Multivariable Linear Regression

X : Input matrix of shape (n, d) where
- $\rightarrow n$ = number of samples
- $\rightarrow d$ = number of features (columns)
- $\rightarrow \theta$ = weight vector (parameter) of shape $(d, 1)$ these are the cofficient need to learn.
- $\rightarrow y$ = True output vector (target values) of shap $(n, 1)$

Linear Model Prediction:

$$\hat{y} = X\theta$$

$\hat{y}$ = predicted output vector of shape $(n, 1)$

For each sample i, $\hat{y}_i = \sum_{j=1}^{d} x_{ij} \cdot \theta_j$

Define the loss function (MSE)

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

$$= \frac{1}{n} e^T e$$

$$= \frac{1}{n} (y - X\theta)^T (y - X\theta)$$

Error vector

$$e = y - \hat{y} = y - X\theta$$

shape $(n, 1)$

How ?

$$L(\theta) = \frac{1}{n} \sum_{i=1}^{n} (y_i - x_i\theta)^2 = \boxed{\frac{1}{n} \sum_{i=1}^{n} e_i^2}$$

This is just summing up squares of error for each sample.

what does $e^T e$ means

$$e = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix}_{(n \times 1)} \qquad e^T = \begin{bmatrix} e_1 & e_2 & \cdots & e_n \end{bmatrix}_{(1 \times n)}$$

$$e^T . e = \begin{bmatrix} e_1 & e_2 & \dots & e_n \end{bmatrix} . \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix}$$

$$= e_1^2 + e_2^2 + \dots\dots + e_n^2$$

$$\boxed{e^T e = \sum_{i=1}^{n} e_i^2} \qquad \boxed{e^T . e = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2}$$

$$\boxed{e = y - X\theta}$$

vector for of loss:

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

our goal is to minimize $L(\theta)$ with respect to θ

→ Find the gradient of $L(\theta)$ with respect to θ

$$\boxed{\nabla_\theta L = \frac{\partial L}{\partial \theta}}$$

To find the derivative $L(\theta)$ we have to simplify $L(\theta)$

$$L(\theta) = \frac{1}{n} (y - X\theta)^T . (y - X\theta)$$

$$= \frac{1}{n} (y^T - \theta^T X^T)(y - X\theta) \qquad \begin{bmatrix} (a-b)^T = a^T - b^T \\ (ab)^T = b^T a^T \end{bmatrix}$$

$$= \frac{1}{n} (y^T y - y^T X\theta - \theta^T X^T y + \theta^T X^T X \theta)$$

$$\boxed{L(\theta) = \frac{1}{n} (y^T y - 2\theta^T X^T y + \theta^T X^T X \theta)} \qquad \begin{bmatrix} y^T X \theta = \theta^T X^T y \\ \text{we will see it later} \end{bmatrix}$$

Final simplified Loss function

Take the derivative of cost function:

① $\frac{\partial}{\partial \theta}(y^T y) = 0 \longrightarrow$ doesn't depend on θ. so constant $\left[\frac{dc}{d\theta} = 0\right]$

② Derivative of $-2\theta^T X^T y$

$$\frac{\partial}{\partial \theta}\left(-2\theta^T X^T y\right) = -2X^T y \quad \left[\text{vector calculus} \;\; \frac{d}{dA} A^T = 1\right]$$

③ derivative of $\theta^T X^T X \theta$

$$\frac{\partial}{\partial \theta}\left(\theta^T X^T X \theta\right) = \frac{\partial}{\partial \theta}\left(X^T X \theta^T \theta\right) = 2X^T X\theta \quad \left[\frac{d}{dA}\left(A^T A\right) = 2A\right]$$

∴ Finally after combining all the derivatives

$$\nabla_\theta J = \frac{\partial L}{\partial \theta} = \left(-2X^T y + 2X^T X \theta\right) \times \frac{1}{n}$$

$$\Rightarrow \frac{\partial L}{\partial \theta} = \frac{-2}{n}\left(X^T y - X^T X \theta\right)$$

$$= \frac{2}{n}\left(X^T X\theta - X^T y\right)$$

both are okay

Express the gradient in terms of vector

$$\frac{\partial L}{\partial \theta} = \frac{-2}{n}\left(X^T y - X^T X\theta\right)$$

$$= \frac{-2}{n} \cdot X^T (y - X\theta)$$

$$= \frac{-2}{n} \cdot X^T \cdot e \quad [e = y - X\theta]$$

Final gradient $\boxed{\nabla_\theta (J(\theta)) = \frac{-2}{n} X^T \cdot e}$

Proof $y^T X\theta = \theta^T X^T y$

basically we have to show that they are symetric.

y shape $(n \times 1)$

y^T shape $(1 \times n)$

X " $(n \times d)$

X^T " $(d \times n)$

θ " $(d \times 1)$

$\theta^T \Rightarrow (1 \times d)$

$$LHS = y^T . X . \theta$$

$$= \underbrace{(1 \times n)(n \times d)}(d \times 1)$$

$$= (1 \times d)(d \times 1)$$

$= (1 \times 1) \longrightarrow$ scalar (a single value)

$$RHS = \theta^T X^T y$$

$$= \underbrace{(1 \times d)(d \times n)}(n \times 1)$$

$$= (1 \times n)(n \times 1)$$

$= (1 \times 1) \longrightarrow$ again a single value

$\boxed{RHS = LHS \text{ (proved)}}$

gradient algo steps:

(i) take any values of thetas

(ii) Do iterations n times

(iii) At each iteration Find the gradient and update theta

$$\theta := \theta - \alpha \nabla_\theta J(\theta)$$

learning rate $\longrightarrow$ a very small number to make change of θ stable.

Till now we calculated the gradient descent without the bias term to make things simple.

now we will just add the bias term and we will find the gradient with respect to theta and bias. But guess we already calculted the gradient with respect to theta.

now prediction, $\hat{y} = x\theta + b$

$b \rightarrow$ bias term (intercept)

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

$$= \frac{1}{n} \sum_{i=1}^{n} (y_i - (x_i\theta + b))^2$$

$$= \frac{1}{n} e^T . e \qquad [e = y - x\theta - b]$$

$$= \frac{1}{n} (y - x\theta - b)^T (y - x\theta - b)$$

gradient w.r.t. θ

$$\nabla_\theta L = \frac{\partial L}{\partial \theta} = -\frac{2}{n} x^T e$$

$$\boxed{\frac{\partial L}{\partial \theta} = -\frac{2}{n} x^T . (y - x\theta - b)}$$

gradient with respect b

$$\frac{\partial L}{\partial b} = \frac{\partial}{\partial b} \left[\frac{1}{n} \sum_{i=1}^{n} (y_i - x_i\theta - b)^2\right]$$

$$= \frac{2}{n} \sum_{i=1}^{n} (y_i - x_i\theta - b) . (-1)$$

$$= \frac{-2}{n} \sum_{i=1} e_i \quad = -\frac{2}{n} \sum_{i=1}^{n} (y - x\theta - b)$$