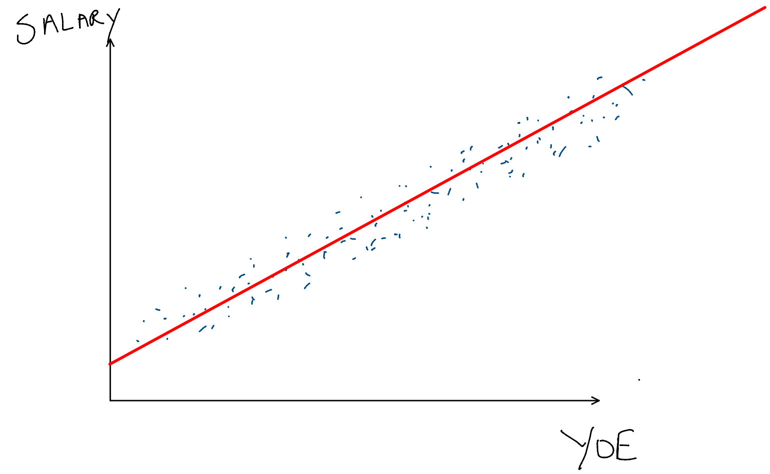
Simple Linear Regression

18 January 2024 11:44 AM

- Regression Algorithm
- SLR is used when there is only one independent feature
- The relationship between independent feature and dependent feature should be Linear

YOE	SALARY
–	–
–	–
–	–
–	–
–	–
IF	DF



- The relationship between independent feature YOE and dependent feature Salary is linear.
- SLR can be used to capture the pattern.
- Since the pattern is linear it can be captured by a function similar to straight line.

$$\boxed{\hat{y} = \beta_0 + \beta_1 X}$$

$\hat{y}$ = Predicted value

X = Input

β_0, β_1 = Parameters

- The goal is to find best values of parameters B0 and B1.
- The best values are the ones which give us predictions as close as possible to the actual values (Y^≊Y).
- Once the best values have been learned, the model/equation can be used to make predictions for new cases.

A metric/criteria is required to decide how close our predictions are wrt actual values.

* COST FUNCTION (LOSS FUNCTION) :-

$$\boxed{J(y, \hat{y}) = \frac{1}{n} \Sigma (y - \hat{y})^2}$$

- A Cost function is a measure of how good the predictions are wrt actual values.
- A higher value of the cost function indicates that the predictions are far from actual values.
- The cost function used for regression is Mean Squared Error(MSE) also known as Residual Sum of Squares(RSS).

- Best values of B0 and B1 are those for which the value of cost function is minimum.

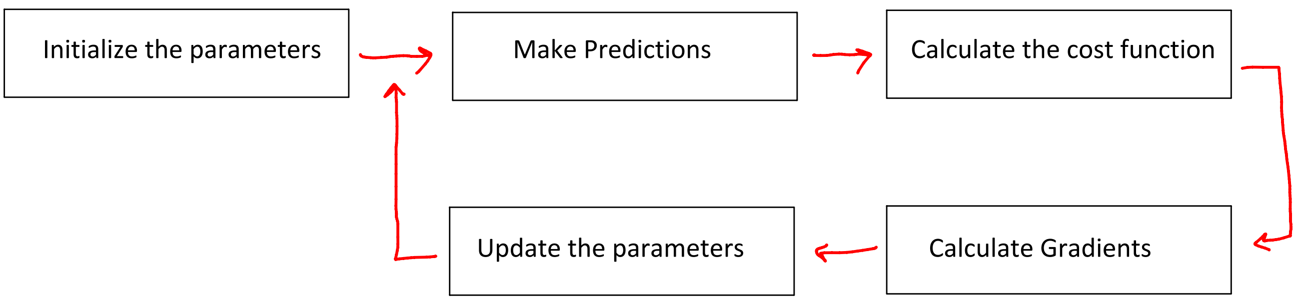
- Best value of B0 and B1 will be at the minima of the cost function.

How to find the best values for B0 and B1?

- Ordinary Least Squares(OLS)
- Gradient Descent

Gradient Descent:

- Initialize the parameters
- Make Predictions
- Calculate the cost function
- Calculate Gradients
- Update the parameters
- Repeat the steps 2 to 5



i) INITIALIZE THE PARAMETERS

ii) MAKE PREDICTIONS :-

$$\hat{y} = \beta_0 + \beta_1 X$$

iii) CALCULATE THE COST FUNCTION :-

$$J(y,\hat{y}) = \frac{1}{m}\Sigma(y-\hat{y})^2$$

$$= \frac{1}{m}\Sigma(y-(\beta_0+\beta_1 x))^2$$

iv) CALCULATE GRADIENTS :-

$$\frac{\partial J}{\partial \beta_0} = \frac{2}{m}\Sigma(y-(\beta_0+\beta_1 X))(-1)$$

$$= \frac{2}{m}\Sigma(y-\hat{y}).(-1)$$

$$\frac{\partial J}{\partial \beta_1} = \frac{2}{n}\sum\left(y-(\beta_0+\beta_1 x)\right).(-x)$$

$$= \frac{2}{n}\sum(y-\hat{y}).(-x) \quad ✓$$

V) UPDATE THE PARAMETERS :-

> $\beta_0 = \beta_0 - \alpha \dfrac{\partial J}{\partial \beta_0}$

> $\beta_1 = \beta_1 - \alpha \dfrac{\partial J}{\partial \beta_1}$

CASE 1 : $\dfrac{\partial J}{\partial \beta}$ is +ve

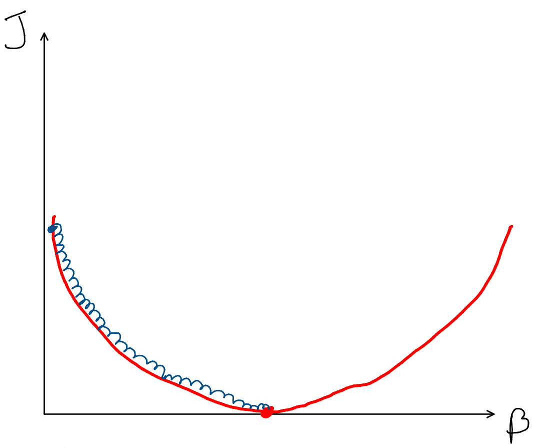
$J \propto \beta$, $\beta\uparrow : J\uparrow$, $\beta\downarrow : J\downarrow$ — $\beta\downarrow$

CASE 2 : $\dfrac{\partial J}{\partial \beta}$ is −ve

$J \propto \frac{1}{\beta}$, $\beta\uparrow : J\downarrow$, $\beta\downarrow : J\uparrow$ — $\beta\uparrow$

---

* EFFECT OF LEARNING RATE (α) :-



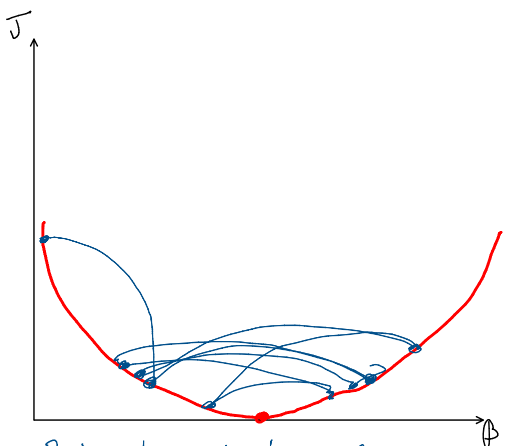
1) Very Small

> The updates will be very small

> Reduction in J will be very small

> A lot of time to reach the minima

> UNDERSTEPPING

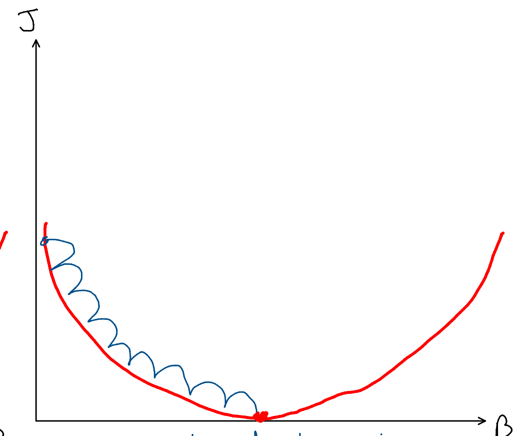


2) Very Large

> Very large updates to the parameters

> May keep missing

> OVERSTEPPING



3) Perfect Learning Rate