Machine Learning

- Field of study that gives computers the ability to learn without being explicitly programmed.

"A computer program is said to be learning from Experience E with respect to some task T and some performance measure P, if its performance on T as measured by P, improves with experience E."

Machine learning Algorithm:

1) Supervised learning
2) Unsupervised learning

Others:-

Reinforcement learning

Supervised learning → output is known

- → Regression
- → classification

Memo No.

Date / /

Regression:-

1) linear Regression
2) Polynomial Regression
3) SVR
4) Decision tree
5) Random forest
6) Xgboost
7) KNN

classification:

1) logistic Regression
2) SVC
3) Decision Tree
4) Random Forest
5) Naive bayes
6) KNN

Memo No.

Date / /

Unsupervised → Output is unknown

clustering

1) DBScan ③ silhouette Scoring

2) Kmeans

Supervised learning

Eg: 1 Independent

dependent feature

Degree	Experience	Salary
BTech	7	100K
PhD	2	90K
-	-	-

o/p → Regression problem

Eg: 2 dependent features

independent features

No. of play	No. of study hrs	pass/fail

o/p → Categorical ⇒ classification problem

Memo No.
Date / /

* Flight Price Prediction → Regression
* Air Quality Prediction → classification
* Buying day of person → classification

Note: Independent features are basically input features.

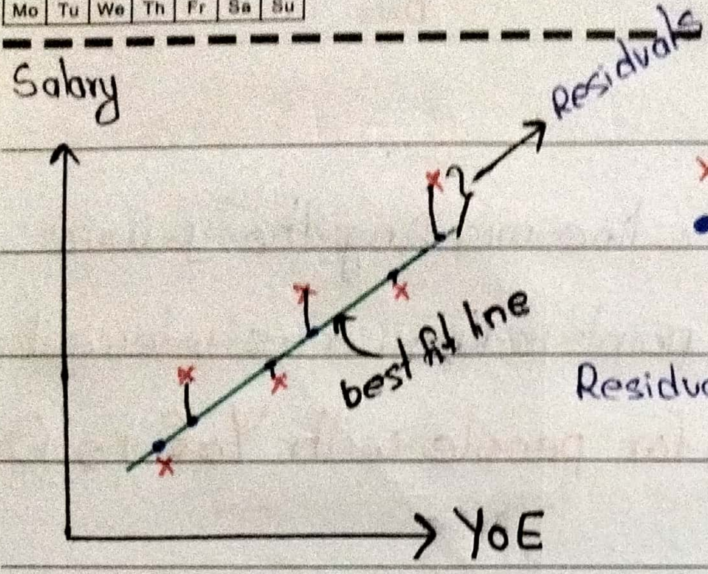
Simple linear Regression
(Univarient Regression)

- One independent feature and one dependent features

Example:

a) To create a model which take input as height and predict weight.

b) To create a model, based on experience, predicting the salary.

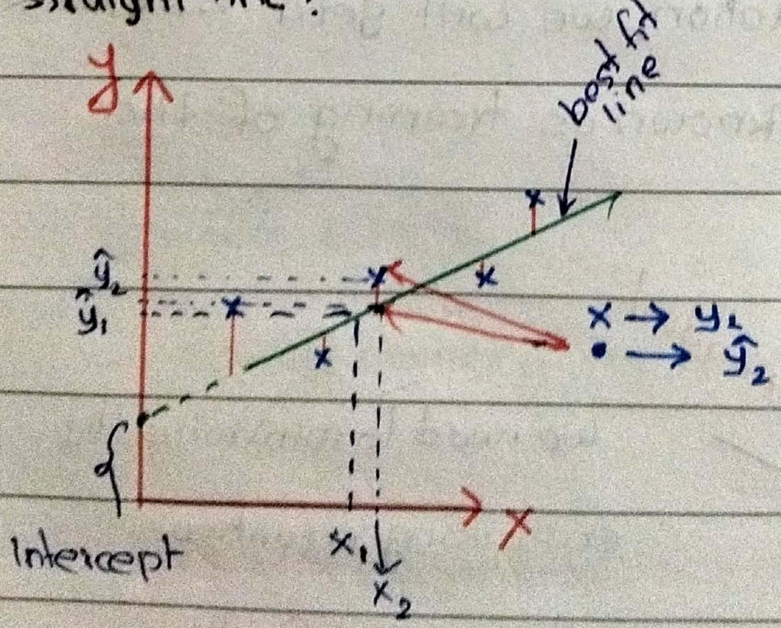


x - datapoint - original
• - datapoint - predicted

Residuals → The difference between real points and predicted points is called residuals or error

Based on the training dataset, it finds the best fit line in a such a way that sum of difference between real points and predicted points should be minimum.

First, we need to understand why are we creating a straight line?



Best fit line is nothing but a equation of straight line

$$y = mx + c$$

or

$$y = \beta_0 + \beta_1 x$$

or

Memo No.
Date / /

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

intercept
slope

Intercept: When $x=0$, the line meeting the y-axis that particular point is known as intercept

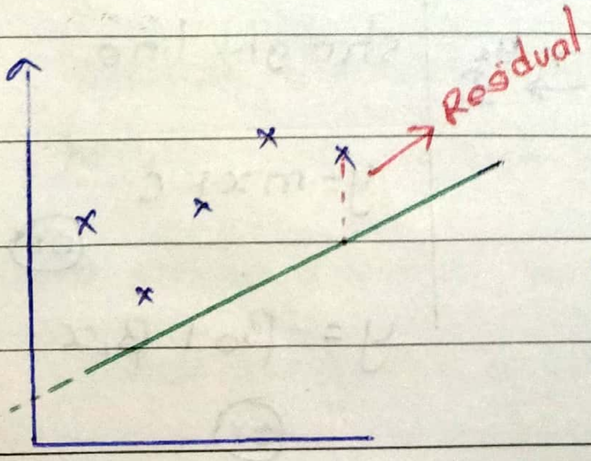
Eg: Default salary for people with YoE is 0.

θ_1 : Slope

↳ With the unit movement in the x-axis what is the movement in the y-axis.

By changing θ_0 and θ_1, best fit line will be change. change $\theta_0, \theta_1 \rightarrow$ line will get changed.

After some iteration, we will get a best fit line which is known as training of the model.



We need to minimize the error using equation called as cost function.

Memo No. ______

Date / /

Cost function

A cost function is used to measure how wrong the model is in finding a relation between the input and output.

→ So, we need to minimize cost function to get the best fit line.

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

mean square error → one of the cost function

* Divided by m to get overage
* Divided by 2 for easy derivative.

Final aim

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

- by changing the values of θ_0 and θ_1

Memo No.

Date / /

Eg:

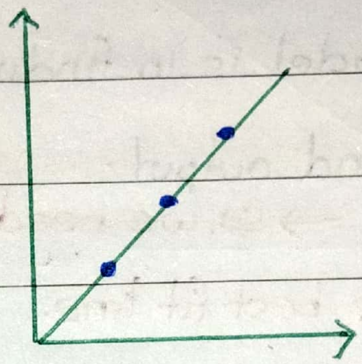
Training dataset

x	y
1	1
2	2
3	3

let us consider,

$\theta_0 = 0$

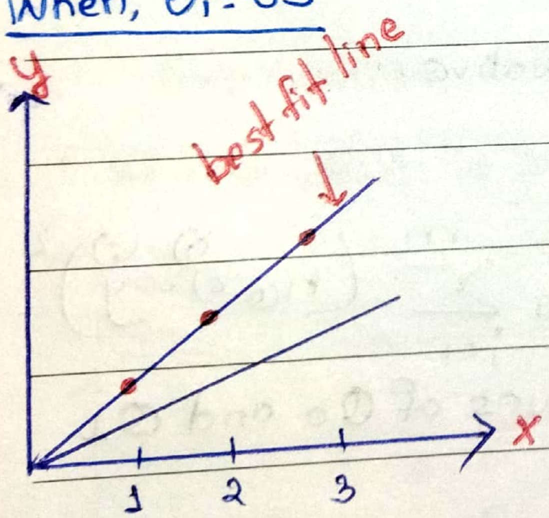
$\therefore h_\theta(x) = \theta_1 x$



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

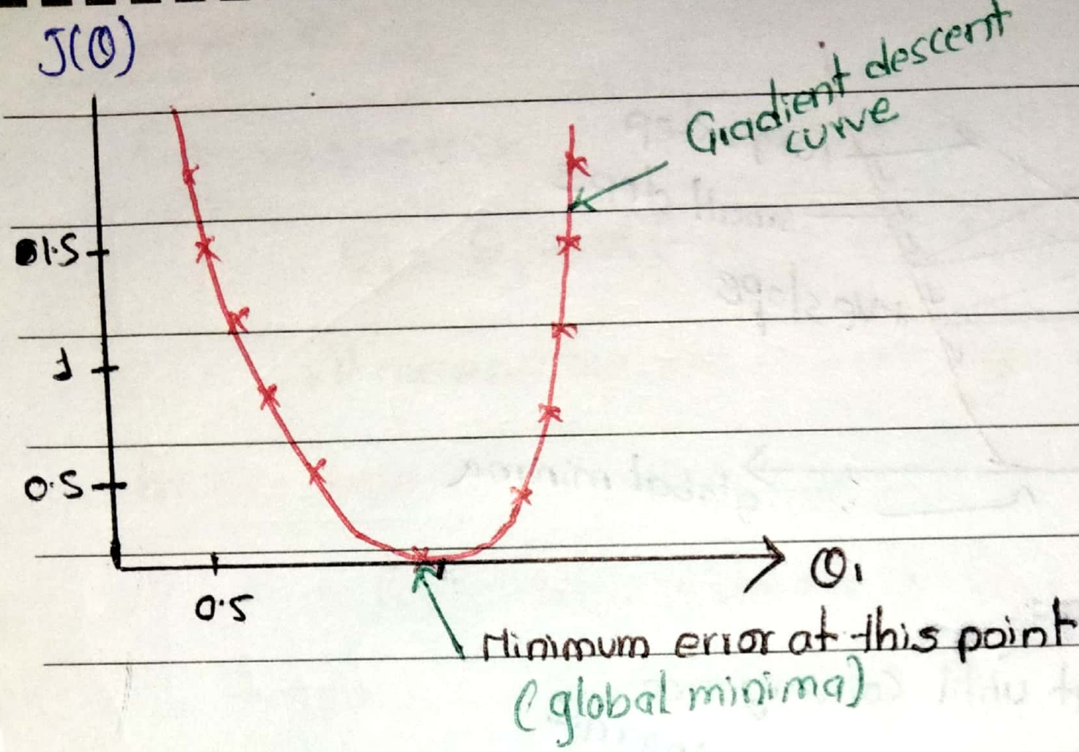
$$= \frac{1}{3}\left[0+0+0\right] = 0$$

When, $\theta_1 = 0.5$



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

$$= 1.16$$



Gradient descent

↳ we have to find the direction in which the error decreases constantly.

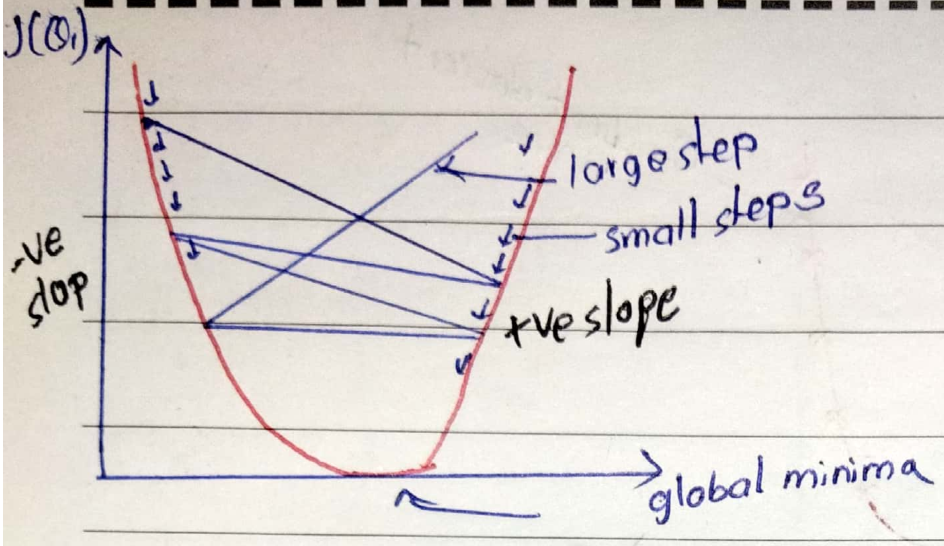
We cannot change Θ value manually, there should be some mechanism to change Θ value to get the global minima. We use convergence algorithm for it.

Convergence Algorithm:

— Optimize the changes of Θ_1 value.

Memo No.

Date / /



Algorithm:

Repeat until convergence:

{

$$\theta_j := \theta_j - \alpha \underbrace{\frac{\partial}{\partial \theta_j} J(\theta_i)}_{\text{slope}}$$

learning rate

}

learning rate(α):

It decides how fast we move down the slope. If alpha is large → you take big steps

alpha is small → you take small step

If α is very high ⇒ you can entirely miss the least error point

If α is very small ⇒ too slow to optimize and waste computational power.

Hence: we need to choose an optimal value of α.

Memo No. ______

Date / /

-Ve or +Ve Slope

for -ve slope :-

$$\theta_j = \theta_j - \alpha(-ve) = \theta_j + ve \text{ value}$$

→ It means we are increasing θ_j

for +ve slope

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

It means we are decreasing θ_j.

Difference between cost function and loss function:

Cost function :- we find error for all points and calculate average of it.

loss function :- we find error at the specified point

loss function: $(h_\theta(x)^i - (y)^i)^2$

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