MACHINE LEARNING CLASS 2. 09/10/22

⇒ Linear Regression Algorithm

$$h_\theta(x) = \theta_0 + \theta_1 x \quad \text{— Simple Linear Regression.}$$

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \cdots + \theta_n x_n$$

⇒ Convergence Algorithm:-

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

$$\boxed{\text{Loss Function}} = \left(h_\theta(x)^{(i)} - y^{(i)}\right)^2 \Rightarrow \left(\hat{y}_i - y(i)\right)^2$$

where $\hat{y}_i$ - Predicted value.
$y(i)$ - Actual value.

Derivatives:-

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

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

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

$\boxed{j=0}$ $$\frac{\partial}{\partial\theta_0} J(\theta_0, \theta_1) = \frac{\partial}{\partial\theta_0}\left[\frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta(x)^{(i)} - y^{(i)}\right)^2\right]$$

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

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

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

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

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

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

Repeat until Convergence :-

{

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

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

}

Cost Functions

[1] MSE [2] MAE [3] RMSE

① **MSE {Mean Squared Error}**

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

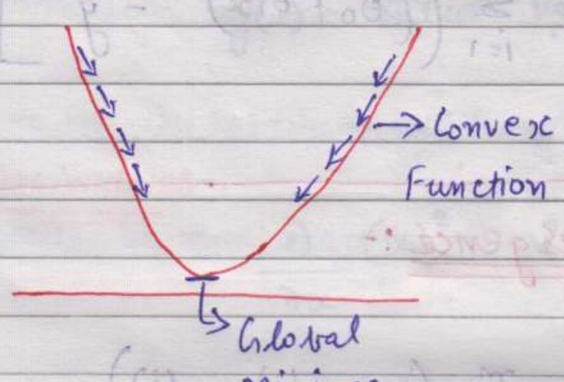
$\hat{y} = \theta_0 + \theta_1 x$ → Predicted value

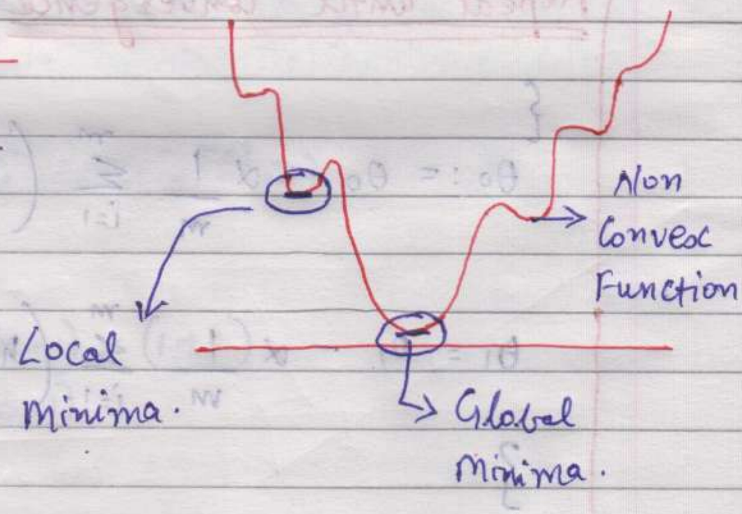
Quadratic Equation $\boxed{ax^2 + bx + c = 0}$

<u>Advantages :-</u>

① This equation is differentiable.

② This equation also has one global Minima.





Disadvantage:-

(1) This is not robust to outliers.

(2) Penalizing the Error {Changing the unit}

(II) MAE {Mean Absolute Error}

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |y - \hat{y}|$$ $|5-3| = |-3| = 3$

Advantages

(1) This is robust to outliers

(2) It will also be in the same unit

Diadvantages:-

(1) Convergence usually takes more time. Optimization is a complex task.

(2) Time Comsuming

(3) Huber Loss It is a combination of MSE and MAE

(IV) RMSE $= \sqrt{MSE}$

- Unit will be same in RMSE
- It is not robust to outliers
- This is also differentiable

Performance Matrix :-

To check wheather the model is good or not, in case of linear regression, we have to see performance matrix

In performance Matrix, we see two things.

1. R Squared Error
2. Adjusted R Squared Error.

① $\boxed{\text{R Squared}}$

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

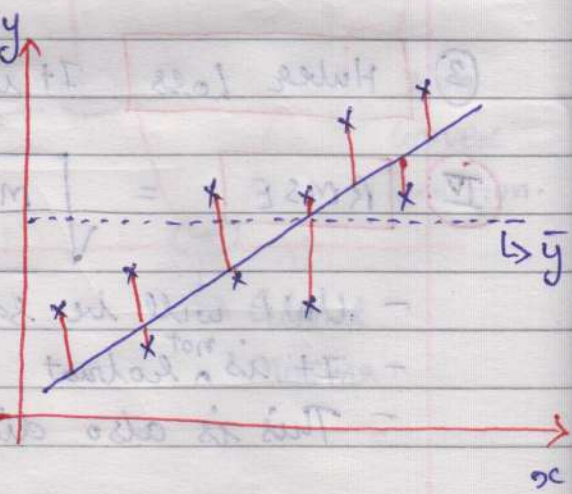
where SS_{Res} = Sum of Square Residuals.
SS_{Total} = Sum of Square Average.

$$R\ \text{Squared} = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}$$

Where $\bar{y}$ = Averaige of y

If the model is fitted well

$$R\ \text{Squared} = 1 - \frac{\text{Small No.}}{\text{Bigger No.}}$$



$\therefore$ R Squared $= 1 - \left\{ \dfrac{\text{Small Number}}{\text{Bigger Number}} \right\} \rightarrow$ Small Number

Therefore, If the model is fitted well

R Squared ≤ 1

Suppose R Squared $= 0.85$ $\therefore$ 85% Accurate.
" R Squared $= 0.75$ $\therefore$ 75% Accurate.

Question:- What R Squared Do?

Answer:- R Squared basically measures the performance of the model that you have created.

Question:- Can R Squared value be -ve?
Answer:- If R Squared value comes -ve, then the model is very bad

(2) Adjusted R Squared

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

where N = No. of data points.
P = No. of Independent features.

Size of House	City Location	No. of Bedrooms	Gender	Price.

Accuracy

R^2 65% {Size of House} Adj R^2 = 63% $P = 1$

R^2 75% {City Location} Adj R^2 = 73%. $P = 2$

R^2 88% {No. of Bedroom} ⋮

R^2 90%. Adj R^2 = 85%.

Question What Adjusted R^2 Do?

Answer:- The Adjusted R^2 is going to evalute the accuracy based on the very important features.

Overfitting And Underfitting

(Bias And Variance)

DATASET

1000 datapoints → Training (700) → Model Training

1000 datapoints → Test (300) → Model Test

TRAINING DATASET (700)

(500) TRAINING ⇓ Train the Model

(200) VALIDATION ⇓ Hyperparameter Tunning the Model

$\boxed{\text{Model}}$

TRAIN DATA	Very Good Accuracy (90%) [BIAS]
TEST DATA	Very Good Accuracy (85%) [Variance]

- Whenever we talk about training data accuracy, we are basically going to use $\boxed{\text{BIAS}}$.

- Whenever we talk about test data accuracy, we are basically going to use $\boxed{\text{Variance}}$.

Note:-

High Train Data Accuracy - Low Bias.
High Test Data Accuracy - Low Variance.
Low Train Data Accuracy - High Bias.
Low Test Data Accuracy - High Variance.

Example:-
Let Say for Training Data, we have very Good Accuracy [90%] [Low Bias]

For Test Data, Bad Accuracy [50%] [High Variance]

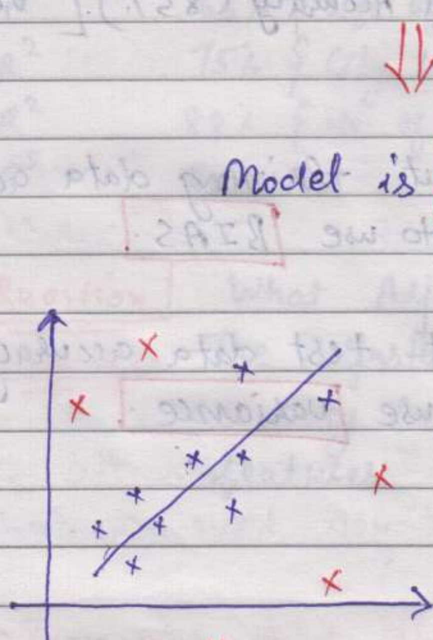
This condition where there is Low Bias and High Variance is called $\boxed{\text{Overfitting}}$

Example:-

For Train Data	Model Accuracy is Low [High Bias]
For Test Data	Model Accuracy is Low / High [Low or High Accuracy]

⇓

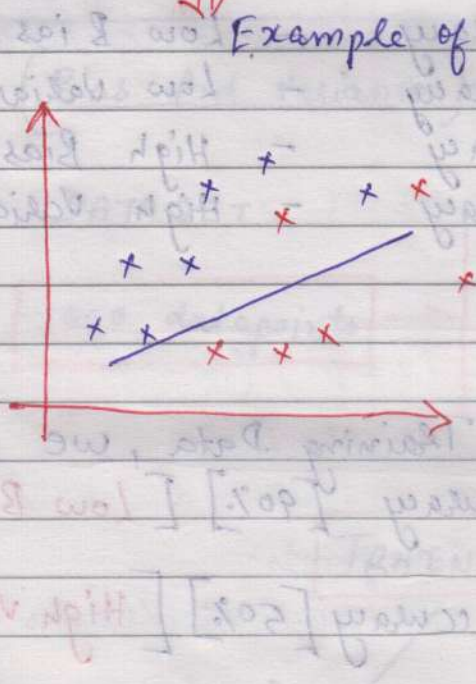
Model is **Underfitting**



x → Training Datapoints.

x → Test Datapoints.

⇓ Example of **overfitting**.



x → Training datapoints

x → Test datapoints.

⇒ Example of **Underfitting**