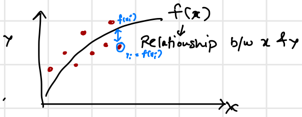
Simple Regression :-

feature / Input — sq.ft (x)

Predict/output — \$ (y)

predict (y) using (x) Input/features.



Regression Model :

$y_i = f(x_i) + \varepsilon_i$

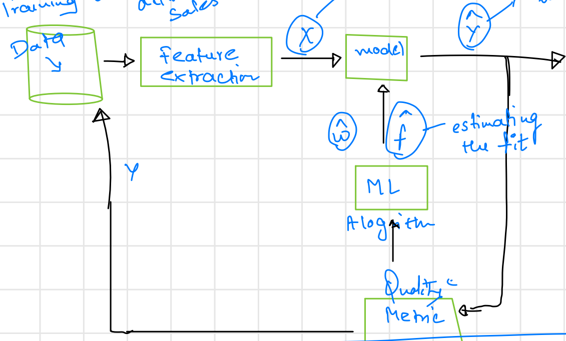
$E[\varepsilon_i] = 0$ ← Expected value

y_i is expected $+ \,\&\, (-)$

which model ?

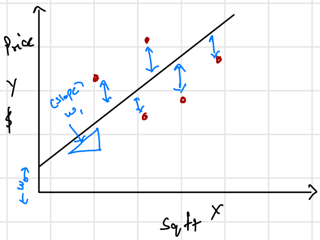
$(y_i = w_0 + w_1 x_i + \varepsilon_i)$

Flow Chart → (Training (house ids, sq ft & attributes Sales Price))



Simple Linear Regression

- RSS [Residual Sum of Squares]



$$RSS(w_0, w_1) = \left(\$_{house_1} - [w_0 + w_1\, sqft_1]\right)^2 + \left(\$_{house_2} - [w_0 + w_1\, sqft_2]\right)^2 + \left(\$_{house_3} - [w_0 + w_1\, sqft_3]\right)^2 + \dots + \left[\$_{house\ n} - [w_0 + w_1\, sqft\ n]\right)^2$$

$$\boxed{RSS(w_0, w_1) = \sum_{i=1}^{n} \big(\underbrace{y_i - [w_0 + w_1 x_i]}_{\text{Error}}\big)^2}$$

estimated line fit = Smallest RSS

$[y_i = w_0 + w_1 x_i + \varepsilon_i]$

w_0, w_1 — Parameters

$\hat{w}_0, \hat{w}_1$ — Estimated Parameters.

$\hat{f}\ [\hat{y}_p = \hat{w}_0 + \hat{w}_1 x_{sqft}]$ — Predicted Price.

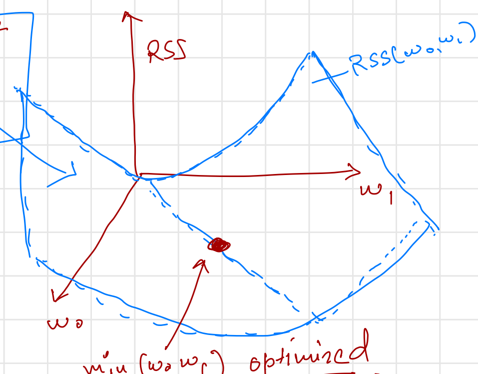
Parameters :

w_0 - Intercept $\quad \hat{y} = \hat{w}_0 \Rightarrow (x = 0)$

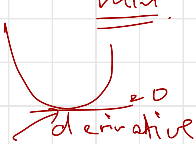
w_1 - slope (Predicted change in O/p / unit change in I/p)

ML Algorithm :-

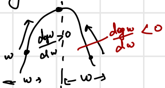
$$\boxed{\min_{w_0, w_1} \underbrace{\sum_{i=1}^{n} \left(y_i - (w_0 + w_1 x_i)\right)^2}_{RSS}}$$



Derivative — For RSS min



Finding the (max via hill climbing)



Optimum Value.

Algorithm min; Hill descent

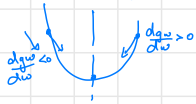
while not converged:

$$w^{(t)} + \eta \frac{dg(w)}{dw}$$

iteration t — step size

while not converged:

$$w^{(t)} - \eta \frac{dg(w)}{dw}$$



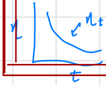
η – step size
– learning rate.

fixed step size work in strongly convex fun.

$$\eta_t = \frac{\alpha}{t}$$

$$\eta_t = \frac{\alpha}{\sqrt{t}}$$

Varing η



$$\nabla g(w) = \begin{bmatrix} \frac{\partial g}{\partial w_0} \\ \frac{\partial g}{\partial w_1} \\ \vdots \\ \frac{\partial g}{\partial w_p} \end{bmatrix}_{(p+1)}$$

$(w_0\ w_1\ w_2 \cdots w_p)$

Partial derivative

Example.

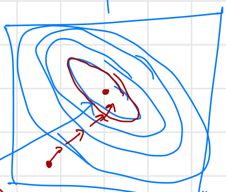
$$g(w) = 5w_0 + 10w_0 w_1 + 2w_1^2$$

$$\begin{bmatrix} \frac{\partial g}{\partial w_0} \\ \frac{\partial g}{\partial w_1} \end{bmatrix} = \begin{bmatrix} 5 + 10w_1 \\ 10w_0 + 4w_1 \end{bmatrix}$$

$$\nabla g(w) = \begin{bmatrix} 5 + 10w_1 \\ 10w_0 + 4w_1 \end{bmatrix}$$

some vector in space.

3-D plots



$g(w_0, w_1)$ contour plots.

Same value.

$$\boxed{w^{(t+1)} \leftarrow w^{(t)} - \eta \nabla g(w^{(t)})}$$

$$\begin{bmatrix} \vdots \end{bmatrix} \leftarrow \begin{bmatrix} \vdots \end{bmatrix} - \eta \begin{bmatrix} \vdots \end{bmatrix}$$

$$\|\nabla g(w)\| < \epsilon$$

converged.

$$\frac{d}{dw} \sum_{i=1}^{n} g_i(w) = \frac{d}{dw} \left(g_1(w) + g_2(w) \cdots + g_n(w) \right)$$

$$\boxed{= \sum_{i=1}^{n} \frac{d}{dw} g_i(w)}$$

$$RSS(w_0, w_1) = \sum_{i=1}^{n} \left(y_i - (w_0 + w_1 x_i)\right)^2$$

$$g_i(w_0) = -\sum_{i=1}^{n} 2 \cdot \left[y_i - (w_0 + w_1 x_i) \right]$$

$$g_i(w_1) = -\sum_{i=1}^{n} 2 \cdot \left[y_i - (w_0 + w_1 x_i) \right] \cdot x_i$$

$$\nabla RSS(w_0 w_1) = \begin{bmatrix} -2 \sum_{i=1}^{n} \left[y_i - [w_0 + w_1 x_i] \right] \\ -2 \sum_{i=1}^{n} \left[y_i - [w_0 + w_1 x_i] \right] \cdot x_i \end{bmatrix}$$

$$\hat{w}_0 = \frac{\sum_{i=1}^{n} y_i}{N} - \hat{w}_1 \frac{\sum_{i=1}^{n} x_i}{N}$$

avg. house sales price.

estimate of slope

avg. sqft

Gradient Descent Approach :-

$$\nabla RSS(w_0, w_1) = \begin{bmatrix} -2\sum_{i=1}^{n}[y_i - \hat{y}_i(w_0, w_1)] \\ -2\sum_{i=1}^{n}[y_i - \hat{y}_i(w_0, w_1)]x_i \end{bmatrix}$$

Algorithm :-

while not converged :-

$$\begin{bmatrix} w_0^{t+1} \\ w_1^{t+1} \end{bmatrix} \longrightarrow \begin{bmatrix} w_0^{t} \\ w_1^{t} \end{bmatrix} + 2\eta \begin{bmatrix} \sum_{i=1}^{n}[y_i - \hat{y}_i(w_0^t, w_1^t)] \\ \sum_{i=1}^{n}[y_i - \hat{y}_i(w_0^t, w_1^t)]x_i \end{bmatrix}$$