INTRODUCTION

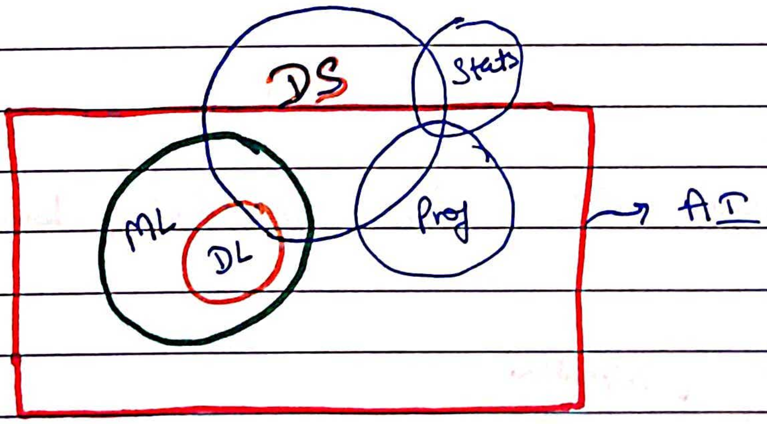
→ AI vs ML vs DL vs DS

AI → creating an application where it performs all its tasks without human intervention. eg. Recommendation in Amaz

ML → Subset of AI. Provides us stats tools to explore visualize, Analyze & perform predictions and other tasks; with the help of data.

DL → Subset of ML. Mimic the human Brain.

DS → Everything is involved. An all rounder.
Stats + Programming + Algebra



→ Machine Learning & Deep Learning

Supervised

Regression

- Linear Regression
- Polynomial
- SVR
- Decision Tree
- Random Forest
- Xgboost
-

Classification

- Logistic Regression
- SVM
- Decision Tree
- Random Forest
- Naive Bayes
- KNN

Unsupervised

Clustering Algorithms

- DB Scan
- K-means
- Hierarchical
- SilChoutte Scoring

We use unsupervised ML when we don't know O/P (output)

Supervised

independent → Dependent feature

Degree	Exp	Salary
-	7	50K
-	2	70K
-	.	.
-	:	:

} Regression

My output feature

No. of hrs Playing	No. of hrs. studying	Pass/ fail
9	1	0
7	2	0
2	5	1

Classification Prob

whenever our output feature is a continuous value →
REGRESSION

whenever our output feature is a categorical value →
CLASSIFICATION

Unsupervised

There is no dependent feature so we use clustering to find some important feature to target.

Eg. Age	Salary	Spending Score (1-10)
24	70K	1
26	100K	9
—	—	—
21	20K	9
25	120K	2

Now let say I wanna sell a Product & give discounts on the basis of Spending Score. Since Spending Score is not directly related to salary.

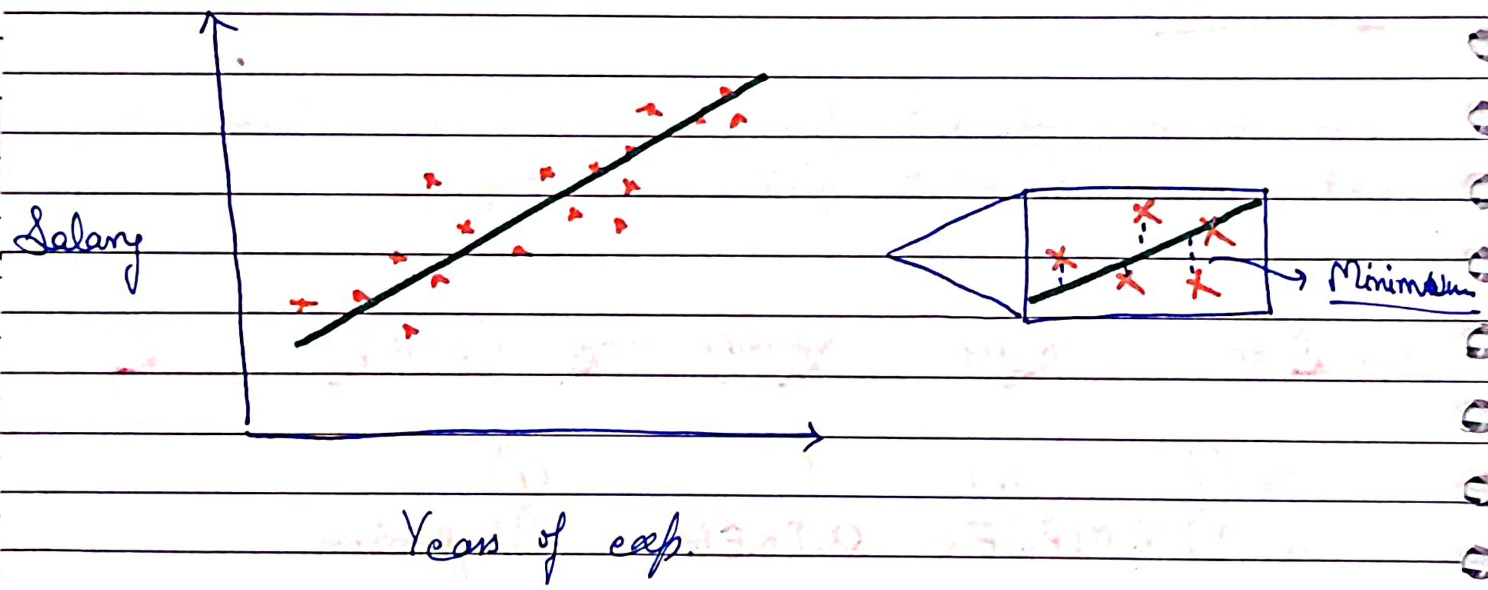
↳ Here we use Clustering to sort data in groups.

Categorizing ✓

Simple Linear Regression

SLR comprise of 1 independent & 1 dependent feature.

eg. Creating a model with Years of Exp. and salary.
To predict the salary based on input year.



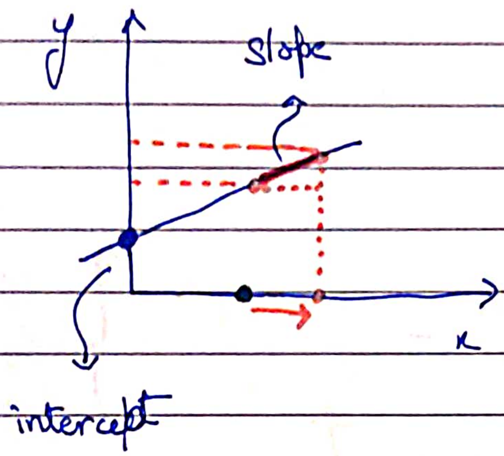
→ Based on this Training Dataset, it will try to find out BEST FIT line, such that the summation of distance of data points from the line is minimal.

Best fit line is the Equation of a straight line

↓

$$y = mx + c$$

Research Paper equation $\Rightarrow$ $\boxed{h_\theta(x) = \theta_0 + \theta_1 x}$



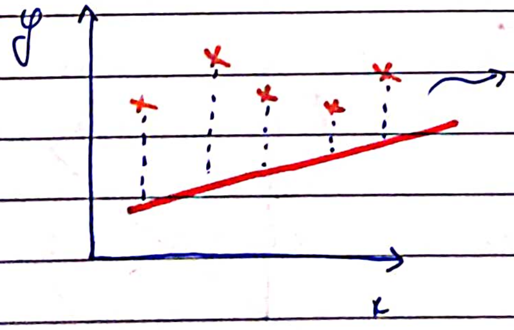
intercepts
(when x is 0, where line meets y axis)

Slope
(when there is unit movement in x axis, what is " in y-axis)

$\therefore$ θ_0 & θ_1 directly impacts Best fit line.

The process of changing θ_0, θ_1 to make line the best fit, its known as TRAINING of THE MODEL.

Let say our initial best fit line is like this....



We can see the errors are quite huge. We need to minimize the errors!

Q How to minimize the error?

A We need to write a equation to minimize error

↳ **Cost Function**

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x)^{(i)} - y^{(i)} \right)^2$$ } Mean Square Error

m = All data points ↑ Predict ↑ Actual ↑

Final Aim

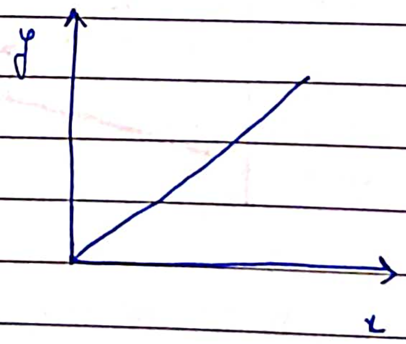
Minimize $J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x)^{(i)} - y^{(i)} \right)^2$

by changing θ_0 & θ_1.

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

Let us consider $\theta_0 = 0$

Now, $\boxed{h_\theta(x) = \theta_1 x}$



$\theta_0 = 0$

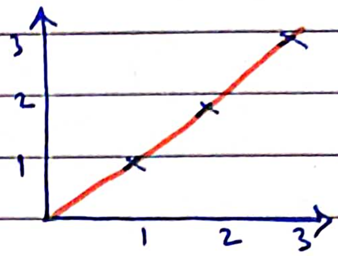
- Now let say $\boxed{\theta_1 = 1}$

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

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

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

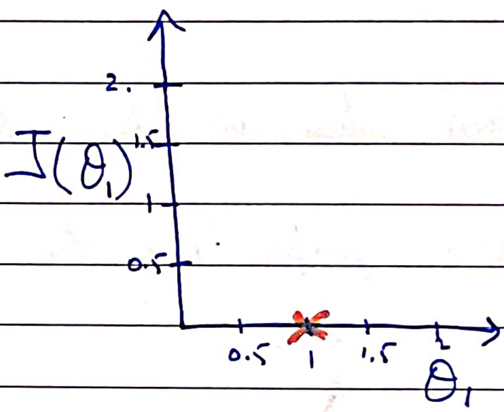
$h_\theta(x) = 3 \qquad x=3$



$$J(\theta_1) = \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x)^{(i)} - y^{(i)}\right)^2$$

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

$$= \underline{\underline{0}}$$



$J(\theta_1) = 0$

$\theta_1 = 1$

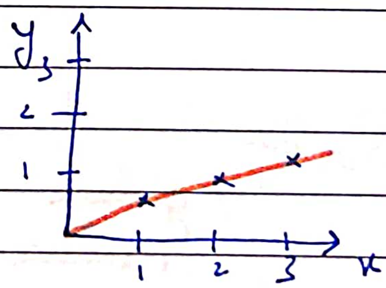
- Now let say $\boxed{\theta_1 = 0.5}$

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

$h_\theta(x) = 0.5 \qquad x=1$

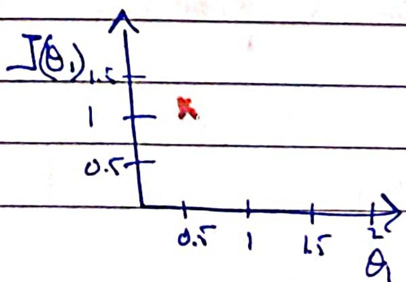
$h_\theta(x) = 1 \qquad x=2$

$h_\theta(x) = 1.5 \qquad x=3$



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

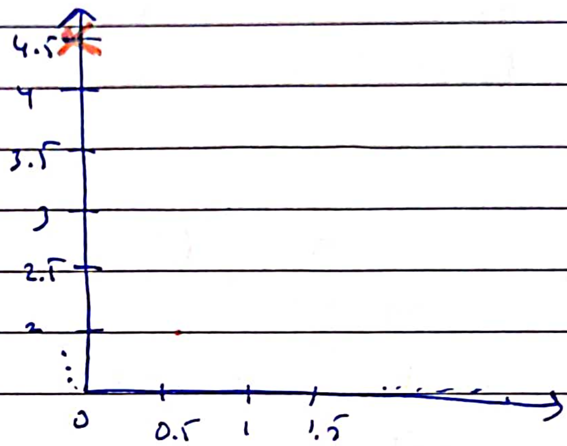
$$= 1.16$$



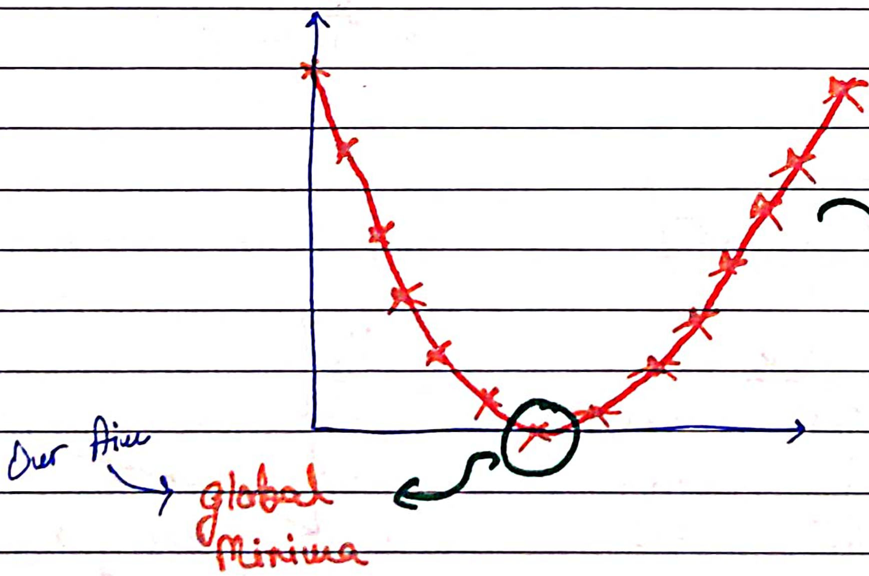
• Now let say $\boxed{\theta_1 = 0}$

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

$= 4.6$



→ Now when we keep changing value of θ_1, we get a graph like this



★★★★★

GRADIENT DESCENT CURVE

★★★★★

→ Super Important graph to understand.

We should try to come to Global Minima.

★★★

How to effectively get Global Minima?

★ CONVERGENCE ALGORITHM

↳ Optimize the changes of θ_1 values.

Repeat until Convergence

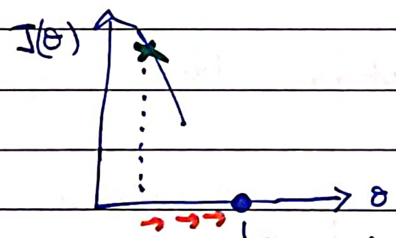
learning rate

derivative

$$\{ \quad \theta_j = \theta_j - \boxed{\alpha \boxed{\frac{\partial}{\partial \theta_j} J(\theta_j)}} = \text{slope} \quad \}$$

★★★ Now when we are here,

↳



↳ global Minima

we need to increase the value of our θ_j in order to get to global Minima.

↳ Since Right side of our line is facing down, so it is a negative slope. & negative slope leads to positive θ_j value.

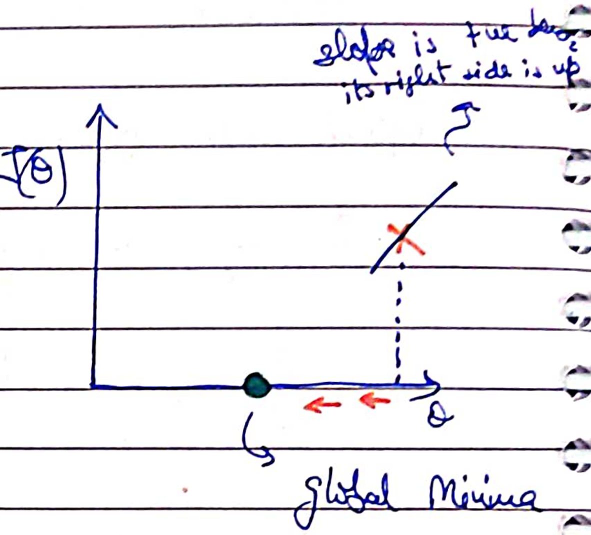
$$\theta_j = \theta_j - \alpha(-ve)$$
$$= \theta_j + \alpha$$

Slope is used to update the θ value!

Reaching Global Minima is our Aim to get Best fit line!

Now if our slope is +ve

↳ To reach global Minima, we need to shift our θ value to negative side.



$$\theta_j = \theta_j - \alpha(+ve)$$
$$= \theta_j - \alpha$$

→ Now what is Learning Rate (α)?

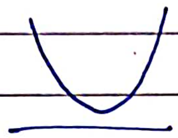
An It is the speed of Convergence. How fast we converge to global Minima.

If α is too small then it will take long.
If α is too big then it will jump.

We select it around $\boxed{\alpha = 0.001}$

At global Minima, the slope is zero.

We can say, **Mean Squared Error** is the reason why we are getting **Gradient Descent Curve**.

$\boxed{\text{MSE}}$ $\Rightarrow$ 

↳ Cost function