Multiple linear Regression

→ Using Adjusted R^2 score is a good idea during Multiple linear Regression

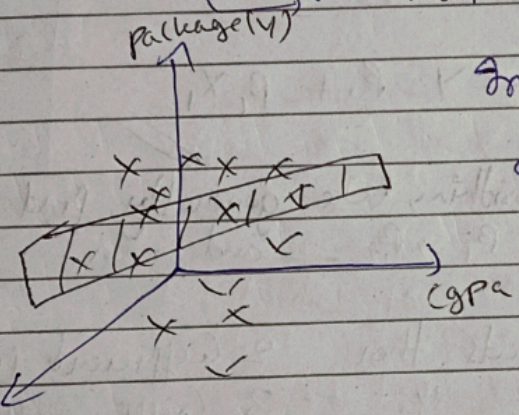
Multiple Linear Regression

→ It is the extension form of Simple linear regression
→ we have more than one input in this.

Let's suppose

cgpa	iq	Package (Y) LPA

→ Now it is in 3D



In 2D → we draw best fit line

In 3D → Plane

In 4D → hyper plane

We take a plane or In 3D we fit a plane.

→ The plane is the surface that minimizes the distance(errors) between actual and predicted value

Formulation → In Simple LR ⇒ $y = mx + c$

but for more inputs, ~~more~~ the equation will look like :-

Suppose for two inputs

$$y = m_1X_1 + m_2X_2 + b$$

→ where y plane intercept at y.

m_2 is the slope of X_2 feature

m_1 is the slope of X_1 feature (deh

Literature expression

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2$$

$\beta_1 X_1$: m_1 X_1 ; $\beta_2 X_2$: m_2

β_0 : b

We have to find $[\beta_0, \beta_1, \beta_2]$ → m_1, m_2 ; $\beta_0 = b$

→ 3 input 1 output

→ If our data is in 4D

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3$$

→ Same as for n inputs

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \cdots \beta_n X_n$$

or

$$\left[Y = \beta_0 + \sum_{i=1}^{n} \beta_i X_i \right]$$ General equation

For 2D

$n=1$ then $Y = \beta_0 + \beta_1 X_1$

→ In this algorithm we generally find the coefficients of data $\beta_0, \beta_1, \beta_2$ -- and so on

In 2 I inputs then 2 Coefficients find

" " " 3 Coeffi - "

if n inputs then n+1 coefficients needs to find

Taking example -

$X_1 \rightarrow$ CGPA , $X_2 \rightarrow$ IQ , $Y \rightarrow$ LPA (Package)

$$Y = \beta_0 + \beta_1 X (GPA + \beta_2 X IQ$$

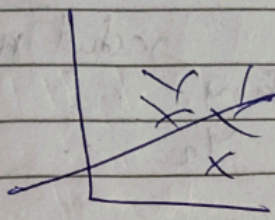
these are the weightage. like ~~Ex~~ Y is dependent on β_2 of IQ

IQ ↑ Y ↑ , IQ ↓ B ↓ Just like that

β_0 = Offset (if all others are zero then)

Mathematical formulation from Scratch

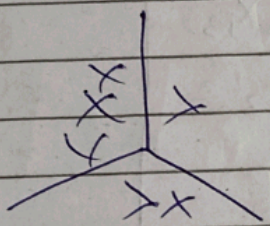
- In Simple



$\rightarrow y = mx + b$

we find (m, b)

here in 2D or >1D



what we find

$$\left[Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots \beta_n X_n \right]$$

We will find $\beta_0, \beta_1, \beta_2 \cdots \beta_n$ here which are the coefficients.

Let's Suppose

cgpa	iq	gender	LPA (Package) → Target
X_1	X_2	X_3	Y

For only 1 input (in 2D) $\rightarrow y = mx + b$

$Y = \beta_0 + \beta_1 X_1$

Similarly for all

$$\left[Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 \right]$$

↳ Predicted value

Suppose we have 100 students (100, 4) ← 4 columns.

In next page

$\beta_0\ \beta_1\ \beta_2\ \beta_3$ → we have to find it

like an original dataset each student has their CGPA & Package (Y) given

1st student	$\hat{Y}_1$ → actual package from the dataset
2nd student	Y_2

let's make a matrices

$$\begin{bmatrix} \hat{Y}_1 \\ \hat{Y}_2 \\ \vdots \\ \hat{Y}_n \end{bmatrix} = \begin{bmatrix} \beta_0 & \beta_1 X_{11} & \beta_2 X_{12} & \beta_3 X_{13} \\ \beta_0 & \beta_1 X_{21} & \beta_2 X_{22} & \beta_3 X_{23} \\ \vdots & \vdots & \vdots & \vdots \\ \beta_0 & \beta_1 X_{100,1} & \beta_2 X_{100,2} & \beta_3 X_{100,3} \end{bmatrix}$$

$\hat{Y}_1$ ⇒ Predicted Package of 1st student

$\hat{Y}_2$ = " " " 2nd st

$\beta_1 X_{11}$ = CGPA of 1st student

↳ 2it student in the table 1st row 1st column

$\beta_3 X_{13}$ ⇒ Genders of the 2st student

$\beta_1 X_{1001}$ ⇒ 100th student's CGPA

100th student, 1st feature

$\beta_1 X_{1002}$ ⇒ IQ of 100th student

* Taking a small Assumption for better

We don't have 100 rows we have n rows

" " " 3 input col & 1 outp → n columns

100 rows → n rows

4 col → 3 col (input) → n cols

First student ka

$$\hat{Y} = \begin{bmatrix} \hat{Y}_1 \\ \hat{Y}_2 \\ \vdots \\ \hat{Y}_n \end{bmatrix} = \begin{bmatrix} \beta_0 & \beta_2 X_{11} & \beta_1 X_{12} & \beta_0 X_{13} & \cdots & \beta \end{bmatrix} \} Y$$

wrong

$$\hat{Y} = \begin{bmatrix} \hat{Y}_1 \\ \hat{Y}_2 \\ \vdots \\ \hat{Y}_n \end{bmatrix} = \begin{bmatrix} \beta_0 & \beta_1 X_{11} & \beta_2 X_{12} & \beta_3 X_{13} & \cdots & \beta_m X_{1m} \\ \beta_0 & \beta_1 X_{21} & \beta_2 X_{22} & \beta_3 X_{23} & \cdots & \beta_m X_{2m} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ \beta_0 & \beta_1 X_{n1} & \beta_2 X_{n2} & \beta_3 X_{n3} & \cdots & \beta_m X_{nm} \end{bmatrix}$$

$\beta_1 X_{11} \Rightarrow$ 1st student ka 1st value (feature 1 me)
$\beta_2 X_{22} \Rightarrow$ 2nd student ka 2nd value (feature 2 me)

We can also write the Matrix $\hat{Y}$ as following (Seperating $\beta_0, \beta_1 \cdots \beta_n$) product ke form me

A X B

$$\hat{Y} = \begin{bmatrix} 1 & X_{11} & X_{12} & \cdots & X_{1m} \\ 1 & X_{21} & X_{22} & \cdots & X_{2m} \\ 1 & X_{31} & X_{32} & \cdots & X_{3m} \\ \vdots & & & & \\ 1 & X_{n1} & X_{n2} & \cdots & X_{nm} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix}$$

by finding its product we will also get $\hat{Y}$

Let's we will called this matrix as X and this β

So

$$\hat{Y} = X\beta \quad \text{---} \textcircled{1}$$

$\hat{Y}$ matrix = X matrix * β matrix

How we call $\hat{Y} = X\beta$ in English or professin

$\hat{Y}$ is → The prediction of all the input rows (of all students)
$\hat{Y}$ is the package of each student predicted

if we have 1000 students the $\hat{Y}$ will contain 1000 predicted values in the matrix

$$\hat{Y} = X\beta$$

Now from here what is β ⇒ in 2D $\begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}$, in 3D $\begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix}$ and so on

Q. Now, what is X matrix
Suppose if we have cgpa | iq | gender
X_1 X_2 X_3

→ for 1

extra col	X_1	X_2	X_3
1	X_{11}	X_{12}	X_{13}
1	X_{21}	X_{22}	X_{23}
1 :	X_{31}	X_{32}	X_{33}

This is 'X' matrices

From a given df → XY, if we extract Y

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ y_n \end{bmatrix}$$ → It is the actual value in the data

Now we are creating a matrix $E = Y - \hat{Y}$ = actual − predicted

$$E = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} - \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_n \end{bmatrix} = \begin{bmatrix} y_1 - \hat{y}_1 \\ y_2 - \hat{y}_2 \\ \vdots \\ y_n - \hat{y}_n \end{bmatrix}$$

Now In single LR

Error $E = e = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$

↳ which is the error

And now we have $E = \begin{bmatrix} y_1 - \hat{y}_1 \\ y_2 - \hat{y}_2 \\ \vdots \\ y_n - \hat{y}_n \end{bmatrix}$

if we will find $E^T E$ it will be equal to e ($\sum_{i=1}^{n} (y_i - \hat{y}_i)^2$)

↓ Proving it

$E^T E$

$E^T \downarrow E$ $\qquad$ E^T $\qquad$ E

$\begin{bmatrix} y_1 - \hat{y}_1 & (y_2 - \hat{y}_2) & (y_3 - \hat{y}_3) & \cdots & (y_n - \hat{y}_n) \end{bmatrix} \begin{bmatrix} (y_1 - \hat{y}_1) \\ (y_2 - \hat{y}_2) \\ \vdots \\ (y_n - \hat{y}_n) \end{bmatrix}$

↳ shape (1, n) $\qquad$ shape (n, 1)

Now after multiply we will get

$= (y_1 - \hat{y}_1)^2 + (y_2 - \hat{y}_2)^2 + (y_3 - \hat{y}_3)^2 + \cdots (y_n - \hat{y}_n)^2$

$\Rightarrow (y_1 - \hat{y}_1)^2 + (y_2 - \hat{y}_2)^2 + \cdots (y_n - \hat{y}_n)^2$

$= \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$ (we same as e)

↳ Error function

So, we can write our Error function E (in matrix as)

$$E = E^T E = (y - \hat{y})^T (y - \hat{y})$$

And we know that (from 12th class matrices)

$(A+B)^T = A^T + B^T$ } by applying this rule
$(A-B)^T = A^T - B^T$

$E = (Y^T - \hat{Y}^T)(Y - \hat{Y})$ — ③

And from eq ① we know that $\hat{Y} = X\beta$

putting the val of eq ① in eq ③

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

$E = Y^TY - Y^TX\beta - Y(X\beta)^T + (X\beta)^T * X\beta$

Now we have to prove these both are eq

↓

Detour for this

$Y^TX\beta$ $(X\beta)^TY$ is same

let's $Y = A$, $X\beta = B$

$A^TB = B^TA$ — ①

② We know that $(AB)^T = B^TA^T$ Same logic

$(A^TB)^T = B^TA$ — ② $(A^T)^T = A$

Means, we have to prove $A^TB = (A^TB)^T$

lets say $A^TB = C$

And we want to prove $C = C^T$

↓

if we can

$C = A^T B$ $\quad [A = Y, B = X\beta$

$C = Y^T X \beta$

we have to prove

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

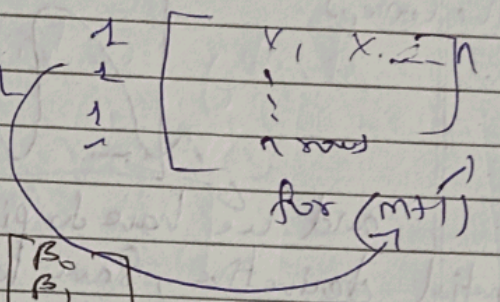
$Y^T X \beta$

→ lets say we have n studen then $Y = \begin{bmatrix} = \\ n \end{bmatrix}$ shape = $n \times 1$

Transpose of Y → shape ⇒ $(1 \times n)$

→ for X → for n stud

Shape ⇒ $n \times (m+1)$ ← 1 extra wl



→ For β for n st $\beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_n \end{bmatrix}$

↳ n rows but with n inputs we have m+1 coef

Shape ⇒ $(m+1) \times 1$

⇒ $(m+1) \times 1$

So → $Y^T X \beta$

$(1 \times n)(n \times (m+1))((m+1) \times 1)$

↳ after multiplying it because n is same

$(1 \times m+1)((m+1) \times 1)$

$1 \times 1 = [1]$ → this is a matrix

if we find its Transpose it will be same

$[1]^T = [1]$ Same

Mean, We have proved $Y^T X\beta = (Y^T X\beta)^T$ then $Y^T X\beta = (X\beta^T Y$ = Proved

Now from last eqn

$$E = Y^T Y - Y^T X\beta - (X\beta)^T Y + (X\beta)^T * X\beta$$

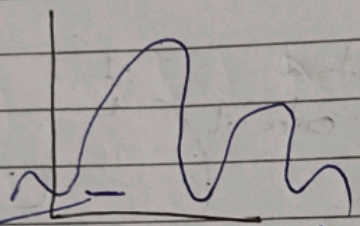
↳ both are same

Now

$$E = Y^T Y - 2Y^T X\beta + X^T \beta^T X\beta$$

The same Concept we use in Simple linear Regression we will use it here also

The Concept



the mse graph going like that

and we have to find the minimum m & b value by partial derivative, same logic here and equating with 0

$$\frac{dE}{d\beta} = \frac{d}{d\beta}\left[\underbrace{Y^T Y}_{0} - \underbrace{2Y^T X\beta}_{2Y^T X} + \underbrace{X^T \beta^T X\beta}_{X^T A}\right] = 0$$

$$= 0 - 2Y^T X + \frac{d}{d\beta}\left[X^T \beta^T X\beta\right] = 0$$

here β & β^T both occurs that's why it is differentiated

⇒ So we have to study Matrix differentiation but I will learn it later during revision, currently direct use

$$Y = A^T X A$$

$$\frac{dY}{dA} = 2XA^T \rightarrow \text{same logic for eqn } \frac{dG}{d\beta}$$

$$= 0 - 2Y^T X + \left[2X\beta^T X^T\right]$$

$$\Rightarrow -2Y^TX + 2XX^T\beta^T = 0$$

$$\Rightarrow 2XX^T\beta^T - 2Y^TX = 0$$

$$XX^T\beta^T - Y^TX = 0$$

$$XX^T\beta^T = Y^TX$$

$$\beta^T = \frac{Y^TX}{XX^T}$$

Now

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

Transpose both side

$$(\beta^T)^T = \left[(Y^TX)(XX^T)^{-1}\right]^T$$

$$\beta = \left[(XX^T)^{-1}\right]^T (Y^TX)^T \qquad (A^T)^T = A$$

$$\beta = \left[(XX^T)^{-1}\right]^T (X^TY)$$

↳ This is a square matrix, and after transposing we will also get the same matrix

$$\boxed{\beta = (XX^T)^{-1}(X^TY)}$$ → final answer which we are finding

here $X \rightarrow X\_train$ ⇒ one for bias, X_{train}

$Y \rightarrow Y\_train$ (actual value)

$\beta = n+1$, where n is the feature., always β will be one more β_0

Verifying

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

$\begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_n \end{bmatrix} = (m+1) \times (m+1) \; [(m+1) \times n] \; [n \times 1]$

$(m+1) \times 1 = [(m+1) \times n] \; [n \times 1]$

Shape

$(m+1) \times 1 = (m+1) \times 1$ Proved the shape is same

Why Gradient Descent?

Sklearn uses
- OLS for finding β_0 & β
- Gradient descent

OLS gives β by using the $\beta = (X^T X)^{-1} X^T y$, but question is that why sklearn or more ML engi use Gradient descent

the answer is $(X^T X)^{-1}$ this inverse function

because OLS compute β with each, but the matrix inversion complexity is too high $O(n^3)$

So, for reducing it we will use Gradient descent

(Task)
OLS for OLS → for Gradient Descent

Linear Regression — SGD Regression

Scratch Code in Laptop