Follow us!

Share it!

LINEAR REGRESSION

LINEAR REGRESSION IS A WAY TO EXPLAIN THE RELATIONSHIP BETWEEN DEPENDENT VARIABLE AND ONE OR MORE EXPLANATORY VARIABLES USING A STRAIGHT LINE.

LINEAR REGRESSION FORMULA

$$y = mx + c$$

y is the criterion variable
x is the predictor variable
c is the constant/intercept
m is the regression coefficient

Linear regression can therefore, predict the value of (y) when only the (x) is known. It doesn't depend on any other factors.

INTUITIVE EXPLANATION

Linear regression is one of the simplest and most commonly used data analysis and predictive modelling techniques. The linear regression aims to find an equation for a continuous response variable known as (y) which will be a function of one or more variables (x).

(1) Linear Regression

Regression => Linear regression → $y = mx + c$
Laso regression (the tugeline for liner Regression is best fit line)
Ridge regression
Elastic Search (Lasso + Ridge)

* ML Project

(1) Collect data
(2) Analysis data (EDA)
→ ~~Numerical~~ Numerical data
→ categorical data
→ outliers
→ distribution data
→ Imbalance data
→ balance data
(3) Preprocess
(4) ML Algo (model)
(5) Evaluate

Regression	Classification
R^2	Accuracy
adj R^2 (adjusted R^2)	Error
	F1
	Precision
	Recall

(6) Retrained [If not getting proper result/accuracy]
[I am getting low Accuracy]

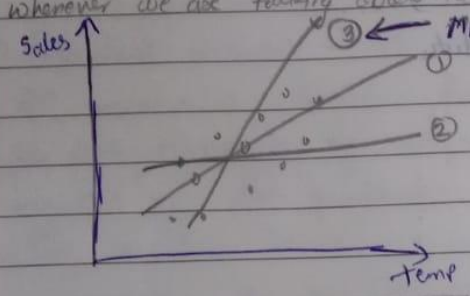
Math

1. Linear Algebra → {Linear Eq, System of Linear Eq, orthogonality, Angle, distance and length, Eigen value/Eigen vector, vector, matrix}
2. Calculus → {Differentiation, Partial diff, Maxima-Minima, Integration}
3. Probability
4. Statistics
5. Geometry

* Linear Regression

$y = mx + c$ {Equation of line}

→ whenever we are talking about LR, we have to find Best fit line

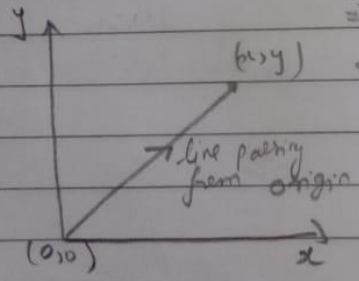


← Minimize Error (Best fit line) (having lowest error)

↑ This is called OLS (ordinary least sq) (Minimize the Error)

$$\boxed{y = mx + C}$$ ...(Point Slope formula)

• Vector Representation



$\Rightarrow \hat{x}i + \hat{y}j$

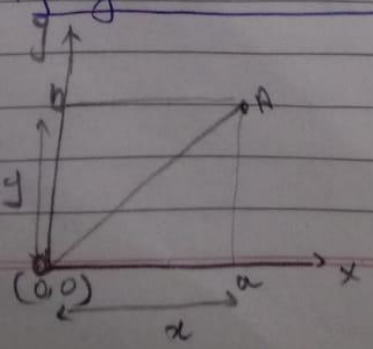
$\Rightarrow [x\ y]$

row vector $[x_1\ x_2\ x_3 \ldots x_n]_{1\times n}$

column vector $\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}_{n\times 1}$

* ~~Pythagoras theorem~~

* Pythogerous theorem



$H^2 = B^2 + L^2$

$OA^2 = Oa^2 + Ob^2$ Euclidean

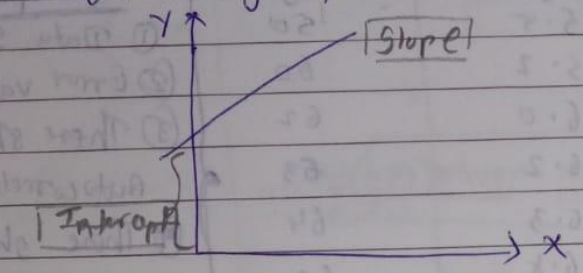
$OA = \sqrt{Oa^2 + Ob^2}$

$$\boxed{OA = \sqrt{x^2 + y^2}}$$

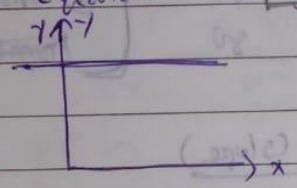
Euclidean distance

$$x_1 x_2 = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

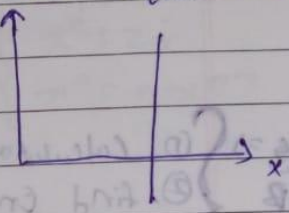
- General Equation of particular line => $\boxed{ax + by + c = 0}$



- horizontal Equation => $\boxed{y = x}$



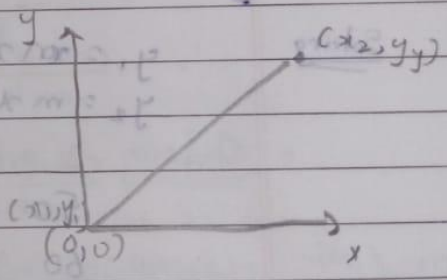
- Vertical Equation => $\boxed{x = y}$



$$\boxed{\text{slope } (m) = \frac{y_2 - y_1}{x_2 - x_1}} \text{ (or) } \frac{\Delta y}{\Delta x}$$

- If my line is going through origin

$$\boxed{y = mx}$$



1. point slope => $y = mx + c$
2. General Equation of Line => $ax + by + c = 0$
3. Horizontal Equation => $y = x$
4. Vertical Equation => $x = y$
5. Slope (m) => $m = \frac{y_2 - y_1}{x_2 - x_1}$ (or) $\boxed{\frac{\Delta y}{\Delta x}}$
6. line passing through origin => $y = mx$

* $y = mx + C$

Dataset → Based on Height we will predit weight

Height	weight
5.5	50
5.7	60
6.0	62
6.2	63
6.3	64
6.8	69
7.1	70
7.2	80

getting (199) error

Autocorrelation = Correlation values by itself

* Assumptions (for LR)
1. Data should be linear
2. Error value tends to zero
3. There should not be any Autocorrelation between error
4. There should not be multicolinearity between colum
5. There should not be ~~homoscedasticity~~. & homoscadisticity

Steps for LR ⇒
1. Calculation ($y = mx + c$) { m (slope) $m = y/x$ (rate of changes); C → Intercept }
2. find Error (MSE)
3. optimize (GD)

Solving

$y_1 = mx_1 + C$

$y_2 = mx_2 + C$

choosing 2 points from the Dataset, randomly

$50 = m \times 5.5 + C$ — ①

$60 = m \times 5.7 + C$ — ②

$60 = 5.7m + (50 - 5.5m)$ $C = (50 - 5.5m)$

$60 = 50 + 5.7m - 5.5m$

$60 = 50 + 0.2m$

$60 - 50 = 0.2m$

$10 = 0.2m$

$m = \frac{10}{0.2}$ $\frac{100}{2}$ 50

$m = 50$

$c = 75$

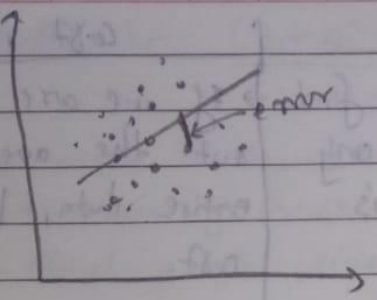
$y = mx + c$

$\boxed{y = 50x + 75}$

$y = 50(6.2) + 75$

$\Rightarrow y = 199$

* Representation



$$\text{Error} = \sum_{i=1} (y - \hat{y}) \quad \text{where, } \boxed{\hat{y} = mx + c}$$

• Differentiation is a rate of change of a particular point

$$\frac{dy}{dx} = \frac{\Delta y}{\Delta x} \quad \lim_{\substack{x \to 0 \\ y \to 0}} \qquad \begin{cases} x^2 = 2x \\ x^n = nx^{(n-1)} \end{cases}$$

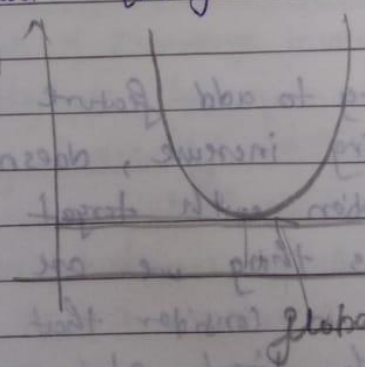
Optimization formula
Gradient decent

Slope → $m_{new} = m_{old} - \eta \frac{d(\text{error})}{dm}$

Intercept → $C_{new} = C_{old} - \eta \frac{d(\text{error})}{dc}$

use of η ⇒ η = we use learning Rate (η) for optimizing this m value for controlling the flow of this particular point. we dont want sudden change in our value we want minute change in our value

GD



Loss	Cost
→ If we are gaining to find out different w.r.t only one point, hence it is called loss.	→ If we are going to find out the average w.r.t this entire data, hence this is called cost.

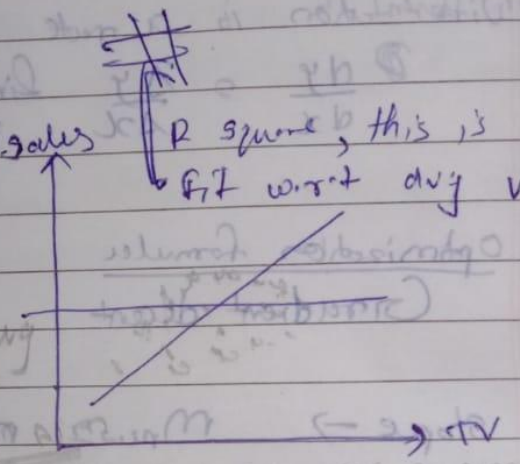
- Homoscedasticity

- $y = mx + C$

  m ↑ coefficient, C ↑ Intercept

- $R^2 = 1 - \frac{RSS}{TSS}$

  $= \frac{TSS - RSS}{TSS}$



R square, this is goodness of fit w.r.t avg value

- **OLS** (Ordinary Least Square)

  → we need to find out optimal parameter m and C means we need to find out this best fit line w.r.t our data.

  → It is just a representation of this linear model.

- **Adj R^2**

  → As we are going to add feature in Dataset. So my R^2 value getting increase, doesn't matter it having any relation with target feature.

  → So, for solving this thing we are calculating Adj R^2

  → Over there only we consider that feature which having any relation with target column.

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

where, R^2 = Sample R-Square
P = Number of Predictors
N = Total sample size

- **Multi-Collinearity**

→ (more than one) (we are going to check dependency, we are going to check linearity)

→ if we have correlation between this two variable

TV	News	radio	Sales
X_1	X_2	X_3	Y

→ In multicollinearity, if we are having correlation between this two variable, this independent variable ~~it should not be~~.

→ In short, correlation between ~~variable~~ this independent variable we are going to ~~check~~ check the correlation.

Imp ⇒ More than one Linear relation in an Equation, hence the name Multi-collinearity.

eg:-

Salary = a*(years of Experience) + b*(age) + C (A typical LR Equation)
where, "a and b are coefficients" and "c is constant"

⇓

Here, age and years of Experience are correlated, since for a person as ~~they~~ the years of Experience increases, his/her age also increase. Mathematically,

Age = years of Experience + d | d → constant, age when person started Job

⇓

So, our LR Equation which was only supposed to have one linear relation, i.e. $Y = mx_1 + mx_2 + C$ now has one more relation

→ $X2 = ax_1 + d$

⇓

More than one Linear relation in an Equation, hence the name Multi-collinearity

correlation

Age Year of Experience Sale

based on Age we can predict Salary

Age YoE Salary

- we can discard one variable from independent feature

→ As the extent of the collinearity increases, there is a chance that we might produce an overfitted model. An overfitted model works with the test data but its accuracy fluctuates when exposed to other data sets.

- Variance Inflation Factor (VIF) = Regression of one X variable against other X variables.

$$VIF = \frac{1}{(1-R^2)}$$

(corr between a variable)

0 — low multicollinearity — 10 — High multicollinearity

* Regularization

→ when we use regression models to train some data, there is good chance that the model will overfit the given training data set. Regularization helps sort this overfitting problem by restricting the degrees of a given equation i.e. simply reducing the number of degrees of a polynomial function by reducing their corresponding weights.

In LR, we do not want huge weight/coefficient as a small change in weight can make a large difference for the dependent variable (Y). so, regularization constraints the weights of such features to avoid overfitting.

Note ⇒ • In Linear Regression we are getting problem that is called overfitting.

→ we need to optimize this parameter m or c value.

overfitting means ⇒ { train acc is v.v. good
test acc is v.v. low.

we need to optimize our m or c parameter.

(L1) Lasso (Least Absolute shrinkage and Selection Operator)

→ Lasso regression penalizes the model based on the sum of magnitude of the coefficients. The regularization term is given by regularization ⇒ $\lambda * \sum |\beta_i|$

$y = mx + c$

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

$m_{new} = m_{old} - \eta \frac{\partial L}{\partial m}$

$\boxed{(y - \hat{y})^2 + \lambda |m|}$

$m_{new} = m_{old} - \eta \partial [(y - \hat{y})^2 + \partial |m|]$

$m_{old} - \eta \frac{\partial [(y - (mx + c))^2 + \lambda |m|]}{\partial m}$

→ λ is the shrinkage factor, and hence the formula for loss after regularization is

$$\boxed{\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij}\right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| = RSS + \lambda \sum_{j=1}^{p} |\beta_j|}$$

(L2) Ridge

→ Ridge Regression penalizes the model based on the sum of squares of magnitude of the coefficient. The regularization term is given by

regularization $= \lambda * \sum |\beta_j^2|$ ... λ is shrinkage factor

$$\boxed{\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij}\right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 = RSS + \lambda \sum_{j=1}^{p} \beta_j^2}$$

- Difference between Ridge and Lasso

→ Ridge regression shrinks the coefficient for those predictors which contribute very less in the model but huge weights, very close to zero. But it never makes them exactly zero. Thus, the final model will still contain all those predictors, though with less weights. This doesn't help in interpreting the model very well. This is were Lasso regression differs with Ridge regression. In Lasso, the L1 penalty does reduce some coefficient exactly to zero when we use a sufficiently large tuning ~~parameter~~ parameter λ. So, in addition to regularizing, lasso also performs feature selection.

Why use Regularization?

⇒ Regularization helps to reduce the variance of the models without a substantial increase in the bias. If there is variance in the model that means that the model won't fit well for dataset different that training data. The tuning parameter λ controls this bias and variance tradeoff. When the value of λ is increased up to a certain limit, it reduce the variance without losing any important properties in the data. But after a certain limit the model will start losing some important properties which will increase the bias in the data. Thus, the selection of good value of λ is the key. The value of λ is selected using cross-validation methods. A set of λ is selected and cross-validation error is calculated for each value of λ and that value of λ is selected for which the cross-validation error is minimum

Note :- ~~The range of~~ If we perform standardization (SD) in data then the minimum range = -3 maximum range +3 { Not more considering $\Rightarrow y = m_1 x_1 + m_2 x_2 + c$ then this not less than this

L_1	L_2
Lasso	Ridge
passing in $loss + \lambda \lvert m_1 + m_2 + m_3 \rvert$	$loss + \lambda \left[\lvert m_1^2 \rvert + \lvert m_2^2 \rvert + \lvert m_3^2 \rvert \right]$
Gradient function	passing in Gradient function
$m_{new} = m_{old} + \eta \frac{\partial l}{\partial m}$	$m_{new} = m_{old} + \eta \frac{\partial l}{\partial m}$