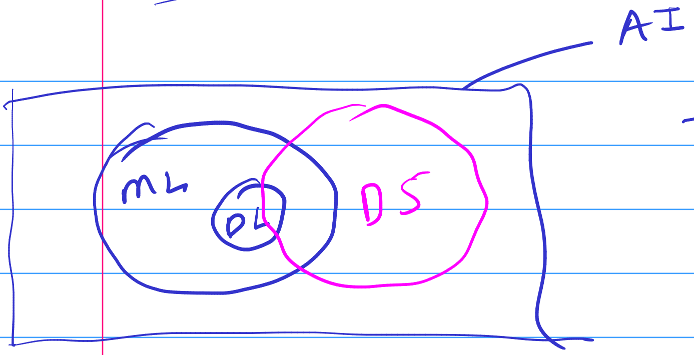
AI - ML - DL & DS



→ Aim → to create an AI

AI → It can do its task without human intervention.

ML → it is a subset of AI which use stats tools, analyzing data, visualizing data, forecasting & Prediction using the data

DL → It is a subset of ML - which is use to mimic human brain, It can train itself - we use Neural Networks in DL

DS → it is a part of everything. focus on Domain & end goal is AI-app.

Machine learning

- Supervised ML (lable data)
  - ① classification
  - ② Regression
- unsupervised ML (No lable data)
  - ① Clustering
- Semisupervised
- → Reinforcement learning

Dataset →

no. of features X $x_1\ x_2\ x_3\ x_4$ ...	O/P Feature
Independent feature	dependent feature (lable data)

① Regression

(output always in Continious value)

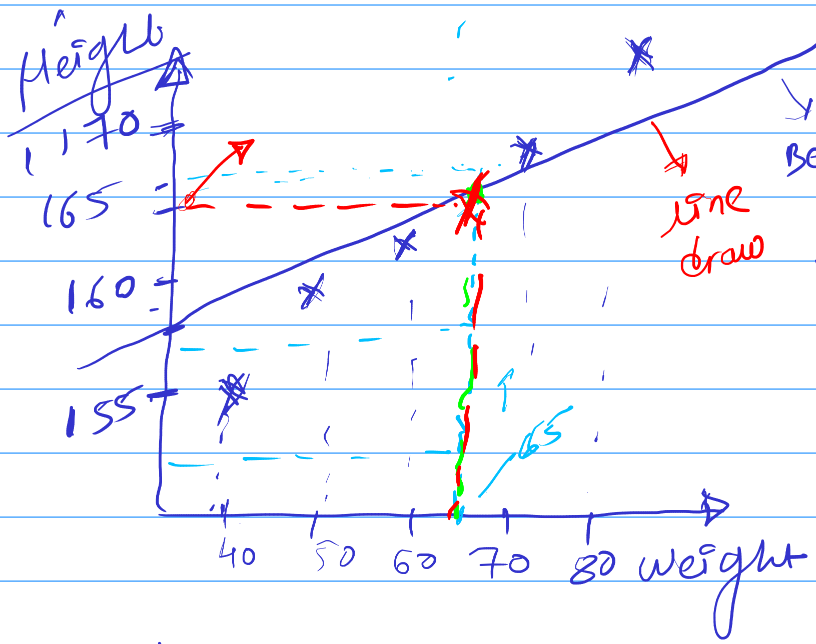
② classification

output always in categorical value or descrete value.

Regression Algo

Simple Linear Regression

Regression → Its all about to draw a best fit line.



i/p	o/p
x	y
weight	Height
sq feet	Prise

linear Algebra

line = mx+c

y = mx+c

Train data

weight	Height
40 →	155
50 →	159
60 →	162
70 →	167
80 →	172

Train data

new data test

	Height
90 kg weight	-
65	?

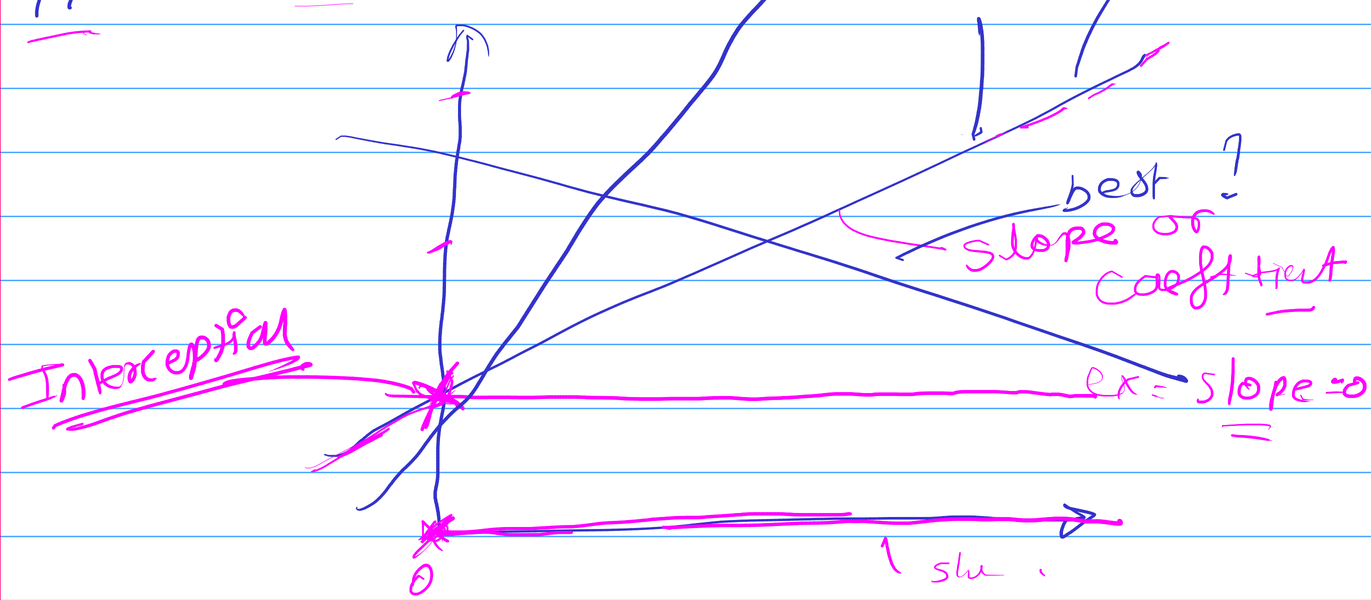
Aim → to draw a best fit line

$y = mx + c \qquad \hat{y} = \beta_0 + \beta_1 x \qquad h_\theta(x) = \theta_0 + \theta_1 x$

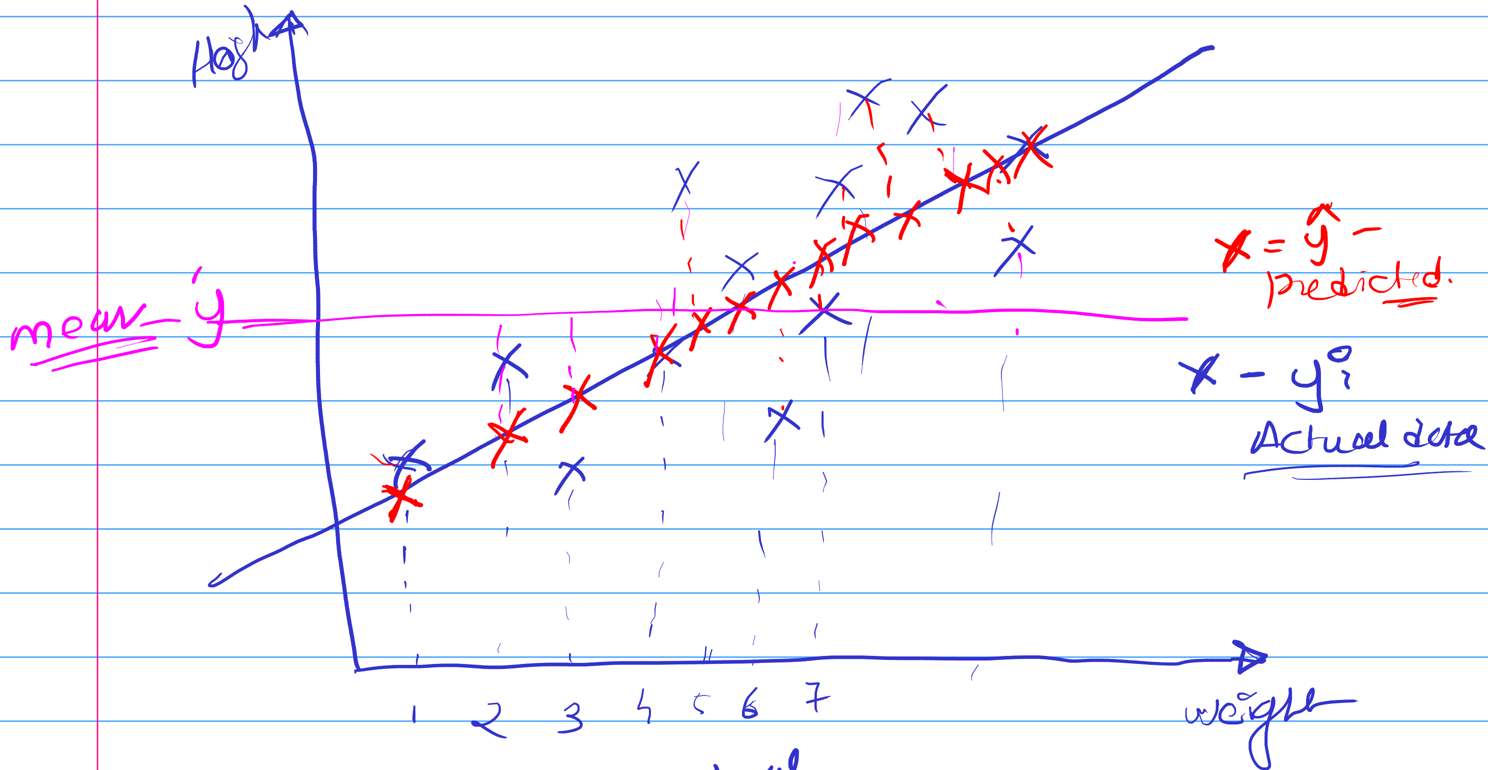
$\hat{y}$ → prediction

β_0 → Interception

β_1 → Coeffitient / slope



min-error = best fit line

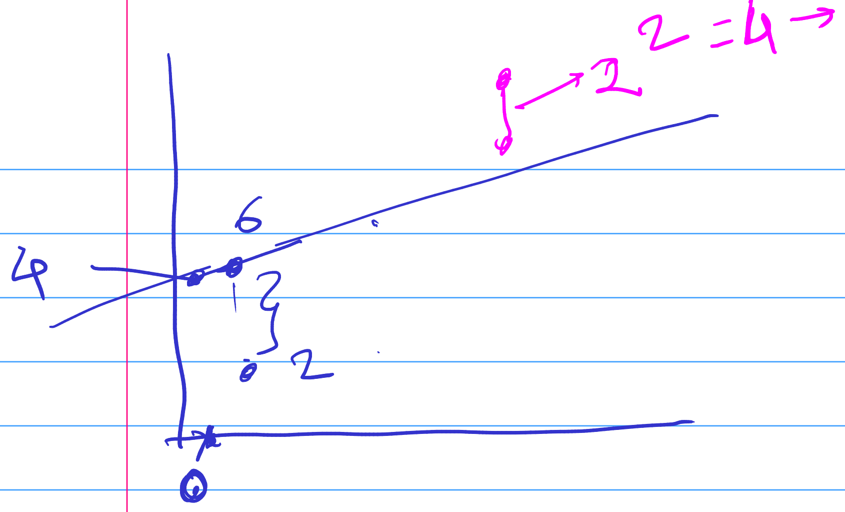


Actual

Residual error $= \sum (y_i - \hat{y})^2$ → prediction

Regression error $= \sum (\hat{y} - \bar{y})^2$ → mean

Total sum $= \sum (y_i - \bar{y})^2$



$(y^i - \hat{y})^2$

$(2-6) = (-4)^2 =$

$(0-4) = (-4)^2$

Linear Regression → Performance Matrix

$R^2 = \frac{\text{Regress}}{\text{total}}$

$\boxed{\text{Approx} = 1 - \frac{\text{residual error}}{\text{total error}}} = 1 - \frac{y^i - \hat{y}}{y^i - \bar{y}}$

$R^2 \approx 1$ $\quad$ $R^2 = 0.70$ − 70% accuracy.

$R^2 = 1$ → overfitting.

Only 1 independent feature

ex = House price | Age old | sq. feet Area | Bedrooms | location

$R^2 \downarrow$ $\quad$ Adjusted R^2 → for multiple i/p.

How to find best fit line ? → Cost function → Squared error function

error = $(y_i - \hat{y})$ = Residual error

→ only for 1 point

All data point = $\Sigma(y_i - \hat{y})$

MSE
mean squared error

$$= \frac{\Sigma(y_i - \hat{y})^2}{n}$$

= disadvantage Robust to outliers

Mean Absolute error
MAE

$$= \frac{\Sigma|y_i - \hat{y}|}{n}$$

⇒ Conversion taking more time

Root mean squared error (RMSE)

$$= \sqrt{\frac{\Sigma(y_i - \hat{y})^2}{n}}$$

Cost function help us to find out best fit line

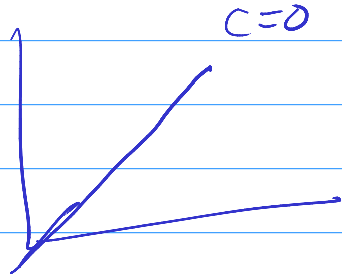
$J \rightarrow$ Cost function $\longrightarrow$ minimize the error

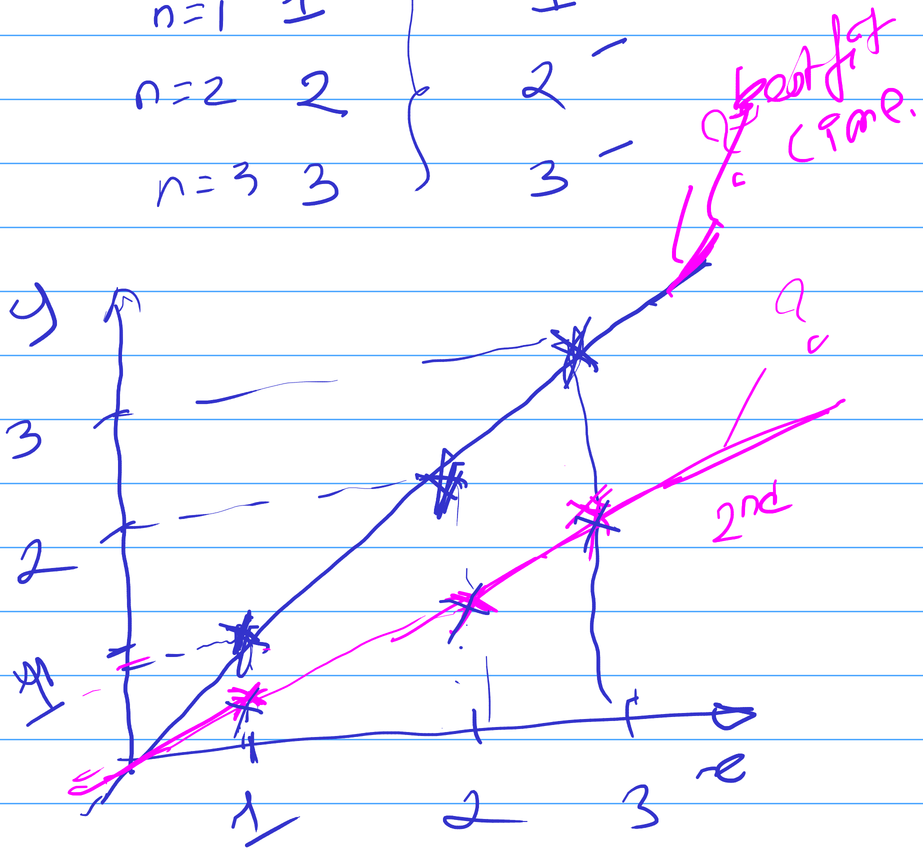
$y = mx + c$

$c = 0$

ex = data point

	x	y
$n=1$	1	1
$n=2$	2	2
$n=3$	3	3





$\hat{y} = mx + c \rightarrow = 0$ consider

$\hat{y} = mx + c$, $c \rightarrow 0$

$m = 1$ =

$\hat{y}(x=1) = 1 \times 1 = 1$

$\hat{y}(x=2) = 1 \times 2 = 2$

$\hat{y}(x=3) = 1 \times 3 = 3$

$m = 0.5$

$\hat{y}(x=1) = 0.5 \times 1 = 0.5$

$\hat{y}(x=2) = 0.5 \times 2 = 1$

$\hat{y}(x=3) = 0.5 \times 3 = 1.5$

$m = 1.5$ =

$$J(m) = \frac{1}{2n} \sum (\hat{y} - y_i)^2$$

Cost function

total no. of data point

$n = 3$

$$J(m) = \frac{1}{2 \times 3} \left((1-1)^2 + (2-2)^2 + (3-3)^2 \right)$$

$$= \frac{1}{6} (0 \; 0 \; 0) = \boxed{0}$$

$J(m) = 1$

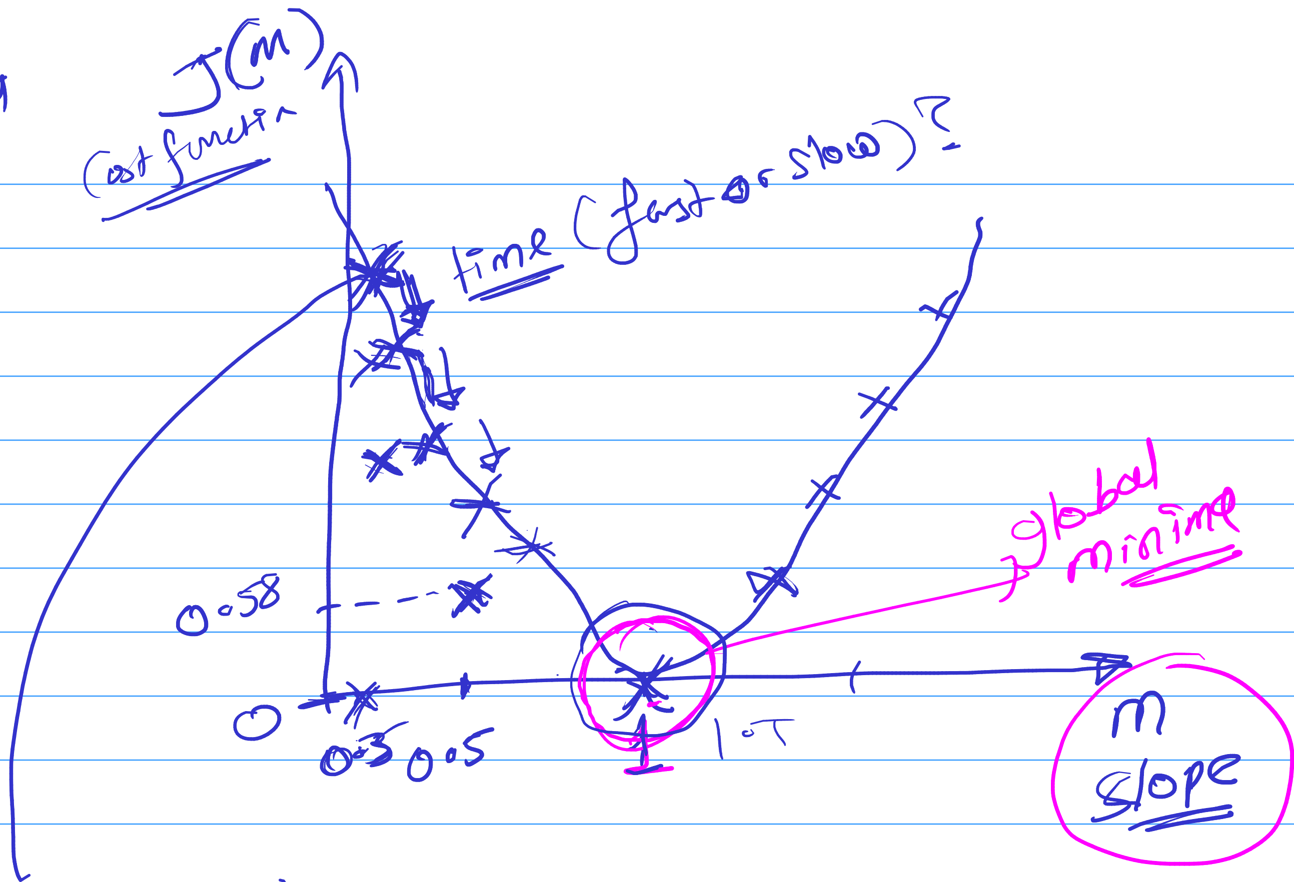
$$J(m=0.5) = \frac{1}{2 \times 3} \left((1-0.5)^2 + (2-1)^2 + (3-1.5)^2 \right)$$

$$= \frac{1}{6} \left((0.5)^2 + (1)^2 + (1.5)^2 \right)$$

$0.25 + 1 + 2.25$

$$\boxed{J(m=0.5) = 0.58}$$

Global minima draw



$m = 0.02 \rightarrow 0.03$

$\alpha \rightarrow$ learning Rate

$$m_{new} = m_{old} - \alpha \frac{d(J)}{dm} \rightarrow \text{slope}$$

x^2

$$\frac{d}{dx} x^2 = 2x$$

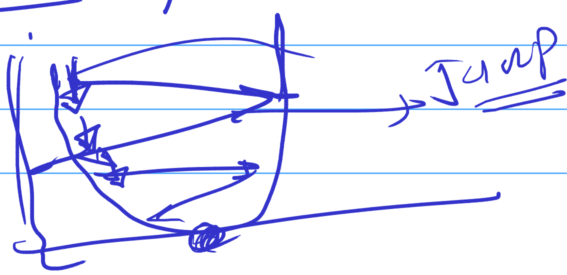
weight

$(\hat{y} - y_i)^2$

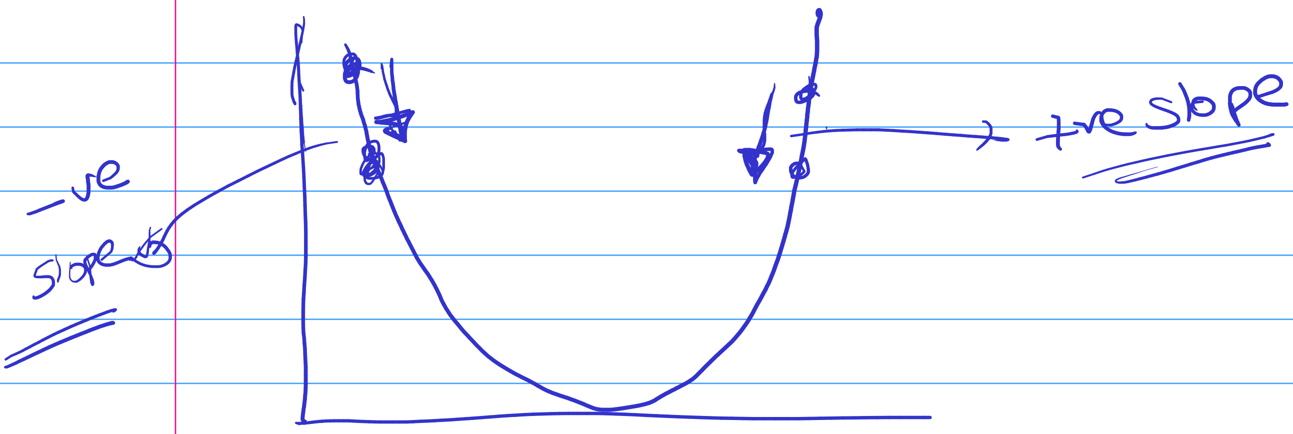
$\alpha \rightarrow$ minimum ex = 0.01

α – maximum

jumping



$$m_{new} = m_{old} - \alpha \frac{d}{dm}(J)$$



+ve slope = decrease the slope value (m)

-ve slope = increase the slope (m) value.

Task → ① Start with bo, b1 (m1, m2)

② keep changing bo & b1 value to reduce cost function

③ until we reached to global minima

④ → Conversion Theorem ⇒

$$\left\{ m_{new} = m_{old} - \eta \frac{d(J(b_0, b_1))}{dm_{old}} \right\}$$