Linear Regression {Regressive Prediction model}

- This Model gives o/p in Continuous Value format i.e [−∞ to ∞]
- Supervised learning model.

$y = f(x)$

- y → o/p Variable (Dependent Variable) (Unknown Value) [−∞ to ∞] Regressive Prediction Value
- f → Straight line method (or) Ordinary least square method (OLS)
- x → i/p Variable (Independent Variable) (known Value) (Explanatory Variable)

{ this Method/Model is used to cal. the Unknown Value based on the known Value }

"Assumes linear relationship b/w independent & dependent Variable"

Regression → Statistical tech. used to model the relationship b/w the dependent Variable & the Independent Variable (one/more)

- Based on the Independent Variable we can Predict the dependent Variable.

m & c Value calculated Based on Historical data

Simple Linear Regression (only one Independent Variable)

→ Straight line Used as Best fit line Prediction

$y = mx + c$

- y → Predicted o/p
- m → Slope
- x → Independent Variable
- c → Intercept

find y based on known Value (x)

Multiple Linear Regression (two or more Independent Variable)

$y = m_1x_1 + m_2x_2 + \dots + m_nx_n + c$

- y → Predicted o/p
- $x_1, x_2, \dots$ → i/p Variable

Goal of linear Regression :- find Best fit line which minimize the error b/w the predicted o/p & actual o/p based on the historical data (Training dataset)

Error (Cost function) = $y - \hat{y}$
residual — actual o/p — predicted o/p

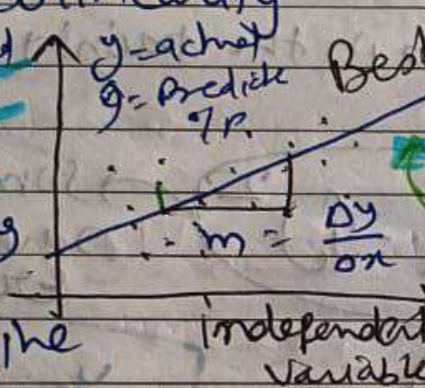
$(X^T X)\beta = (X^T Y)$

$\beta = (X^T X)^{-1}(X^T Y)$

Assumption in the linear Regression
- Linearity ($x \propto y$ or $x \propto \frac{1}{y}$)
- Homoscadasicity
- Multicollinearity

distance b/w data points less

$\beta = (m, c)$



with training Set If we fix optimal line

(find best line use performance matrix)

$\begin{pmatrix} 1 & x_1 \\ 1 & x_2 \end{pmatrix}\begin{pmatrix} \quad \end{pmatrix} = m$

$\begin{pmatrix} 1 & x_1 \\ 1 & x_2 \end{pmatrix}\begin{pmatrix} 1 & x_1 \\ 1 & x_2 \end{pmatrix} =$

Performance Matrices :-

1. R^2 measure
2. Adjusted R^2 measure
3. Mean square error (MSE) → Penalize large errors more than smaller ones
4. Root Mean Square Error (RMSE)
5. Mean Absolute Error (MAE)
6. Mean Absolute Percentage Error (MAPE)

R^2 Measure :-

$$R^2 = 1 - \left[\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}\right]$$

n → no. of samples
$\hat{y}_i$ → Predicted o/p
RSS → $\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$
TSS → $\sum_{i=1}^{n}(y_i - \bar{y})^2$
($\bar{y}$ → mean value of actual o/p)

Variance in the dependent variable that is predictable from independent variable

- R^2 value will be measured [0 to 1]
- R^2 value is "1" means No error in the model

No error in the model (100% accuracy)

R^2 close to zero then more errors present in the model - error is high

Re-training Req: $\beta(X^TX) = (X^TY)$

$\beta = (X^TX)^{-1}(X^TY)$

m c

Adjusted R^2 Measure

$$\text{Adjusted } R^2 = 1 - \left[\frac{(1-R^2)(n-1)}{n-k-1}\right]$$

n: # samples in training set (rows)

K: # i/p Attributes in the training set (col)

Mean Square Error (MSE)

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

Sum of squared error

$$MSE = \frac{SSE}{n}$$ # n samples in training set

$y = mx + c$

① $\bar{y} = m\bar{x} + c$

② $m = \frac{Cov(X,Y)}{Var(X)}$

$E(XY) = mE(X^2) + cE(X)$

④ RMSE (Root Mean Square Error)

$$RMSE = \sqrt{MSE}$$

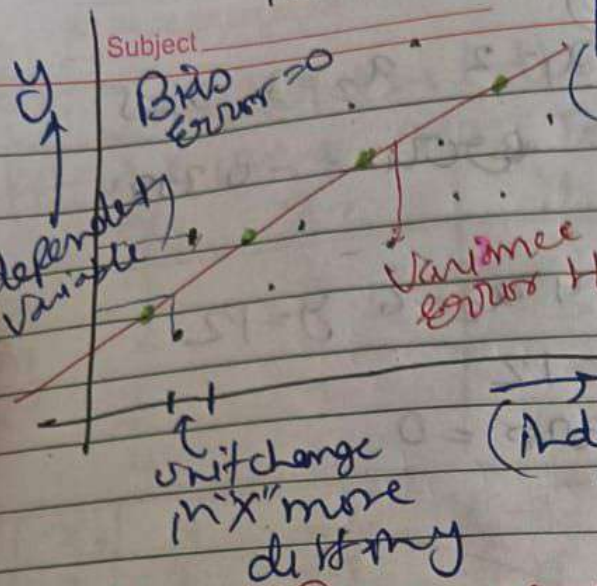
⑤ Mean Absolute Error

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

⑥ MAPE (Mean Absolute Percentage Error)

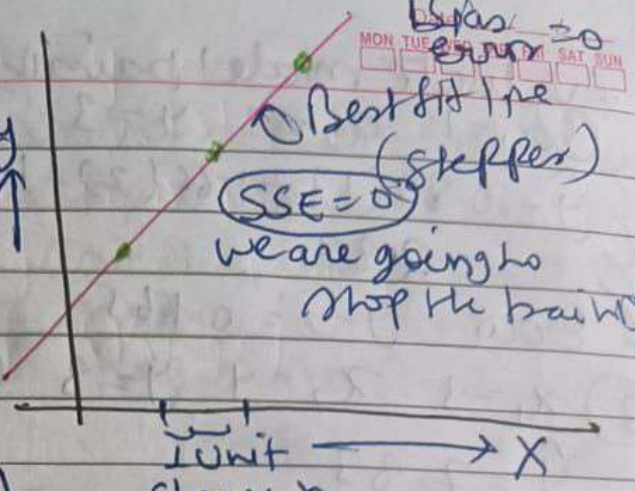
$$MAE \times 100$$

• train
• test



(Over fitting Model)

• Train



Regularization Techniques → Introduce some error in the model to improve the performance Matrics the model undergoes Overfitting.

Ridge Regularization (L-2 Reg)

Lasso Regularization

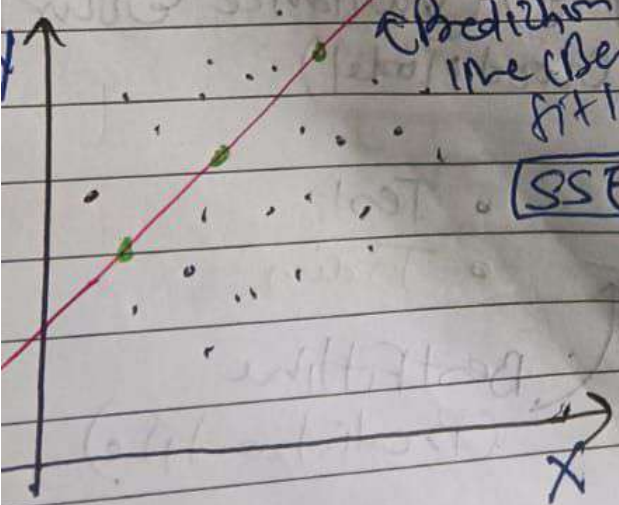
Ridge Regularization :- error is $\lambda(\text{slope})^2$

$\lambda : [0 - \infty]$

error is added to the SSE Prediction line

$$\text{Model} = SSE + \lambda(\text{slope})^2$$

Ridge Boundary



$SSE + \lambda(\text{slope})^2$

$0 + \lambda(0)(\text{slope})^2$

if SSE again zero Model back on "overfitting"

if ($\lambda = \infty$) then Prediction line is close to 'X' axis i.e. straight

→ L1 Regularized (Kite shape)

lasso Regression : lasso penality error is $\lambda|slope|$ $(0-\infty)$

Add lasso penality to the SSE, to improve the model accuracy when the model undergoes overfitting

lasso penality = SSE + $\lambda|slope|$
Model Error

Major diff. b/w Ridge & lasso

Ridge

In this tech, new prediction line (slope) close to origin (X-axis) but it never match to origin

$\lambda(slope)^2$

lasso

- In this tech, new prediction line (slope) is equal to origin (X axis)

$\lambda|slope|$

slope of any attribute is "0" then we remove that attribute

feature selection

Simple linear Regression ($\hat{y} = mx + c$)

lasso penality = SSE + $\lambda|m|$

Multiple linear Reg. ($\hat{y} = m_1x_1 + m_2x_2 + \dots + m_nx_n + c$)

lasso penality = SSE + $\lambda[|m_1| + |m_2| + \dots + |m_n|]$

Eg. Consider the following multiple linear Reg.

$$\hat{y} = m_1x_1 + m_2x_2 + m_3x_3 + m_4x_4 + c$$

Suppose lasso penality = SSE + $\lambda[|m_1| + |m_2| + |m_3| + |m_4|]$

Assume m_2 & m_4 are zero then realize that Attributes 2 & 4 are not so important (less) in model

So delete 2 & 4 Attributes from dataset $\{\hat{y} = m_1x_1 + m_3x_3\}$