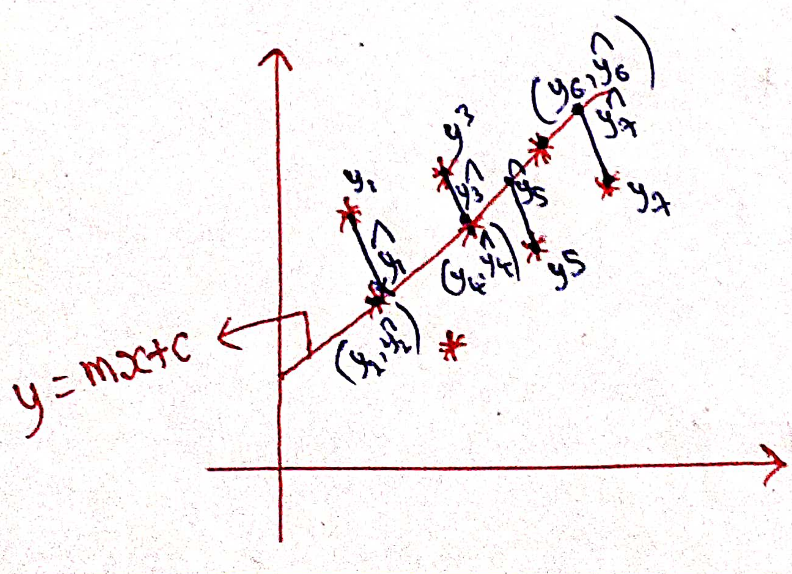
ML Algorithms

* Simple Linear Regression:

* It is one of the main Algorithm in Machine Learning, which is used for Supervised Learning.
* It can be used, if we have one Independent and one Dependent feature.
* It tries to find the best fit line which will have low cost of error from all the points.



where,

$y_1, y_2, y_3, y_4, y_5, y_6$ and y_7 are original points

And

$\hat{y}_1, \hat{y}_2, \hat{y}_3, \hat{y}_4, \hat{y}_5, \hat{y}_6$ and $\hat{y}_7$ are Fitted points

* The best fit line is represented by the formulae, $y = mx + c$

where,

y is predicted point (Dependent point)

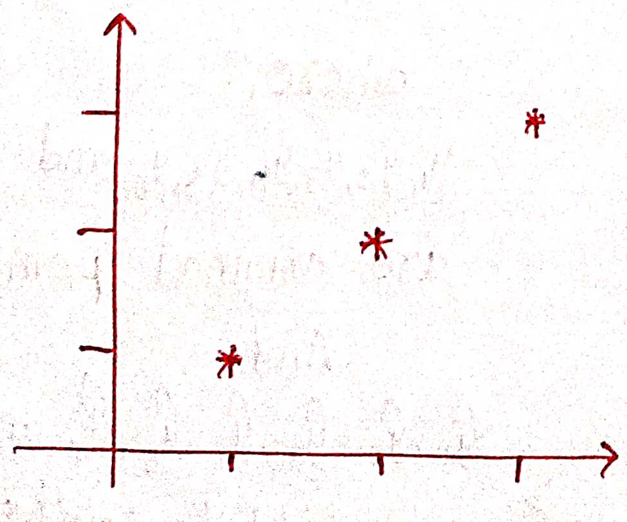
x is Given point (Independent point)

m is slope

c is Intercept (point of y at which Line meets x at zero)

Procedure of Finding best fit Line :

Lets take the points $(1,1)$, $(2,2)$ and $(3,3)$



* Let's find y by using mx+c

case 1: Let's consider m = 1 and c = 1

we have x as 1, 2 and 3

$y_1 = m(x) + c$
$= 1(1) + 1$
$= 2$

$y_2 = m(x) + c$
$= 1(2) + 1$
$= 3$

$y_3 = m(x) + c$
$= 1(3) + c$
$= 4$

we got $\hat{y}$ values as 2, 3 and 4.

our original values of y are 1, 2 and 3.

Let's find cost function using the formulae:

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

where n = No of points

y_i = original y value.

$\hat{y}_i$ = Fitted y value.

$$\Rightarrow \frac{1}{3} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \Rightarrow \frac{1}{3} \times \left((2-1)^2 + (3-2)^2 + (4-3)^2\right)$$

$$\Rightarrow \frac{1}{3} \times (1+1+1) \Rightarrow \frac{3}{3} \Rightarrow 1$$

i.e, at $m=1$ and $c=1$,

we got cost function as $\boxed{1}$ i.e, the error is 1

case (2):

Let's take $m=0$ and $c=1$

$y_1 = m(x)+c$
$= 0(1)+1$
$= 1$

$y_2 = m(x)+c$
$= 0(2)+1$
$= 1$

$y_3 = m(x)+c$
$= 0(3)+c$
$= 1$

$$\text{cost Function} = \frac{1}{n}\left((1-1)^2+(2-1)^2+(3-1)^2\right)$$
$$= \frac{1}{3}(0+1+4)$$
$$= 5/3$$

case (3):

Lets take $m=1$ and $c=0$

$y_1 = m(x)+c$
$= 1(1)+0$
$= 1$

$y_2 = m(x)+c$
$= 1(2)+0$
$= 2$

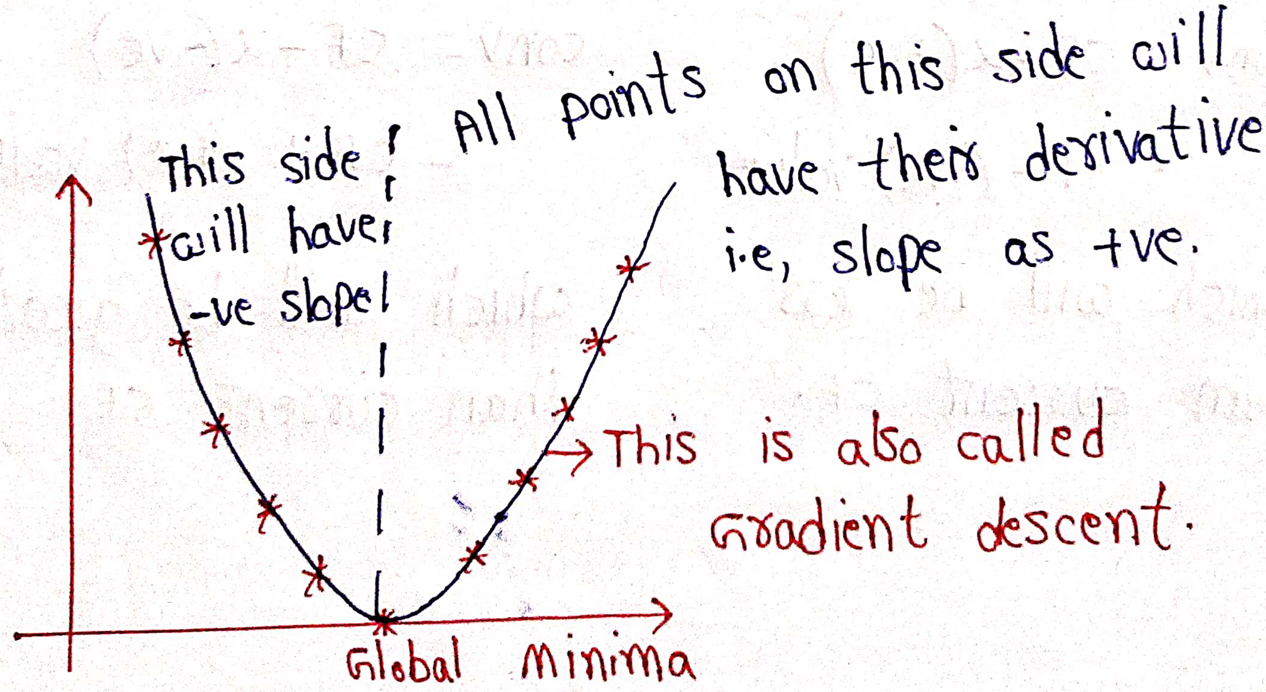
$y_3 = m(x)+c$
$= 1(3)+0$
$= 3$

$$\text{cost function} = \frac{1}{3}\left((1-1)^2+(2-2)^2+(3-3)^2\right)$$
$$= 0/3$$
$$= 0$$

→ out of case 1, case 2 and case 3, we got Global minima cost function at $m=1$ and $c=0$ where we got 'zero'.

* So while predicting $y = mx+c$, we can take $m=1$ and $c=0$ for better predictions.

How all the cost functions Looks on Graph:



* while getting cost functions, we will use something called as convergence to get the global minima point.

convergence ⇒ $CF - \alpha(\text{slope})$

where CF = current cost Function.

α = Learning Rate.

slope = Derivative.

If we have +ve Derivative

$\text{conv} = CF - \alpha(+ve)$

$= CF - (-ve)$ value

which will be less than current CF.

If we have −ve Derivative

$\text{conv} = CF - \alpha(-ve)$

$= CF + (+ve)$ value

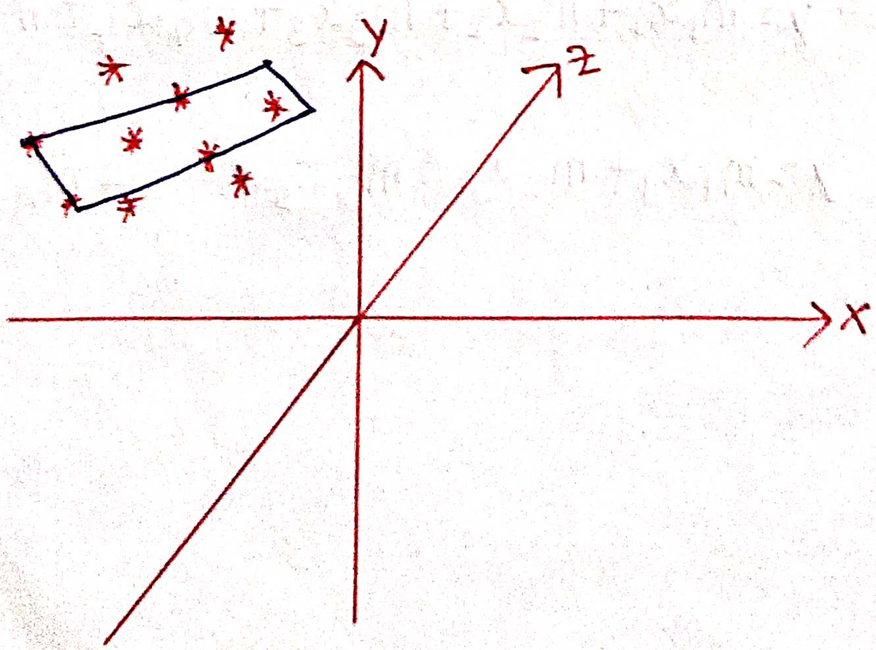
which will be greater than current CF.

* while taking Learning rate (α) if it is very high, there may be times it can't find better Global minima.

* But if we take it less, although it moves slowly, But can find better Global minima.

Multi Linear Regression:

* It is same as simple Linear Regression, but will be used if we have more than 1 Independent Feature.



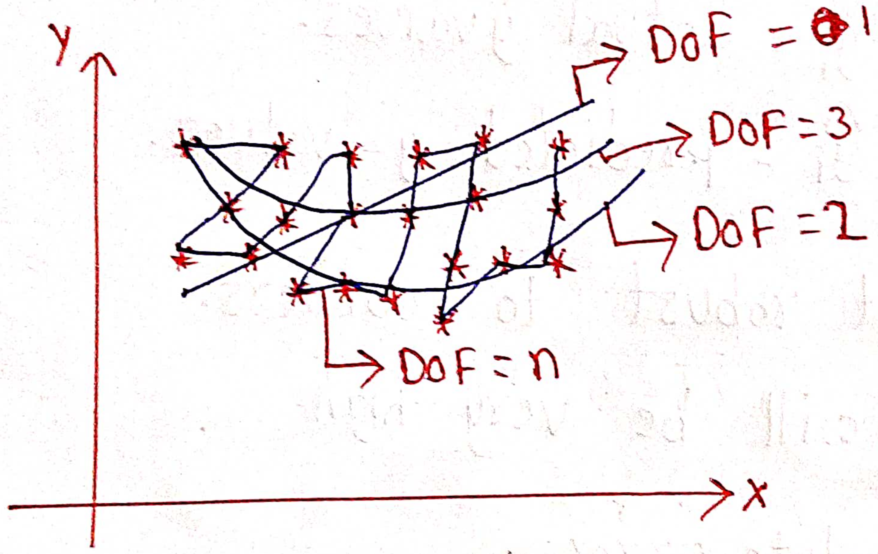
* Instead of 2D Line, we'll use 3D Plane box here to fit the points.

* Formulae for plane depends on number of Independent Features.

Independent Features	Formulae
2	$y = m_1x_1 + m_2x_2 + c$
3	$y = m_1x_1 + m_2x_2 + m_3x_3 + c$
5	$y = m_1x_1 + m_2x_2 + m_3x_3 + m_4x_4 + m_5x_5$
n	$y = m_1x_1 + m_2x_2 + m_3x_3 + \cdots + m_nx_n$

Polynomial Regression:

* It is used while data is non-Linear.
* we will have a new term in it called Degree of Freedom.



* Higher the DoF, Higher (or) best the best fit Line will be.

Polynomial Degree	Formulae
0	$m_0 x_1^0$
1	$m_0 x_1^0 + m_1 x_1^1$
2	$m_0 x_1^0 + m_1 x_1^1 + m_2 x_1^2$
3	$m_0 x_1^0 + m_1 x_1^1 + m_2 x_1^2 + m_3 x_1^3$

Types of cost Functions:

1) Mean squared error

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

where, n = no of data points.

y_i = actual y values.

$\hat{y_i}$ = predicted y values.

* It's not robust to outliers.
* values will be very high.

2) Mean Absolute Error:

$$\frac{1}{n}\sum_{i=1}^{n}(|y_i - \hat{y_i}|)$$

* It's robust to outliers
* values will be less.

Root Mean squared error:

* It's square root of Mean squared Error.

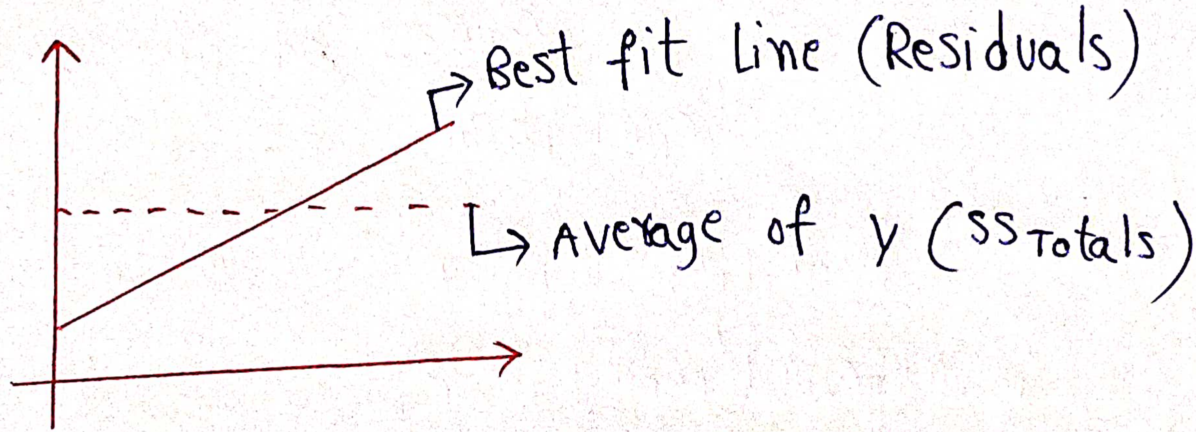
$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

* It's not robust to outliers.
* values will be less.

Accuracy checks:

→ In Regression (Linear), we'll check accuracy of our models by r^2 and adjusted r^2's.

$$R^2 = 1 - \frac{SS_{Residuals}}{SS_{Total}}$$



Adjusted R^2:

$$1 - (1-R^2)(N-1) / N-P-1$$

* P = No. of Independent features