17/04/23

Machine Learning Day-2

Linear Regression Algorithm

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

Multiple Linear Regression

$h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x + \theta_3 x \cdots \theta_n x$

Mean Square Error (MSE)

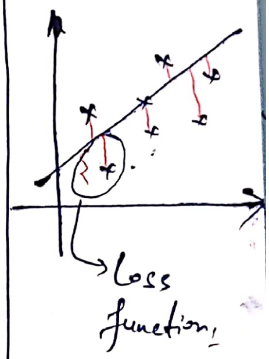
$$J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x)^{(i)} - y^{(i)} \right)^2$$

↑ Cost function

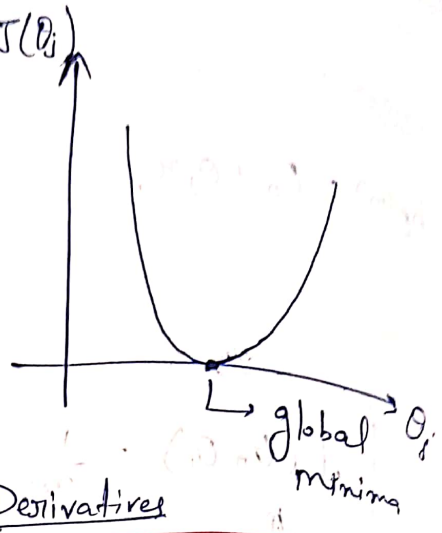
Q: What is the difference b/w Loss function and Cost function.

Soln:

Loss function	Cost function
(*) In case of loss function it will be w.r.t every observation.	(*) In Cost function we will be doing calculations w.r.t all the data points. i.e $i = 1 \cdots m$.
(*) In case of loss function we do the calculation w.r.t each & every point.	(*) Here we calculate the distance b/w MSE w.r.t all the data points. basically we do the summation.
(*) Loss function $= \left(h_\theta(x)^{(i)} - y^{(i)} \right)^2$	(*) Cost function $= \frac{1}{m} \sum_{i=1}^{m} \left(\hat{y}_i - y_i \right)^2$
$L.F = \left(\hat{y}_i - y_i \right)^2$ (Predicted value, Actual value)	(*) Summation of square of each and and every point difference
(*) Single Point difference.	
(*) Calculate the difference of every point independently and then do square of it.	(*) Here after finding the difference and square of it we do the Summation & then avg.



Convergence Algorithm :→



Repeat until Convergence

{

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

where j = 1 to n

}

(θ_0, θ_1)

Find the value of this derivative

i.e $\frac{\partial}{\partial j} J(\theta_j)$

θ_0, θ_1

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

when, $j = 0$ $\quad h_\theta(x) = \theta_0 + \theta_1 x$

$$= \frac{\partial}{\partial \theta_0} \left[\frac{1}{m} \sum_{i=1}^{m} \left((\theta_0 + \theta_1 x)^{(i)} - y^{(i)} \right)^2 \right]$$

$$\frac{\partial}{\partial \theta_0} J(\theta_0) = \frac{2}{m} \sum_{i=1}^{m} \left[(\theta_0 + \theta_1 x)^{(i)} - y^{(i)} \right] * 1$$

when $j = 1$

$$\frac{\partial}{\partial \theta_1} J(\theta_1) = \frac{\partial}{\partial \theta_1} \left[\frac{1}{m} \sum_{i=1}^{m} \left((\theta_0 + \theta_1 x)^{(i)} - y^{(i)} \right)^2 \right]$$

$$\frac{\partial}{\partial \theta_1} (J\theta_1) = \frac{2}{m} \sum_{i=1}^{m} \left(\theta_0 + \theta_1 x \right)^{(i)} - y^{(i)}) \; x[x]$$

Derivatives

$$\frac{\partial}{\partial x} (x^2) = 2x$$

$$\frac{\partial}{\partial x} (x)^n = n x^{n-1} \; (1)$$

$$\frac{\partial}{\partial x} (x+1)^2 = 2(x+1)(1+0)$$

$$= 2(x+1)$$

Simultaneously θ_0 and θ_1 value will change according to this formula.

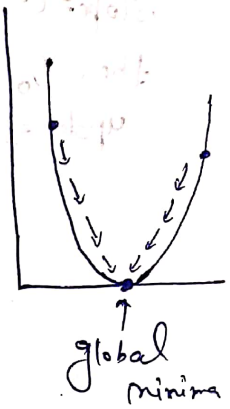
Repeat until Convergence

$$\theta_0 : \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x)^{(i)} - y^{(i)})$$

$$\theta_1 : \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x)^{(i)} - y^{(i)}) x^{(i)}$$

α determines the speed of convergence.

and By Using this the loss will be minimum.



Types of Cost function

(i) MSE (Mean Squared Error)

(ii) MAE (Mean Absolute Error)

(iii) RMSE (Root Mean Square Error)

(i) MSE (Mean Squared Error)

$$MSE = \sum_{i=1}^{n} \frac{(y - \hat{y})^2}{n}$$ ← quadratic eqn.

where $\hat{y} = \theta_0 + \theta_1 x$

↑ Predicted value

We know that,

$(a-b)^2 = a^2 - 2ab + b^2$

i.e II^{nd} to

$ax^2 + bx + c = 0$

↳ this a quadratic equation

→ and we plot the graph of a quadratic eqn it gives bell curve or Convex function.

Advantage and Disadvantage of MSE

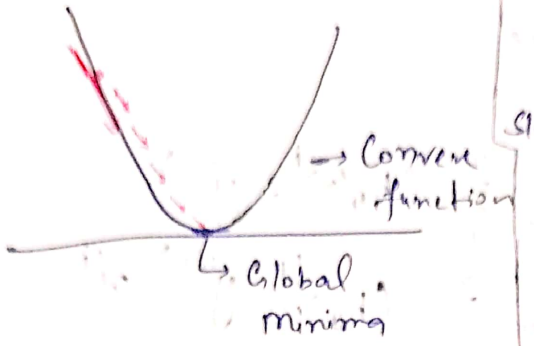
(i) This eqn is differentiable.

& Why do we need differentiability?

= To calculate θ_0, θ_1 and slope.

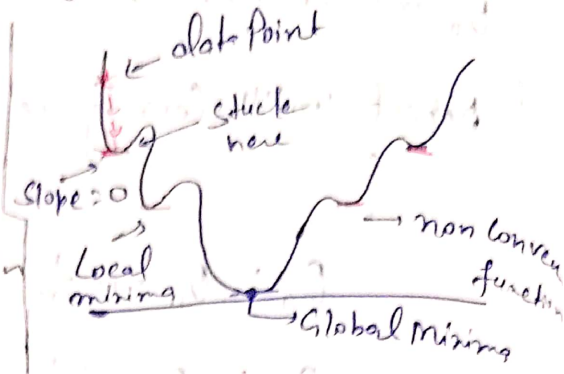
differentiable means the derivatives exists at every point in its domain. Consequently, the only way for derivative to exist is if the function also exists (i.e continious) on its domain. Thus a differentiable fn is also a continious fn.

(ii) This eqn also has only one global minima.



(*) At the point of global minima slope = 0 and if slope is 0 then θ_0 and θ_1 will not get updated

(*) Hence our AIM is to always work with Convex function so that we can get only One global minima.



(*) In this case at Local minima also slop = 0 Hence the θ_0 and θ_1 will not update further and it will get stuck at local minima.

(*) Hence this Scenario will never come w.r.t MSE

(*) Hence We can't use non-Convex function for Cost function.

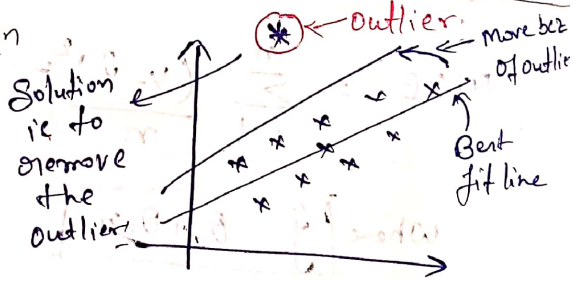
(*) Disadvantage of MSE.

(i) It is not Robust to outliers.

Q: Bez of outlier whether the cost fn will increase or decrease?

Soln

It will Increase by tremendous amount (↑↑↑).



How?

(*) Bez of outlier the error will increase and θ_0 and θ_1 will get adjusted and Since we are Squaring the error the MSE will increase tremendously.

(*) Hence to Solve this problem we need to remove the Outliers.

(*) Hence if we use MSE eqn directly it will not robust to outliers bez there will be bigger Switch bez of this.

Example: dependent feature ← Independent feature

Exp	Salary (In lakhs per annum)
–	–
–	–
–	–
–	–
–	–

Suppose we are trying to predict the Salary based on Experience.

In that case,

Let's calculate $(y - \hat{y})^2$

i.e $(\text{Salary} - \text{Predicted Salary})^2$

$= (\text{lakh})^2$

↑ Here Unit will change from lakh to millions.

(*) Here Error present in $(\text{lakh})^2$ and bcz of that unit is changing and Error is getting Penalized

Eg: $(100 - 110)^2 = 10^2 = 100$

3digit 3digit 3digit

but Suppose

$(999 - 100)^2 = (899)^2 = 8021$

3digit 3digit 3digit 4digit (unit changed)

(*) and Based on that we need to reduce the cost function.

(*) Bcz of changing of unit time Complexity will increase and that's why outlier affect will also happen.

(*) we are Suppose to reduce MSE but here in this case it is increasing tremendously. Hence it is Penalizing the Error i.e changing the Unit. Hence it is not robust to outliers.

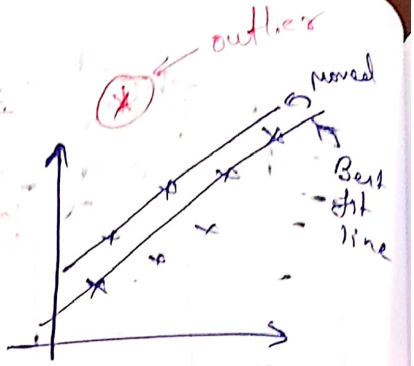
(ii) Penalizing the Error i.e Changing the Unit.

Hence to Solve this problem we remove the outliers.

(2) Mean Absolute Error (MAE)

$$MAE = \frac{1}{m} \sum_{i=1}^{m} |y - \hat{y}|$$

Advantage and Disadvantage of MAE



(1) Advantage of MAE

(i) Robust to outliers.

(ii) It will also be in the same Unit.

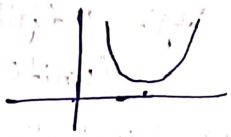
Since here we are not squaring the errors.
Hence bcz of outlier even if best fit line moves it will change little bit only that much difference won't be there.
Hence it is Robust to outliers.

(2) Disadvantage of MAE

(i) Convergence usually takes more time.

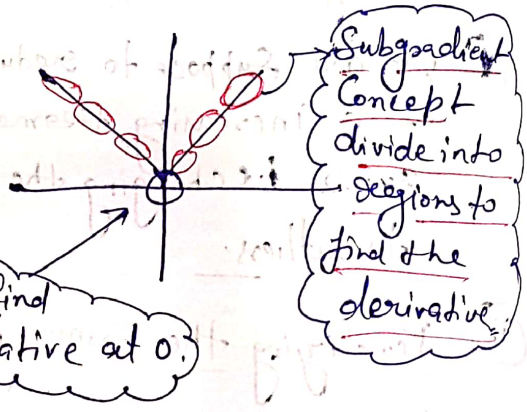
(ii) Optimization is a complex task.

(*) In MSE the graph comes as parabola since it gives quadratic eqn.



(*) But In Case of MAE, we have linear eqn Hence the graph comes in straight line.

(*) Here we find the derivative with the help of Subgradient Concept.



(*) Subgradient basically says that you have to divide into region by region and try to find out the derivative to come near 0.

(*) Divide into parts by parts and find or in Subgradient and then take the derivative.

(iii) Time Consuming.

Assignment

① Huber Loss → Combination of MAE & MSE. { Study yourself } same as MSE

② RMSE → $\sqrt{MSE}$ ⇒ { Read about Advt. & Disadv. } w.r.t. Unit

NOTE

* If there is outlier we use MAE.
  If there is no outlier we use MSE.

Same as MSE → Outliers & Differentiable

diff.

Same graph ∪

Performance Metrics :→

Q. How do we check if the model is good or not in case of Linear Regression.

Loss function :- MAE, MSE, RMSE
Huber Loss

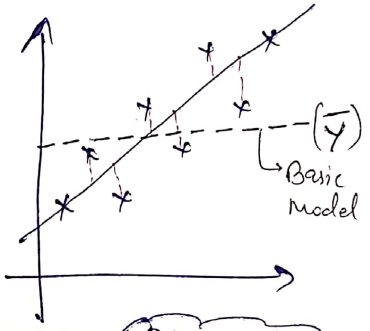
→ for that basically we need to check the performance matrix.

(i) R Squared
(ii) Adjusted R Squared

{ It helps to check the performance of the model.

(i) R Squared :→

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



where SS_{Res} = Sum of Square Residuals.

SS_{Total} = Sum of Square Avg.

$$R\text{-}Square = 1 - \frac{\sum_{i=1}^{n}(Y-\hat{Y})^2}{\sum_{i=1}^{n}(Y-\bar{Y})^2}$$

Low → If model is fitted well numerator will be less since error will be less and denom will be high.

High →

where $\bar{Y}$ = Avg. of Y.

$$= 1 - \frac{\text{Small no.}}{\text{Bigger no.}} \rightarrow \text{Small no.}$$

$$= 1 - \text{Small no.} \quad \text{i.e } \leq 1$$

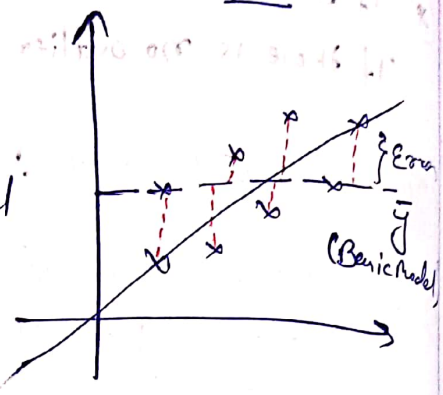
Hence R-Squared value will be always less than 1.

iff. R.Squared = 0.85 → Model is 85% Accurate.

= 0.75 → Model is 75% Accurate.

Q: Why denominator is high?

W.r.t $\bar{y}$ Error will be more since it is not best fitted line. Hence Squaring that will make the Error bigger..



(*) R Squared measures the performance of the Model that you have actually created.

Q: Can R^2 Squared value be -ve?

Sol: If at all R-Squared value is -ve it means Model is very bad. Even bad than the $\bar{y}$.

Bez

$$R\text{-Squared} = 1 - \frac{\text{Big no.}}{\text{Small no.}}$$

If num. is big means Error is big means very very bad model

$$1 - \text{Big no.}$$

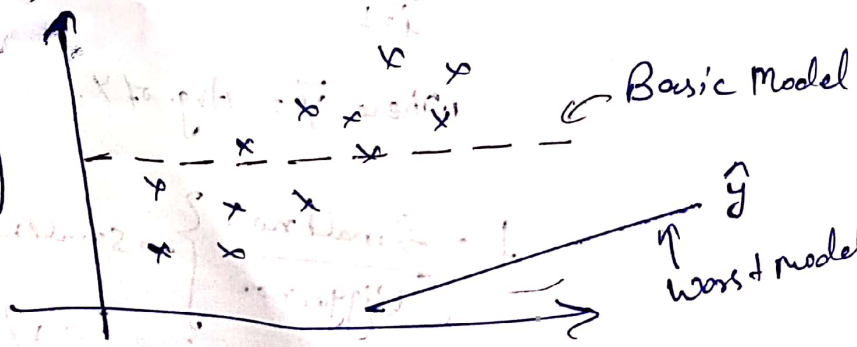
Even Bad than Basic model i.e $\bar{y}$.

$$R\text{-Squared} = -ve \quad i.e\ 1 < 0$$

Eg:-

In this Case

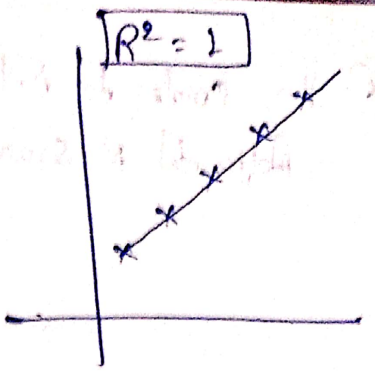
$R^2 = -ve.$



(*) If all the data points are coming over the best fit line in that case $R^2 = 1$

Bcz Error = 0

Hence $R^2 = 1 - 0 = \boxed{1}$



(ii) Adjusted R Squared

Size of house	City Location	No. of bedrooms	Gender	Price
—	—	—	—	—
—	—	—	—	—
—	—	—	—	

no correlation

(*) If we keep adding feature if it is correlated with our o/p value i.e Price then R-Squared value ↑↑ Increases since the model can predict better Price.

(*) But if we add a feature 'Gender' in the above example which is not correlated with price in that case R^2-Squared Price still increases: to remove this problem we use Adjusted - R Squared

R^2	Accuracy.
Size of home	65%
Size of home, City Loc,	75%
Size of house, City Loc, No. of bedroo	88%
Size of home, City Loc, No. of bedroo, Gender	90%

Whenever the features are highly correlated the increase in accuracy happens very quickly.

→ no correlation still slight increase in Accuracy.
↳ But this shouldn't happen since Gender & Price is not correlated

(*) In order to solve this problem we use Adjusted R. Square.

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

Where, N = no. of data Points

P = No. of Independent factors

Feature	R^2	Adjusted R^2	
Size	65%	63%	P=1
Size, City	75%	73%	P=2
Size, City, No. of bedroom	88%		
⋮	⋮	⋮	
⋮, Gender	90%	85%	P=4

It will be always be less than R^2

decrease in accuracy when feature is not Correlated

(*) Adjusted R^2 will always be less than R^2 as we can see from the formula.

(*) In case of Adjusted-R^2 if the feature is not Correlated i.e Gender → Price

not Correlated

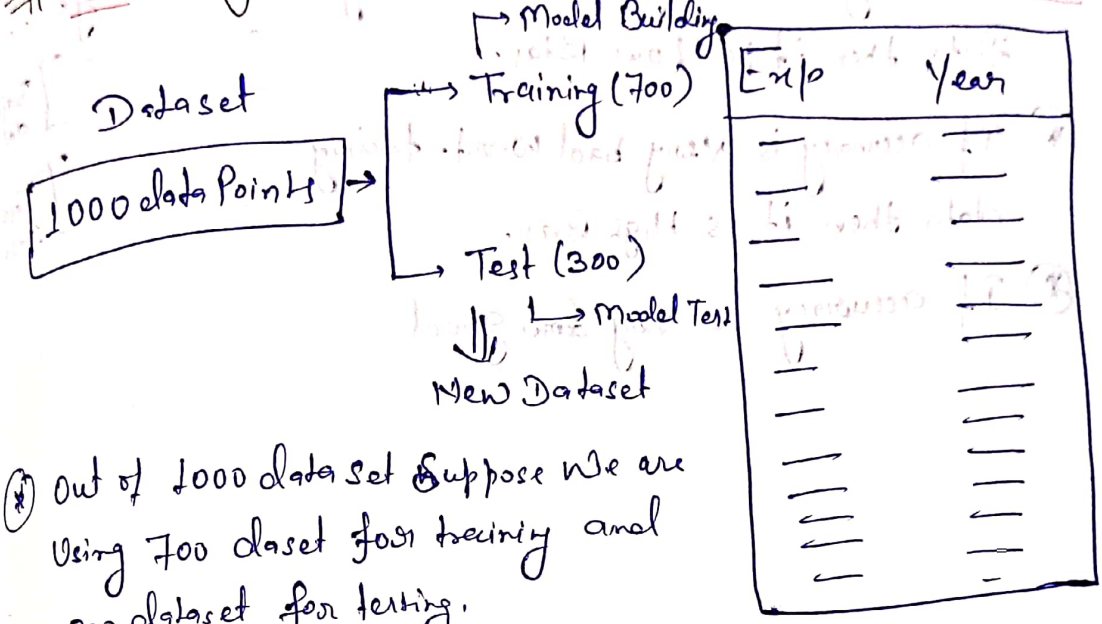
In that case Accuracy decreases.

(*) Means Adjusted-R^2 will not make any impact on Accuracy when unnecessary feature is added.

(*) Adjusted R^2 evaluates the accuracy based on only Important features. Hence it is better than R^2.

(iv) In Case of Performance Matrix we apply both R^2 and Adjusted R^2.

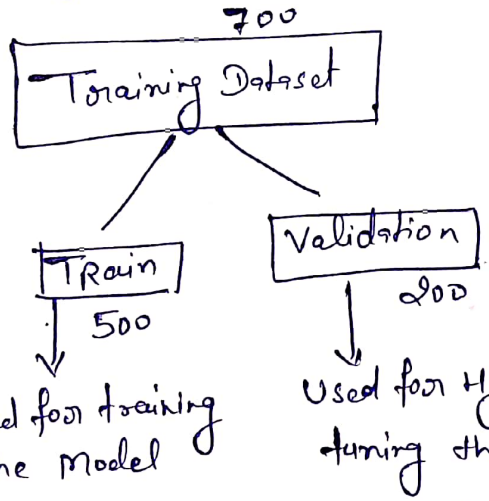
Overfitting and Underfitting (Bias and Variance)



(v) Out of 1000 data set Suppose We are using 700 dataset for training and 300 dataset for testing.

(vi) Training dataset has no idea about that 300 testing dataset. It is completely new for them

⇒



⇒ Suppose We have created one Model

Now,

TRAIN DATA → Very good Accuracy (90%) [Low Bias]

TEST DATA → Very good Accuracy (85%) [Low Variance]

↳ Generalized Model

(vii) The main of Model is to haven Very good accuracy with TRAIN DATA and TEST Data and i.e called Generalized Model.

(*) Training data Accuracy is given by Bias and Test data Accuracy is given by Variance

(*) If accuracy is very good w.r.t training Data then it is Low Bias.

(*) If accuracy is very bad w.r.t. training data then it is High Bias.

(*) If accuracy is very ~~bad~~ good w.r.t test data then it is Low Variance

(*) If accuracy is Very bad w.r.t test data then it is High Variance.

Hence we learn the Train Value

Exp	Age
—	—
—	—
—	—

Train data

Test Data

Overfitting :→

TRAIN DATA → Very good Accuracy [90%] [Low Bias]

TEST DATA → Bad Accuracy [50%] [High Variance]

⇓ This condition is called

Overfitting → Solve using → Hyperparameter Tuning

(*) Overfitting is a situation where in Model gives Very good accuracy w.r.t Train data but gives bad accuracy w.r.t Test data.

Eg:- Ratta Mar, Cramming things without understanding

Q: How do we Solve this Problem?

By Performing Hyperparamete Tuning or We increase the data.

Underfitting :→

TRAIN DATA	→	Model Accuracy is Low [High Bias]
TEST DATA	→	Model Accuracy is Low/High [Low or High Variance]

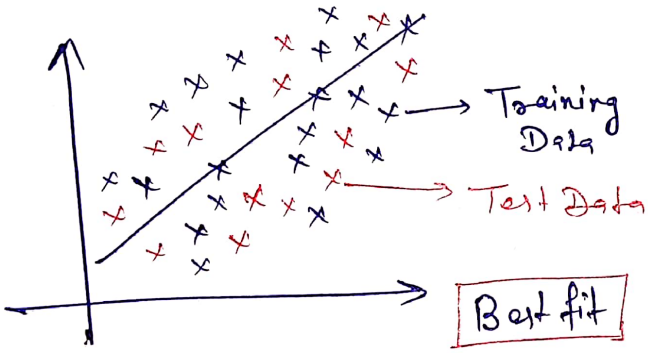
↓

Underfitting

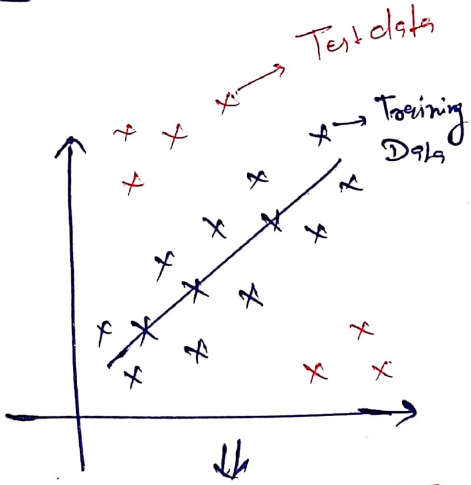
(*) When Model accuracy is very low w.r.t training data and accuracy is High or low w.r.t test data that Situation is called Underfitting.

Eg:- Students who doesn't Study and In exam they just do tukka mar. So result could be either good or Bad i.e unpredictable.

In Case of generalized Model

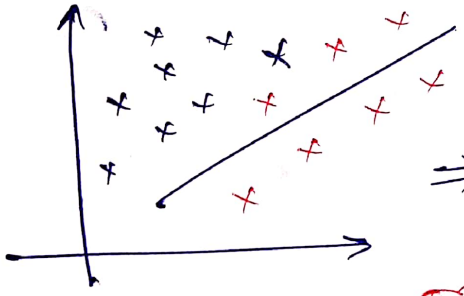


Model is ~~Underfitting~~, Generalized model



Model is Overfitting

Trained well with training data but not with the test data



⇒ Model is Underfitting

Test data may perform good in this case