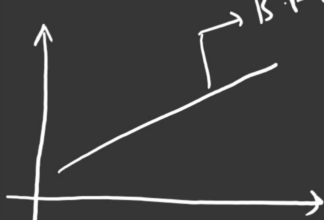


α is ↑↑

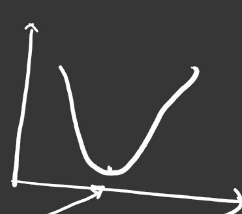
Algorithm optimized α

$\boxed{\alpha = 0.2}$ Sklearn

Conclusion



B.F.L



Min Error

Gradient Descent → Optimizer Function

∅ Convergence Algorithm

(θ_j)

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

B.F.L

Multi-linear Regression

DATASet

I/P (>1) → No. of Room (x_1), Size of Room (x_2)

O/P → Price ($h_\theta(x)$)

No. of Room	Size of Room	Price
x_1	x_2	$h_\theta(x)$

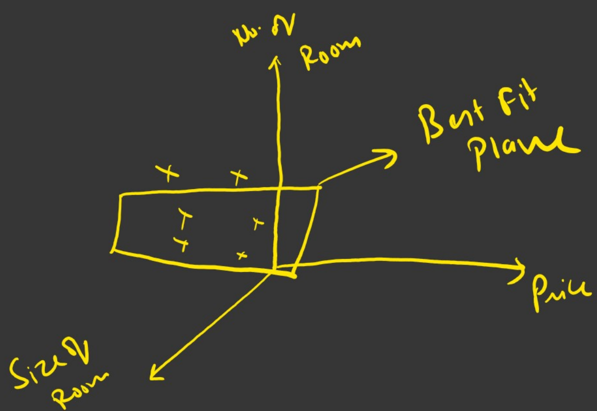
Simple linear Regression

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

Multi linear Regression

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

"BEST FIT PLANE"



I >4 >5 $\boxed{>6}$ → Reduce feature → 2,3 → Best fit plane / Best fit line

>4, >5 → No → Best fit plane

$n \to$ Feature

Generic Eqn for Multi-linear Regression

$$\boxed{h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots \theta_n x_n}$$

Imp

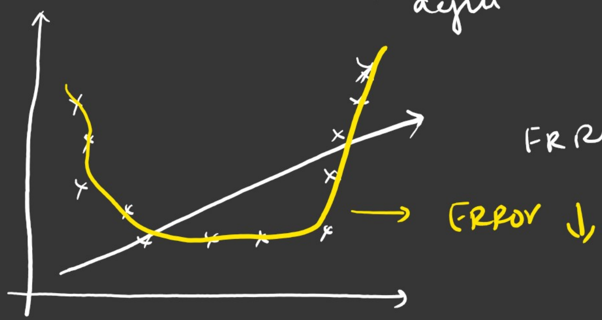
4D - Diagram

$$\left\{ \begin{array}{l} 2D \to \text{line} \\ > 2D \to \text{Plane} \end{array} \right\}$$

③ Polynomial Regression

↳ Polynomial degree

Regression with Non Linear Element



→ Non linear Relationship

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

Multiple :- $h_\theta(x) = \theta_0 + \theta_1 x_1 + \dots + \theta_n x_n$

Polynomial Degree (PD)

PD = 0 $(\theta_0 = 0)$

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

$h_\theta(x) = \theta_0 x_1^0$

PD = 1

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

$= \theta_0 + \theta_1 x_1^{(1)}$

(simple linear Rg)

PD = 2

$h_\theta(x) = \theta_0 x_0^0 + \theta_1 x_1^1 + \theta_2 x_2^2$

∴ Overfitting

PD = 3

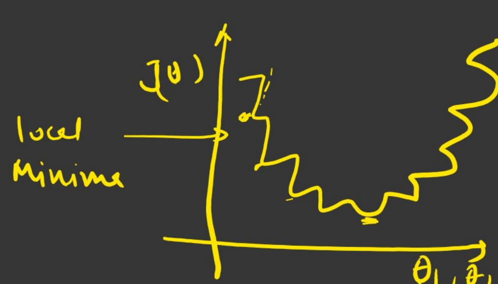
$h_\theta(x) = \theta_0 x_0^0 + \theta_1 x_1^1 + \theta_2 x_2^2 + \theta_3 x_3^3$

PD = n

$$\boxed{h_\theta(x) = \theta_0 x_0^0 + \theta_1 x_1^1 + \theta_2 x_2^2 + \theta_3 x_3^3 + \dots + \theta_n x_n^n}$$

Polynomial Regression

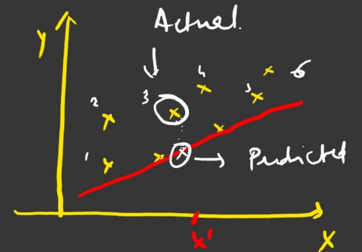
→ create line which does not overfit



(4) PERFORMANCE OF Model

(1) R-Squared
(2) Adjusted R-Squared } Statistics

1. R-Squared



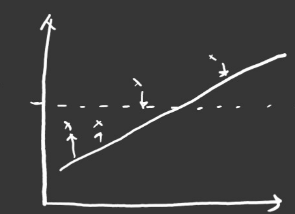
$$R\text{Squared} = 1 - \frac{SS_{Res}}{SS_{Total}}$$

SS_{Res} = Sum of Squared Residual or ERROR
SS_{Total} = Sum of Squared Total

Residual = Actual - Predicted

$$\boxed{SS_{Res} = (ERROR_1)^2 + (ERROR_2)^2 + \dots + (ERROR_6)^2}$$

SS Total



$$R\text{squared} = 1 - \frac{\sum (y - \hat{y})^2 \rightarrow \text{ERROR}}{\sum (y - \bar{y})^2 \rightarrow \text{Big value}}$$

(Mean Value)

$$= 1 - \left(\frac{4}{10}\right)$$

$$R\text{squared} = 1 - 0.4 = \underline{60\%}$$

$\boxed{\text{Max Rsquared} = 1}$ → linear Reg Model → overfitting

Min Rsquared = 0

Adjusted Rsquared

~~Gender~~ # of Room   Size   location   PRICE

$R_{sq} = 90\%$
$R_{sq} = 91\%$

Input feature → ↑
Model Acc → ↑

Adding New feature → R-Squared will keep on Increasing

Adjusted R-Squared

$$= 1 - \frac{(1-R^2)(N-1)}{N-P-1}$$

N = No of data point
P = No of Independent feature

$R^2 = 80\%$, $N = 11$, $P = 2$

$$\text{Adjusted } R^2 = 1 - \frac{(0.2)(10)}{11-2-1} = 1 - \frac{(0.2)(10)}{11-2-1} = 1 - \frac{2}{8}$$

$$= 1 - 1/4$$
$$= 3/4 \simeq 75\%$$

★ Adjusted R^2 will always be less than R^2

$P = 3$ $R^2 = 85\%$ ⇒ Adjusted $R^2 = 77\%$ [Important Feature]

$P = 4$ $R^2 = 86\%$ ⇒ Adjusted $R^2 = 77\%$ [Not important feature]

4 LIMITATIONS of Linear Regression

1. Overfit
2. Multiple feature → R^2 ↑

RIDGE / LASSO / Elastic Net

SP-1 RIDGE (L2 Regularization)

Reducing Overfitting

L.R

$$\text{Cost Function} = \frac{1}{n} \sum_{i=1}^{n} (\underset{\text{Actual}}{y_i} - \underset{\text{Predicted}}{h_\theta(x)})^2$$

Reduce over fitting → cost function → 0

$$\text{Cost Func}^n = \frac{1}{n}\sum_{i=1}^{n}(y_i - h_\theta(x))^2 + \boxed{\lambda \sum_{i=1}^{n}(\text{slope})^2} \rightarrow \text{Penalizing cost function}$$

λ → hyper-parameter

② LASSO Regression

L_1 → Regularization

Feature Selection

$$\text{Cost Func}^n = \frac{1}{n}\sum_{i=1}^{n}(y_i - h_\theta(x))^2 + \lambda \sum_{i=1}^{n}|\text{slope}|$$

Multi linear Regression

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3$$

$$= 0.52x + 0.65x_1 + 0.72\,x_2 + \cancel{\boxed{0.12\,x_3}}$$

Very less Co-related !

③ Elastic Net

- Combination of Ridge and Lasso Regression

Ridge → Reduce overfitting

Lasso Regression → feature selection

$$\text{Cost Func}^n = \frac{1}{n}\sum_{i=1}^{n}(y_i - h_\theta(x))^2 + \underbrace{\lambda_1 \sum_{i=1}^{n}(\text{slope})^2}_{\text{Reduce overfitting}} + \underbrace{\lambda_2 \sum_{i=1}^{n}|\text{slope}|}_{\text{Feature selection}}$$

Hyperparameter tuning = λ_1, λ_2

In Python (Sklearn)

Hyper Parameter = C

$$\boxed{C = \frac{1}{\lambda}}$$

---

② LOGISTICS REGRESSION

- To solve classification problem
  - ↙ Binary
    - ↙ O/P [2 categories]
  - ↘ Multi class classification
    - > 2 categories

DATASET

No. of Play Hours	Result
9	Fail
8	Fail
7	Fail
6	Fail
5	pass
4	pass
3	pass

TRAIN ↓ Model

New DATA → Model → PASS / Fail

Model ↓ Accuracy

PASS

→ × Best fit line

→ Fail

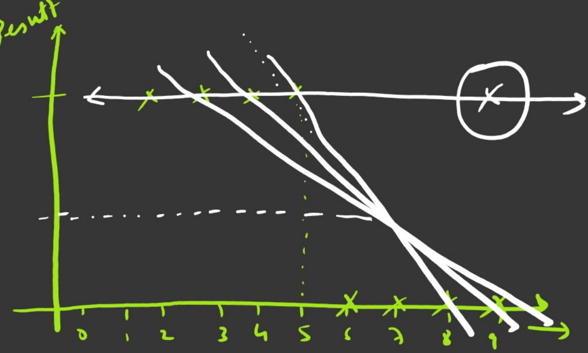
→ Linear Regression ✗

Result

(PASS) 1

Threshold 0.5

(Fail) 0



Play hrs

Pred (5) = Pass ✓

Pred (8) = pass ✓

★ Why we cannot use linear Regression for classification

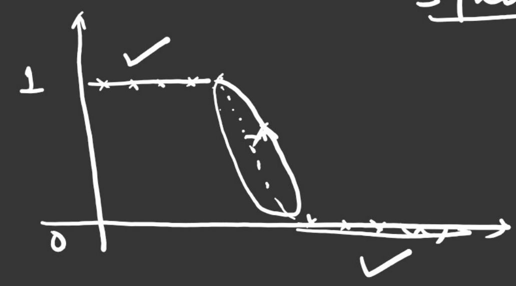
Ans

① Prediction will go wrong

② B.F.L changes because of outlier

③ classification O/P {1 or 0}

"Squashing Algorithm"



Pred (x)

threshold principle.

$\geq 0.5 \Rightarrow 1$

$< 0.5 \Rightarrow 0$

Linear Regression

$h_\theta(x) = \theta_0 + \theta_1 x \rightarrow$ Best fit line

"Sigmoid Activation function" ★

Transform the line {0, 1}

Generic eqn of Logistic func → $\boxed{\sigma = \frac{1}{1+e^{-z}}}$

$\boxed{z = h_\theta(x) = \theta_0 + \theta_1 x}$

From linear Regression

Cost Func^n $= \frac{1}{n} \sum_{i=1}^{n} (y_i - h_\theta(x))^2$

Logistic Regression

Log loss function $= -y_i \log(h_\theta(x_i)) - (1-y_i)\log(1-h_\theta(x))$

$h_\theta(x_i) \Rightarrow$ Predicted Value

y_i = Actual Value

If $y_i = 1$

$L.L.F = -1\log(h_\theta x_i) - 0$

If $y_i = 0$

$L.L.F = -(1-\cancel{y_i})\log(1-h_\theta(x))$

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

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

Goal

Minimize cost fun (LLF) by changing (θ_0, θ_1)

* Convergence Algorithm

Repeat Untill Convergence

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

Logistic Regression with Regularization Parameter

Cost Funcⁿ

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

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

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

L1 / L2 / Elastic Net

L2 Regularization ⇒ Reduce Overfitting

$$J(\theta_0,\theta_1) = -y\log(h_\theta(x)) - (1-y)\log(1-h_\theta(x)) + L2$$

$$= \qquad + \lambda\sum_{i=1}^{n}[\text{slope}]^2$$

L1 Regularization (Feature Selection)

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

Elastic Net [L1 + L2]

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

Model Performance of Logistic Regression

1. Confusion Matrix
2. Accuracy
3. Precision
4. Recall
5. F-Beta Score

DATASET

Actual Value

Predicted value

x_1	x_2	y	$\hat{y}$	
—	—	0	1	× wrong Prediction
—	—	1	1	✓
—	—	0	0	✓
—	—	1	1	✓
—	—	1	1	✓
—	—	0	1	×
—		1	0	×

1. Confusion Matrix

→ Actual Value

	1	0
1	|||	~~1~~ 2
0	|	|

Predicted Value

	1	0
1	3	2
0	1	1

Predicted value

	1	0 (Actual)
1	TP	FP
0	FN	TN

(Predicted)

② Model Accuracy $= \dfrac{TP + TN}{TP + FP + FN + TN} = \dfrac{3+1}{3+2+1+1} = 4/7$

$= 57\%$

③ PRECISION

Dataset → Imbalanced Dataset

1000 D.P → 900 (1)
1000 D.P → 100 (0)

Precision

TP	FP
FN	TN

$$\text{Precision} = \frac{TP}{TP + FP}$$

"Out of All the actual value, how many are Correctly Predicted"

When FP is important

④ ~~Recall~~ RECALL

$$R = \frac{TP}{TP + FN}$$

TP	FP
FN	TN

FN is important

Out of all the Predicted Value, how many are Correctly Predicted with Actual Value

---

USE CASE → SPAM Classification — FP is important

Text ⇒ Model → SPAM
Text ⇒ Model → Not-SPAM

	1	0
1	TP	FP
0	FN	TN

TP: Mail → SPAM, Model → SPAM } Accurate

TN: Mail → Not SPAM, Model → Not a Spam } ✓

FP: { Mail → Not Spam, Model → Spam } ✗ Wrong Predictions

PRECISION

✉ → 0 → SPAM } "Blunder"

FN: Mail → Spam, Model → Not Spam } ✗

Use case #2 False Negative is important

$\rightarrow$ M $\rightarrow$ Diabetes

Actual

Pred \ Actual	D	ND
D	TP	FP
ND	FN	TN

TP — Actual → Diabetes, Model → Diabetes } Correct Scenario

TN — Actual → N.D, Model → N.D } Correct Scenario

FP — Actual → ND, Model → Diabetes } Wrong Prediction (X Blunder)

FN — Actual → Diabetes, Model → N.D } Blunder'

F-Beta Score

$$F = (1+\beta^2)\left(\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}\right)$$

\# If FP and FN, both are important

than $\beta = 1$

$$\text{F-1 Score} = (1+1^2)\left(\frac{P \times R}{P+R}\right) = 2\left[\frac{PR}{P+R}\right]$$

"Harmonic Mean"

If, FP is more imp than FN

$\beta = 0.5$

$$F = (1+0.25)\left(\frac{PR}{P+R}\right)$$

If FN is more important than FP

then $\beta = 2$

$$F\text{-}2\ \text{Score} = (1+4)\left(\frac{PR}{P+R}\right) = 5\left(\frac{PR}{P+R}\right)$$

———x———