Types of GRADIENT DESCENT

Batch GD	Stochastic GD	Mini Batch G.D

Batch GD: The first GD was Batch G.D. It is generally slow and some problem with computation as slope is found out based on all rows

$$m_n = m_0 - \eta \times (\text{slope})_{\text{wrt } m}$$
$$b_n = b_0 - \eta \times (\text{slope})$$

To prevent this, use $\boxed{\text{Stochastic G.D}}$ as it find error / updates are based on 1 row & then change m & b

→ fast but prone to error
(suitable for larger datasets)

Mini Batch GD: In this we define batch size of 30 (originally 300) means the update ~~will~~ in m & b based on 30 rows, which means value of m & b will be updated 10 times (30 × 10 = 300)

In stochastic, 300 rows means 300 updates in m & b.

Note :- Mostly use stochastic (and sometimes internally Mini Batch)

2 Batch GD is rarely used, when we have a convex function (data should not be too large)

Batch GD (more than 2 dimension)

cgpa, iq, gender, lpa

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3$$

$\{\beta_0, \beta_1, \beta_2, \beta_3\}$ coef to find.

If n dimension
No of coef $\{\beta_0, \beta_1, \beta_2 \cdots \beta_n\}$

β_0 → intercept

MATHEMATICAL FORMULATION

Let's say rather than n, we have 2 dimension, and 2 rows. (for understanding)

cgpa x_1	iq x_2	lpa y
8.1	93	3.2
7.5	95	3.5

$$\underset{(\text{lpa})}{y} = \beta_0 + \beta_1 \underset{(\text{cgpa})}{x_1} + \beta_2 \underset{(\text{iq})}{x_2}$$

Calculation → $(\beta_0, \beta_1, \beta_2)$

step 1) Random values
Generally intercept is started with 0
$\beta_0 = 0$, and coeff (features) $\beta_1, \beta_2 = 1$
(generally norm)

step 2: epoch = 100, lr = 0.1 (let's say)

$$\beta_0 = \beta_0 - \eta \cdot slope$$
$$\beta_1 = \beta_1 - \eta\, slope$$
$$\beta_2 = \beta_2 - \eta\, slope$$

Loss function $\Rightarrow L(\beta_0, \beta_1, \beta_2) \to$ 4D graph
1 axis pe $\longrightarrow$ Loss f^n, 1 $\to \beta_0$, 1 $\to \beta_1$, 1 $\to \beta_2$
(Loss f^n now will depend on $\beta_0, \beta_1, \beta_2$)

1 component in direction of β_0
" " " " β_1
" " " " β_2

Calculate Loss function based on each ~~values~~ coeff

$$\frac{\partial l}{d\beta_0}, \frac{\partial l}{d\beta_1}, \frac{\partial l}{d\beta_2}$$

slope wrt β_0, slope wrt β_1, " β_2

If we had n dimensions we have to calculate $(n+1)$ derivatives
$\beta_0, \beta_1 \ldots \beta_n$ no of derivatives

Page No.
Date

{rows = 2, cols = 2+1}

Calculate derivative of loss f^n w.r.t intercept.

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

$$= \frac{1}{2} \left[(y_1 - \hat{y}_1)^2 + (y_2 - \hat{y}_2)^2 \right]$$

$$\cancel{\hat{y}_i = \beta_0 + x}$$

row no		cgpa x_1	iq x_2	lpa y
x_{11}	x_{12}	8.1	93	3.2
x_{21}	x_{22}	7.5	95	3.5

1 Row ← $\hat{y}_1 = \beta_0 + \beta_1 x_{11} + \beta_2 x_{12}$

2 Row ← $\hat{y}_2 = \beta_0 + \beta_1 x_{21} + \beta_2 x_{22}$

$$= \frac{1}{2} \left[(y_1 - \beta_0 - \beta_1 x_{11} - \beta_2 x_{12})^2 + (y_2 - \beta_0 - \beta_1 x_{21} - \beta_2 x_{22})^2 \right]$$

Loss function

$$\frac{\partial L}{\partial \beta_0} = \frac{1}{2} \left[2(y_1 - \hat{y}_1) \cdot \frac{d}{d\beta_0} (\underset{0}{y_1} - \underset{(-1)}{\beta_0} - \underset{0}{\beta_1 x_{11}} - \underset{0}{\beta_2 x_{12}}) \right]$$

$$+$$

$$\left[2(y_2 - \hat{y}_2) \cdot \frac{d}{d\beta_0} (\underset{0}{y_2} - \underset{(-1)}{\beta_0} - \underset{0}{\beta_1 x_{21}} - \underset{0}{\beta_2 x_{22}}) \right]$$

$$\frac{\partial l}{\partial \beta_0} = \frac{1}{2}\left[2(y_1-\hat{y}_1)(-1) + 2(y_2-\hat{y}_2)(-1)\right]$$

$$= \frac{-2}{2}\left[(y_1-\hat{y}_1) + (y_2-\hat{y}_2)\right]$$

not cancelling we may have n dimension

Eqn

$$\frac{\partial l}{\partial \beta_0} = \frac{-2}{n}\left[(y_1-\hat{y}_1) + (y_2-\hat{y}_2) + \cdots (y_n-\hat{y}_n)\right]$$

$$\boxed{\frac{\partial l}{\partial \beta_0} = \frac{-2}{n}\sum_{i=1}^{n}(y_i-\hat{y}_i)}$$

$$\beta_0 = \beta_0 - \eta\,(\text{slope})$$

we calculating this slope

<u>calculate slope w.r.t β_1</u>

X_1	X_2	Y
8.1	93	3.2
7.6	95	3.5

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

$$L = \frac{1}{2}\left[(y_1-\hat{y}_1)^2 + (y_2-\hat{y}_2)^2\right]$$

$$L = \frac{1}{2}\left\{\left[y_1 - \beta_0 - \beta_1 X_{11} - \beta_2 X_{12}\right]^2 + \left[y_2 - \beta_0 - \beta_1 X_{21} - \beta_2 X_{22}\right]^2\right\}$$

$$\frac{\partial l}{\partial \beta_1} = \frac{1}{2}\left[2(y_1-\hat{y}_1)\frac{d}{d\beta_1}(\underbrace{y_1}_{0} - \underbrace{\beta_0}_{0} - \underbrace{\beta_1 X_{11}}_{-X_{11}} - \underbrace{\beta_2 X_{12}}_{0})\right]$$

$$+ \left[2(y_2-\hat{y}_2)\frac{d}{d\beta_1}(\underbrace{y_2}_{0} - \underbrace{\beta_0}_{0} - \underbrace{\beta_1 X_{21}}_{-X_{21}} - \underbrace{\beta_2 X_{22}}_{0})\right]$$

$$\frac{\partial l}{d\beta_1} = \frac{1}{2}\left[2(y_1-\hat{y}_1)(-x_{11}) + 2(y_2-\hat{y}_2)(-x_{21})\right.$$

Converting for n numbers

$$\frac{\partial l}{d\beta_1} = \frac{-2}{n}\left[(y_1-\hat{y}_1)(x_{11}) + (y_2-\hat{y}_2)(x_{21}) + \dots (y_n-\hat{y}_n)(x_{n1})\right]$$

no of rows

$$\boxed{\frac{\partial l}{d\beta_1} = \frac{-2}{n}\sum_{i=1}^{n}(y_i-\hat{y}_i)\,x_{i1}}$$

Similarly for β_2

$$\frac{\partial l}{d\beta_2} = \frac{-2}{n}\sum_{i=1}^{n}(y_i-\hat{y}_i)\,x_{i2}$$

generall eq^n ($\beta_1, \beta_2 \dots \beta_m$)

$$\boxed{\frac{\partial l}{d\beta_m} = \frac{-2}{n}\sum_{i=1}^{n}(y_i-\hat{y}_i)\,x_{im}}$$

This is the mathematical formula-tion.

How to implement in code (Code)

$\beta_0 = \beta_0 - \eta$ slope

$$\frac{\partial l}{\partial \beta_0}\left(\frac{-2}{n}\sum_{i=1}^{n}(y_i-\hat{y}_i)\right)$$

y_train, ypred

-2 (np.mean(y_train - ypred))

Ypred ?

$\hat{y}_1 = \beta_0 + \beta_1 X_{11} + \beta_2 X_{12} + \beta_3 X_{13}$.

$\hat{y}_2 = \beta_0 + \beta_1 x_{21} + \beta_2 x_{22} + \beta_3 X_{23}$

Can be written as :-

$$\hat{y}_1 = \beta_0 + \beta \begin{bmatrix} X_{11} & X_{12} & X_{13} \end{bmatrix}_{(1\times3)} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix}_{(3\times1)} \leftarrow \text{coeff}$$

$\therefore$ $\hat{y}$ = np.dot(coef_, X_train) + β_0

X_train shape = (353, 10)

Coef shape = (10, 1)

$\hat{Y}$ = β_0 + dot(X_train, coeff) $\Rightarrow$ (353, 1) + β_0

$\hat{y}$ = (353, 1)

X_1	X_2	y	$\hat{y}$
1	2	5	6
3	4	7	8

$y = [5, 7]$

$\hat{y} = [6, 8]$

$y - \hat{y} = [-1, -1]$

$$\frac{\partial L}{\partial \beta_1} = \frac{-2}{n} \sum_{i=1}^{n} (y_i - \hat{y}) x_{i1} \rightarrow x_{i1} = [1, 3]$$

$$= \frac{-2}{n} \left[[-1, -1] \begin{bmatrix} 1 \\ 3 \end{bmatrix} \right]$$

$$\frac{-2}{n}(-8)$$

$$= \frac{-2}{2}(-8) = -8 \text{ (single number)}$$

Similarly

$$\frac{\partial L}{\partial \beta_2} = [y - \hat{y}] \begin{bmatrix} 2 \\ 4 \end{bmatrix} \times \left(\frac{-2}{n} \right) = \text{single number.}$$

Ab ~~alag~~ eksath karne ke liye we will multiply $[y-\hat{y}]$ matrix with whole isse

$$\begin{array}{cccc} x_1 & x_2 & y & \hat{y} \\ 1 & 2 & 5 & 6 \\ 3 & 4 & 7 & 8 \end{array}$$

$$[y-\hat{y}]_{(1,2)} \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}_{(2,2)} \times \frac{-2}{n}$$

output $(1, 2)$ $[\beta_0 \quad \beta_1]$

In short trick

$$\text{np.dot}((y_i - \hat{y}_i), x\_train) \times \frac{-2}{n}$$

$y_i = 353$,

$y = 353$

$(353, 1)$ $(353, 10)$

Transpose

$(1, 353)$ dot $(353, 10)$

$(1, 10) \rightarrow (1, 10) \times \frac{-2}{n}$

$= (1, 10)$

derivatives of coefficient