# Simple Linear Regression

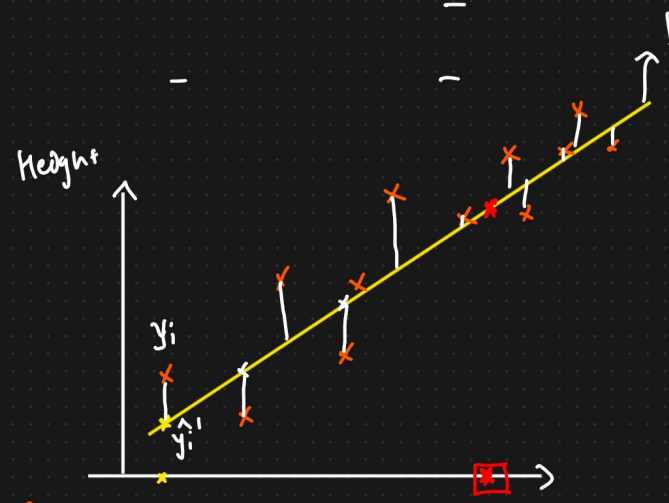
Supervised ML → Regression — O/P → Continuous
→ Classification — O/P → Binary, multiclass categories

Dataset

Independent feature ↑ | Dependent feature ↓

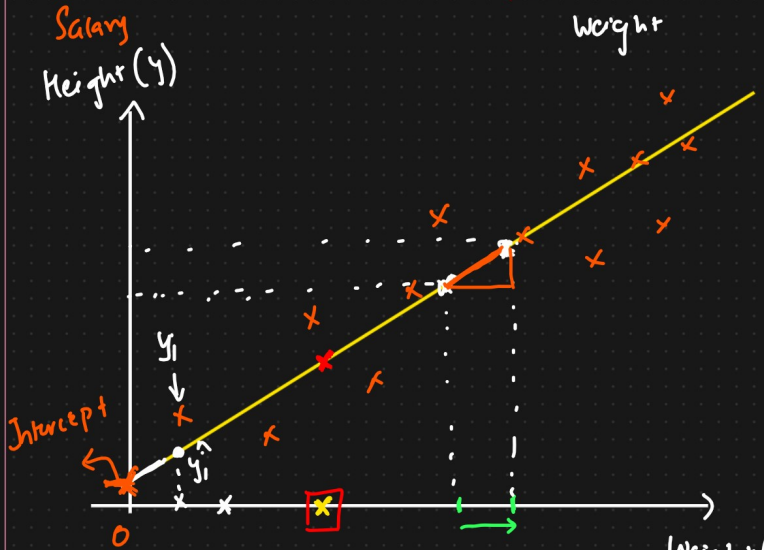
| Weight | Height |
|---|---|
| 74 | 170 |
| 80 | 180cm |
| 75 | 175.5cm |
| – | – |
| – | – |

TRAIN → Model → Height
New Weight →




$\sum$ Error → Minimal wt

Geometrical Intution




$\hat{y} = mx + c$

$\hat{y} = \beta_0 + \beta_1 x$

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

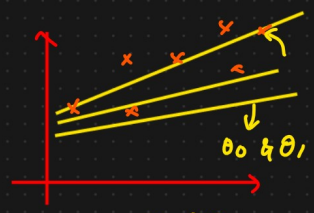
$\theta_0$ = Intercept

$\theta_1$ = Slope or coefficient

$x_i$ ⇒ data points

Predicted

Error $\{y_i - \hat{y}_i\}$




Error wt $\theta_0$ & $\theta_1$

⇓

Optimization Process

## Cost function

actual ⇑ predicted ⇑

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

⇒ **Mean Squared Error**

$n$ = no. of datapoints

$y_i$ = Actual value

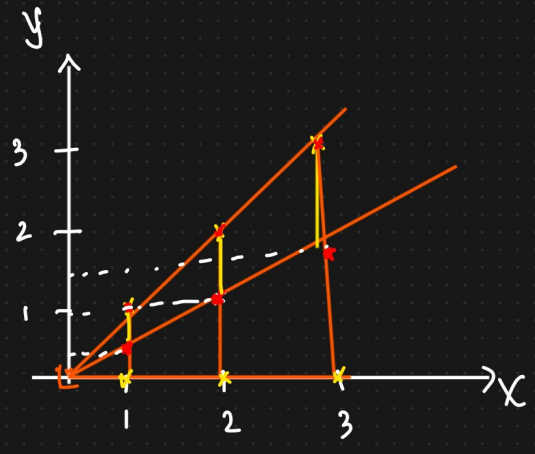
$h_\theta(x)$ = predicted value

## Final Aim {To get Best fit line}

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

$\theta_0, \theta_1$

## Optimization {Minimize the cost function}




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

$\theta_0 = 0$

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

Dataset

| | x | y | $h_\theta(x)$ |
|---|---|---|---|
| → | 1 | 1 | 1 |
| | 2 | 2 | 2 |
| | 3 | 3 | 3 |

| | Let $\theta_1 = 1$ | Let $\theta_1 = 0.5$ | Let $\theta_1 = 0$ |
|---|---|---|---|
| $x=1$ | $\Rightarrow h_\theta(x) = 0 + 1(1)$ <br> $h_\theta(x) = 1$ | $h_\theta(x) = 0 + 0.5(1)$ <br> $= 0.5$ | $h_\theta(x) = 0 + 0(1)$ <br> $= 0$ |
| $x=2$ | $h_\theta(x) = 0 + 1(2)$ <br> $h_\theta(x) = 2$ | $h_\theta(x) = 0 + 0.5(2)$ <br> $= 1$ | $h_\theta(x) = 0 + 0(2)$ <br> $= 0$ |
| $x=3$ | $h_\theta(x) = 3.$ | $h_\theta(x) = 0 + 0.5(3)$ <br> $= 1.5$ | $h_\theta(x) = 0.$ |

*

## Cost function

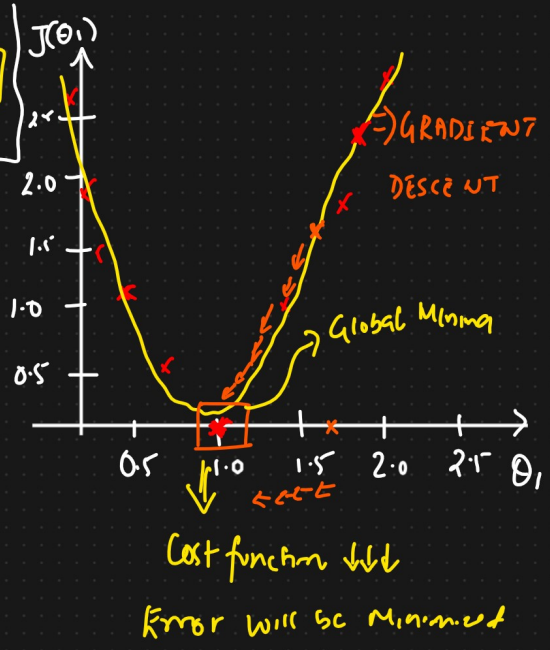
$\theta_0 = 0 \Rightarrow h_\theta(x) = \theta_1 x_i$ $[\theta_1 = 1]$

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

$n = 3$

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

$$J(\theta_1) = 0$$




## Cost fn

$\theta_1 = 0.5$

$$J(\theta_1) = \frac{1}{3}\left[(1-0.5)^2 + (2-1)^2 + (3-1.5)^2\right]$$

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

$$J(\theta_1) = 1.16$$

## Cost fn :

$\theta_1 = 0$

$$J(\theta_1) = \frac{1}{3}\left[(1-0)^2 + (2-0)^2 + (3-0)^2\right]$$

$$= \frac{1}{3}\left[1 + 4 + 9\right]$$

$$J(\theta_1) = \frac{14}{3} = 4.66$$

## Convergence Algorithm {optimize the change of $\theta_j$}

Repeat until Convergence

{

$j = 0, 1$

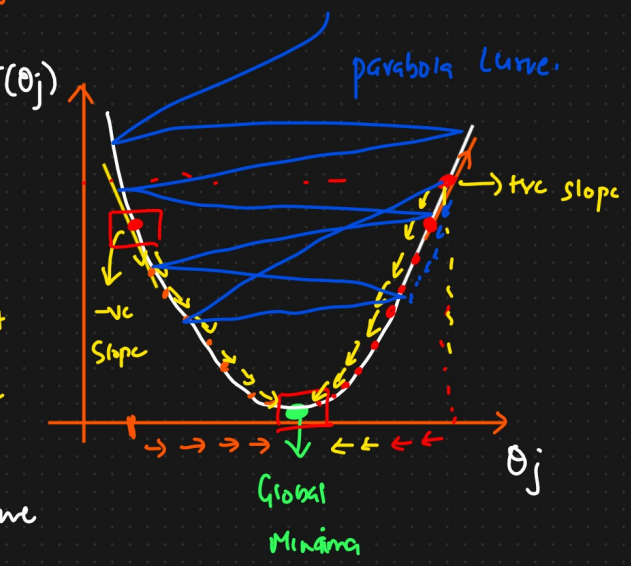
$$\theta_j := \theta_j - \alpha \left[ \frac{\partial J(\theta_j)}{\partial \theta_j} \right]$$

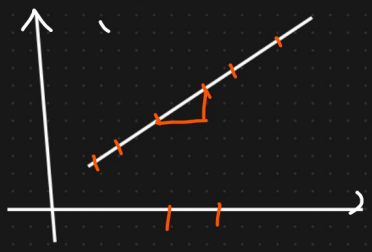
$\alpha$ → learning Rate

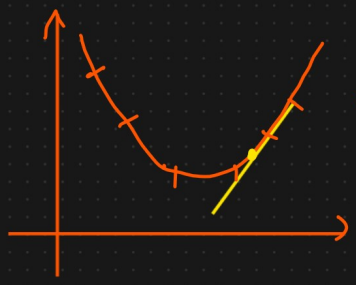
$\frac{\partial J(\theta_j)}{\partial \theta_j}$ → Slope at a point → Derivative

$\alpha = 0.01$

}





⇒ Derivative Slope of a point

$\theta_1 = \theta_1 - \alpha$ (+ve) ← Slope

$\theta_{1new} = \theta_{1old} - \alpha$ (+ve)

$\theta_{1new} << \theta_{1old}$ ⇒ $\theta_1$ is getting Reduced

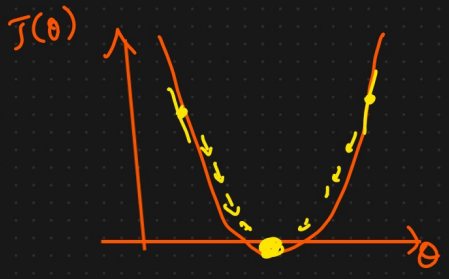
$\theta_{1new} = \theta_{1old} - \alpha$ (-ve)

learning Rate decides the convergence Speed

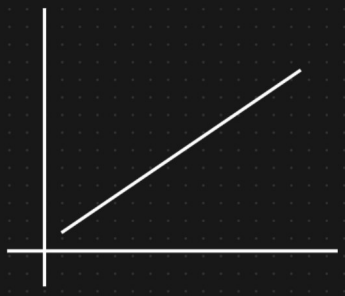
$\theta_{1new} = \theta_{1old} +$ (+ve)

$\theta_{1new} >> \theta_{1old}$

## Conclusion




## GRADIENT DESCENT



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

## Convergence Algorithm

Repeat until convergence

{

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

}

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