→ Machine Learning Algorithms :-

1) Linear Regression :-

LR → Supervised

LR → Polynomial LR

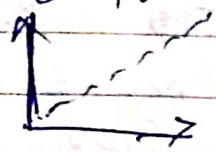
LR → Simple LR

LR → Multiple LR

Simple → 1 i/p | 1 o/p

Suppose,

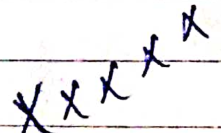
cgpa	package
6.66	3.01.



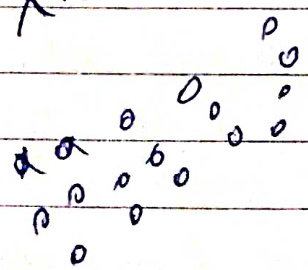
Multiple LR → One or more i/p | 1 o/p.

cgpa	gender	12th Marks	state	package

Polynomial :- Suppose if your dataset is not ~~p~~ Linear, then we use polynomial Regression.



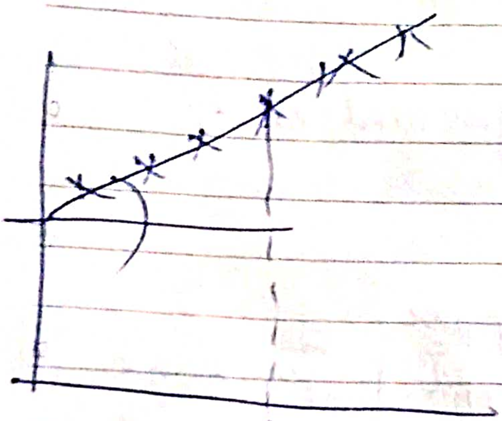
→ Completely linear



→ Sort of linear

Why datasets are not completely linear or sort of linear?

→ Real world datasets are not linear as they are affected by various parts & factors. (these are the stochastic errors).



$y = mx + b$

m = slope

b = y-intercept

To handle the sort of linear data we'll draw a ~~sort of~~ best fit line.

→ Basically, finds the value of m & b, so that it makes less error.

b = offset, doesn't makes the value to zero.

(m, b)

Closed form sol^n

Non-Closed form sol^n.

$\boxed{\text{OLS}}$ → Scikit learn uses OLS, to find the value of m & b.

1) Closed form Sol^ns :- If we can create any formula & find the corresponding value of the variable, then that is closed form sol^n.

For ex. $ax^2+bx+c = 0$

$$x = \frac{b \pm \sqrt{b^2-4ac}}{\quad}$$

Ex:- OLS

→ kind of direct formula

2) Non-closed form sol^ns :- Approximation based sol^ns.

Ex: Gradient descent

→ So, if there are less dimensions, we'll use OLS (direct formula method / closed form sol^n), if we are working on higher dimensions, we'll use Gradient descent.

→ Scikit learn Linear Regression uses OLS internally to find the values of m & b.

→ SGD Regressor uses, GD Gradient descent internally.

Y = package
X = cgpa.

OLS :-

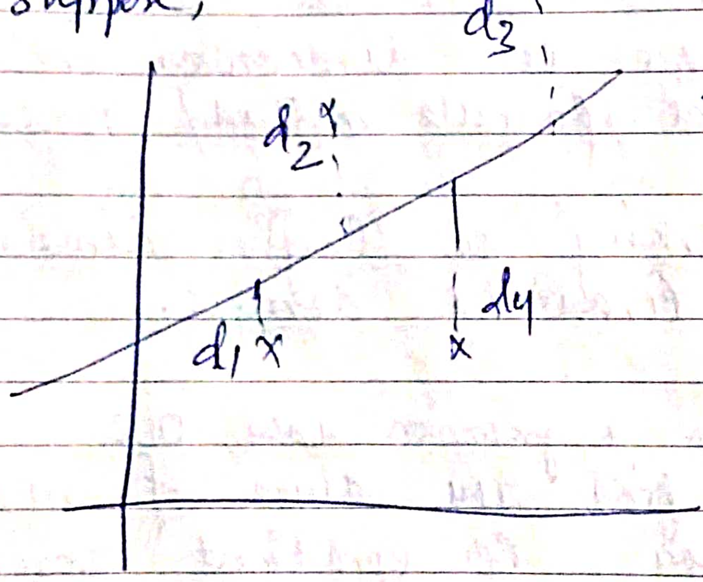
$$b = \bar{y} - m\bar{x}$$

where,

$$m = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

$\bar{x}$, $\bar{y}$ $\rightarrow$ mean values of x and y.

Suppose,



$E = d_1 + d_2 + d_3 + \dots + d_4$

$E = d_1^2 + d_2^2 + d_3^2 + \dots + d_4^2$

$$\boxed{E = \sum_{i=1}^{n} d_i^2}$$

$\rightarrow$ The idea of squaring is due to:

1) To make the values positive / decision positive

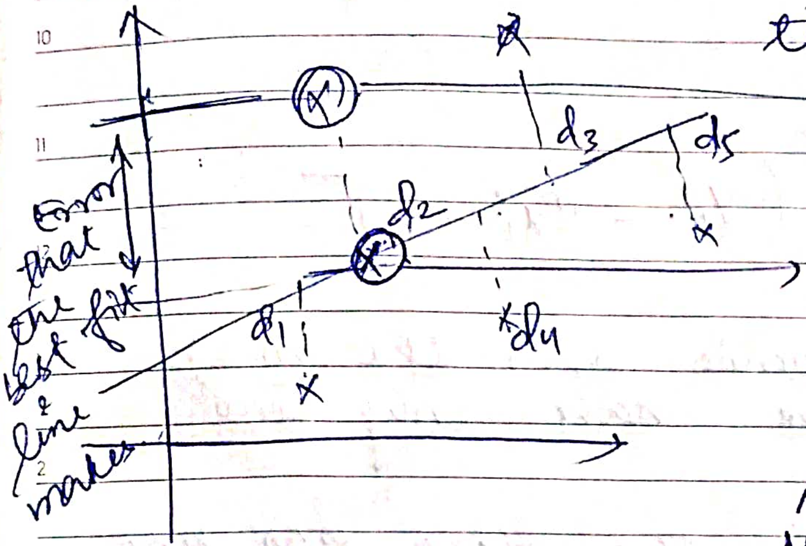
We don't use modulus as

$R_1 \rightarrow$ modulus is not differentiable at origin.

$R_2 \rightarrow$ We need to peanalise the outliers to make a better model.

$$\boxed{E = \sum_{i=1}^{n} d_i^2}$$ → Error/Loss function

(m, b) → gives the value of m & b, that ~~mo~~ minimizes the value of the eqn.



→ but it is here

& actual point should be here ~~to~~ according to best fit line.

$\hat{y}_i$ → prediction value.

So, $d_i = (y_i - \hat{y}_i)$

(actual − predicted)

Thus, error formula becomes,

Total Error → $$\boxed{E = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2}$$ → we need to find such a value that gives the min^m value

or, we need to find such a line that minimizes the value of the equation.

Total Error $\Rightarrow E = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$.

Avg. Total Error $\Rightarrow E = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$

$\hat{y}_i = mx_i + b$

$E(m,b) = \sum_{i=1}^{n} (y_i - mx_i - b)^2$

The function depends upon the (m, b) values, as, we have only two options.

1) Either we fix the slope and move the intercept.
2) or fix the intercept & ~~move~~ change the value.

To get the minimized value, for the loss function.

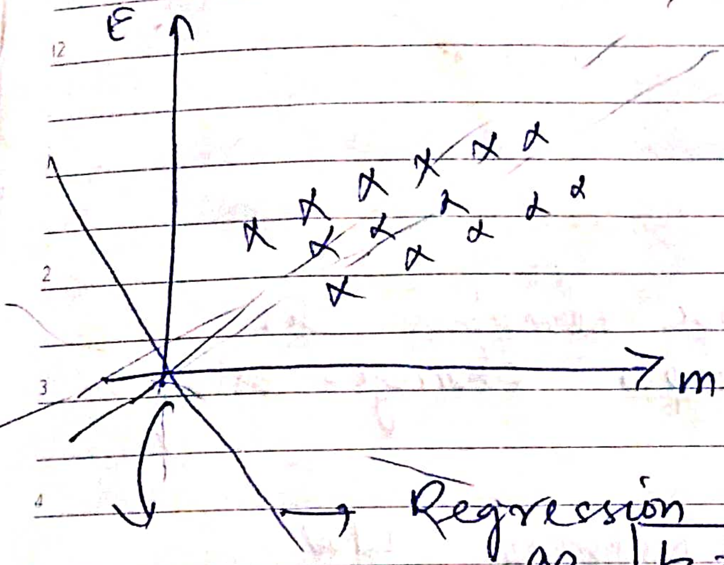
$Y = f(x) \rightarrow$ i.e. Y is a function of x.

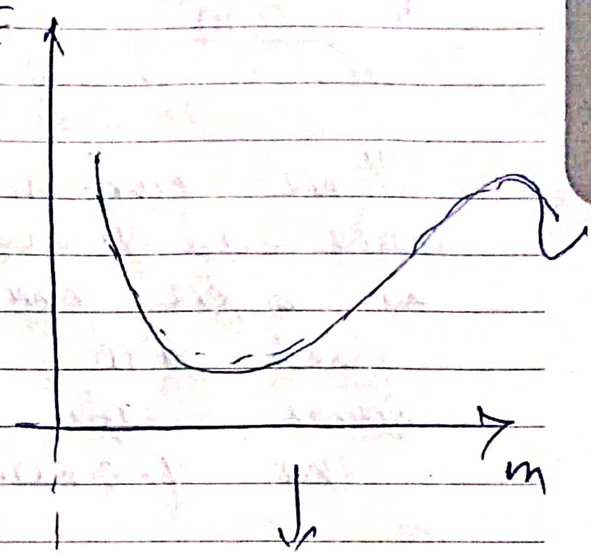
Suppose, $\boxed{b=0}$, then the error function

$$E(m,b) = \sum_{i=1}^{n} (y_i - m x_i - b)^2$$

becomes
$\Downarrow$

$$E(m) = \sum_{i=1}^{n} (y_i - m x_i)^2$$





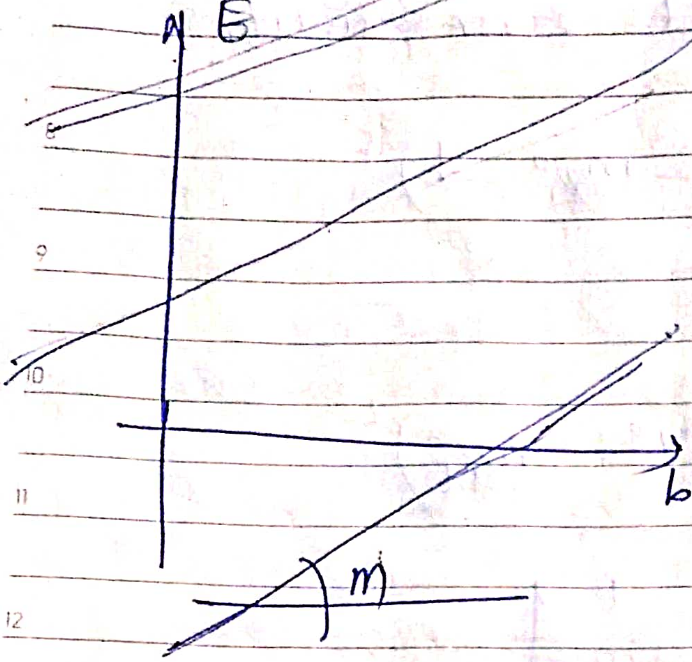
Regression line as $\boxed{b=0}$

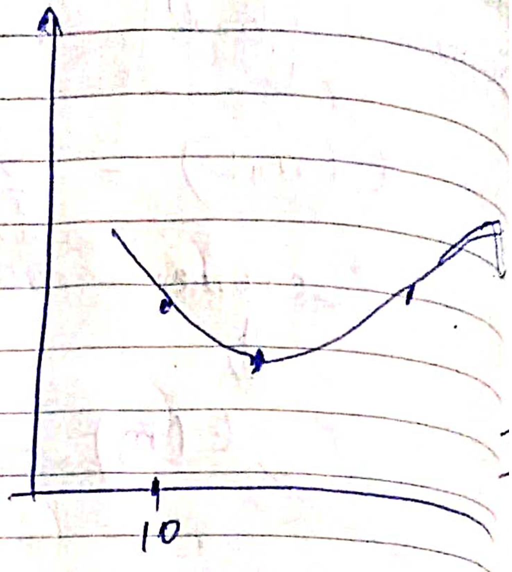
Rotates along the origin.

after we rotate the best fit line at $\boxed{b=0}$ along origin.

Now, ~~len~~ if we make, $m =$ constant

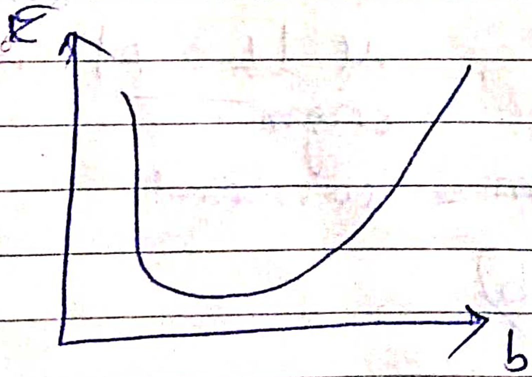
$$E(b) = \sum_{i=1} (y_i - x_i - b)$$

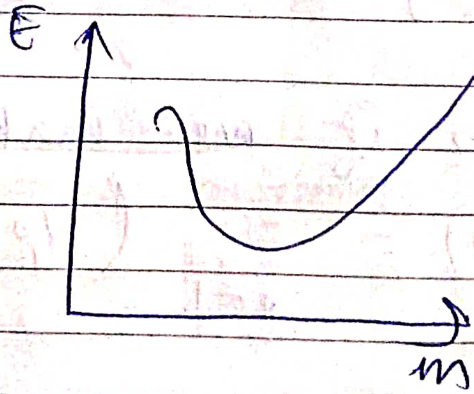




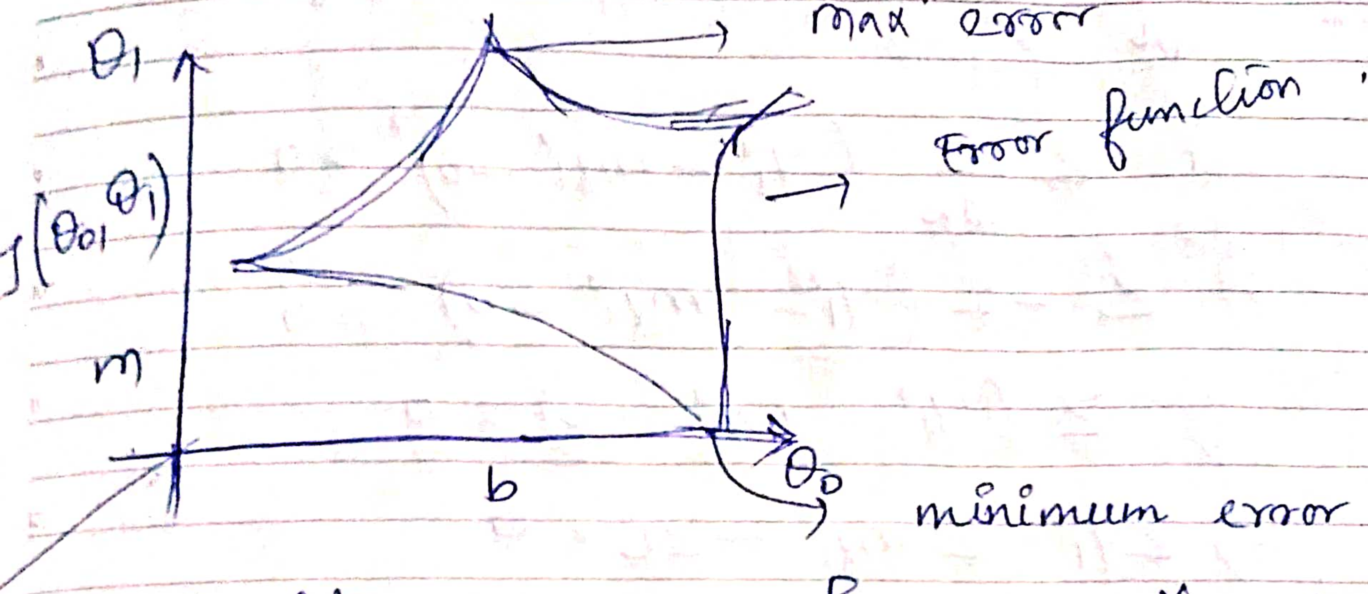
↓ we can only move up & down as we have fixed 'b' 'm', so, slope remains the same, only intercepts changes along the y-axis.

So, roughly, the relation b/w E & b & E & m is :





Lets plot a 3-D plot, including the two.



To find the value of maxima & minima, we use, differentiation, of maxima & minima.

minima = 0, (we know)

so, we'll calculate the derivative for maxima & minima for $E(m)$ & $E(b)$.

$$\frac{dE}{dx} = 0$$

$$f(x, y) \longrightarrow \frac{\partial E}{\partial m} = 0, \quad \frac{\partial E}{\partial b} = 0$$

we get two eq^ns,

& hence calculate (m & b).

$$\frac{\partial E}{\partial b} = \frac{\partial}{\partial b} \sum_{i=1}^{n} (y_i - mx_i - b)^2 = 0$$

$$\Rightarrow \sum \frac{\partial}{\partial b} (y_i - mx_i - b)^2 = 0$$

$$\Rightarrow \sum -2 (y_i - mx_i - b) = 0$$

$$\Rightarrow \sum (y_i - mx_i - b) = 0$$

$$\Rightarrow \sum y_i - \sum mx_i - \sum b = 0$$

$$\Rightarrow \frac{\sum y_i}{n} - \frac{\sum m x_i}{n} - \frac{\sum b}{n} = \frac{0}{n}$$

($\because$ n is the no. of students or samples)

$$\Rightarrow \bar{Y} - m\bar{x} - \frac{nb}{n} = 0$$

$$\Rightarrow \bar{Y} - m\bar{x} = b$$

$$\Rightarrow \boxed{b = \bar{Y} - m\bar{x}}$$

Fitting the terms in b in the loss/error eqn :—

$$E = \sum (y_i - m x_i - \bar{y} + m\bar{x})^2$$

$$\frac{\partial E}{\partial m} = \sum \frac{\partial}{\partial m} (y_i - m x_i - \bar{y} + m\bar{x})^2 = 0$$

$$\Rightarrow \sum 2 (y_i - m x_i - \bar{y} + m\bar{x})(-x_i + \bar{x}) = 0$$

$$\Rightarrow \sum -2 (y_i - m x_i - \bar{y} + m\bar{x})(x_i - \bar{x}) = 0$$

$$\Rightarrow \sum (y_i - m x_i - \bar{y} + m x)(x_i - x) = 0$$

$$\Rightarrow \sum [(y_i - \bar{y}) - m(x_i - \bar{x})](x_i - \bar{x}) = 0$$

$$\Rightarrow \sum [(y_i - \bar{y})(x_i - \bar{x}) - m(x_i - \bar{x})^2] = 0$$

$$\Rightarrow \sum (y_i - \bar{y})(x_i - \bar{x}) = m (x_i - \bar{x})^2 = 0$$

$$\Rightarrow \sum (y_i - \bar{y})(x_i - \bar{x}) = m \sum (x_i - \bar{x})^2$$

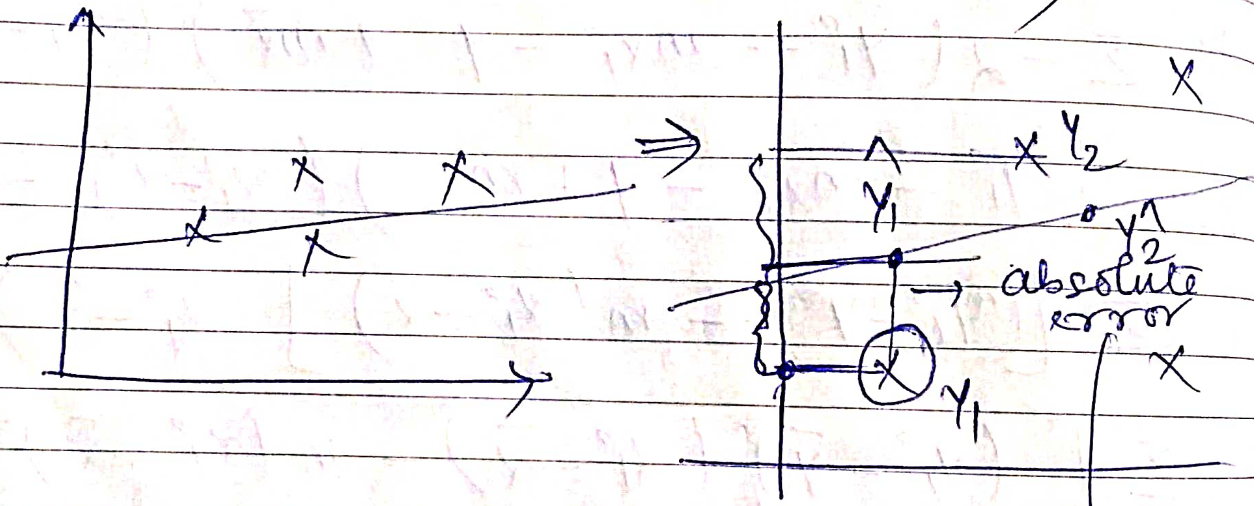
$$m = \sum_{i=1} (x_i - \bar{x})(y_i - \bar{y}) = m \sum (x_i - \bar{x})$$

$$\boxed{m = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}}$$

Regression Metrics :-

1) MAE
2) MSE
3) RMSE
4) R^2 Score
5) Adjusted Score.

1). MAE :- (Mean absolute Error)



Not considering the sign.

$$\frac{|Y_1 - \hat{Y}_1| + |Y_2 - \hat{Y}_2| + \dots + |Y_n - \hat{Y}_n|}{n}$$

$$mae = \frac{\sum_{i=1}^{n} |Y_i - \hat{Y}_i|}{n}$$

Advantage:

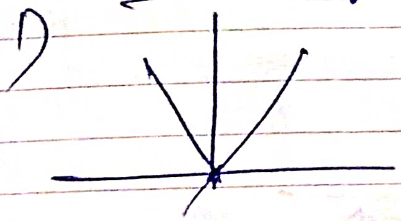
1) MAE should ~~be~~ have min^m error.

mae o/p unit = model o/p unit

same unit.

2) Robust to outliers (Can handle outliers easily)

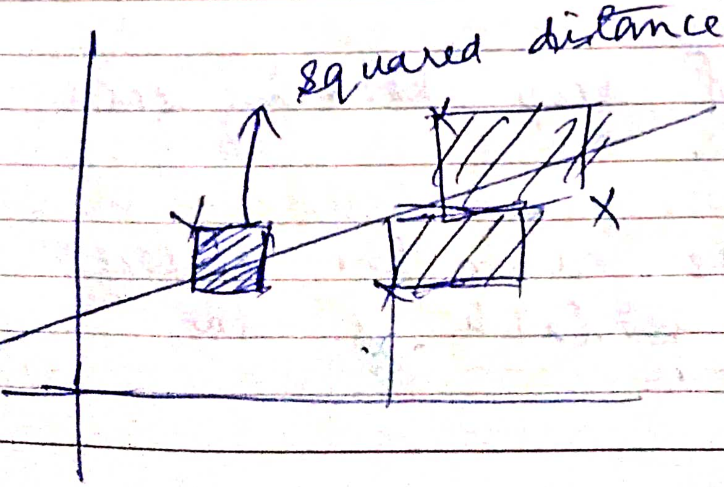
Disadvantage:

1) 

Not differentiable at zero.

MSE :-

$$MAE = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n}$$



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

→ Square function is differentiable at all points

The square represents → $(y_i - \hat{y}_i)^2$

→ Advantage :-
1) The Loss function is differentiable.

$y - (lpa) \rightarrow y^2, (lpa)^2$

2) Penalise the outliers too much. Not Robust to outliers, as the square will be too much.

3) RMSE: $= \sqrt{mse}$

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

Benefit : 1) Same unit

Disadv) Not that Robust to outliers

In most cases of deep learning RMSE is mostly used.

MSE, MAE, depends upon the content of the output variable & the problem statement.