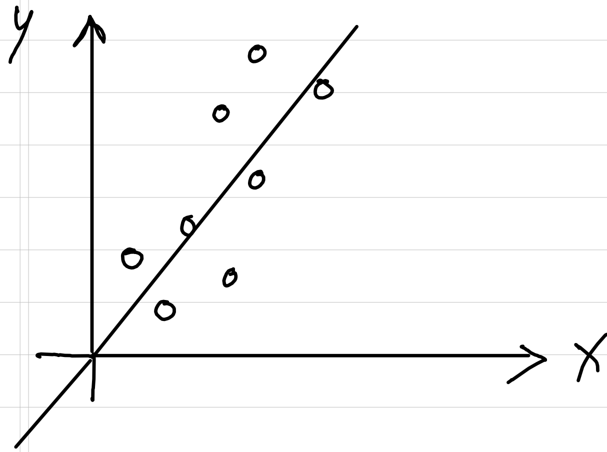
Linear Regression

We have a few datapoints. We want to fit a line through the points



We can use this line to predict a variable Y given input X

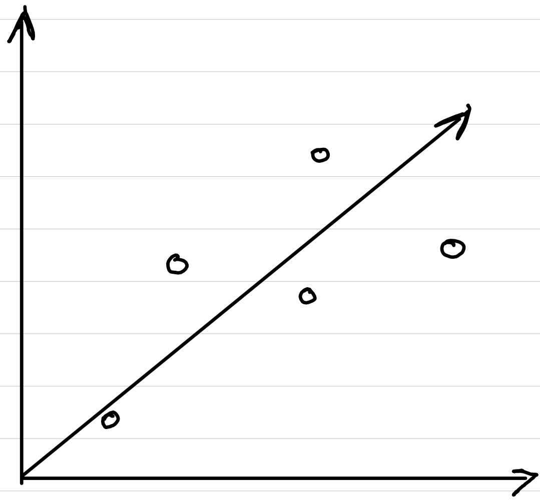
In ML, Linear regression is done through gradient descent

This is called as Least sum square method

Linear regression is a very basic ML model.

This idea can be extended into multiple dimentions as well

Simple Linear Regression



$$y = mx + c$$

$$\text{Error} = (y' - y)^2 \qquad MSE = \frac{1}{n} \sum (y' - y)^2$$

Gradient descent

$$\frac{\partial E}{\partial m} = \frac{2}{n} \sum_{1}^{n} (y'_i - y_i) \cdot x_i$$

$$\frac{\partial E}{\partial c} = \frac{2}{n} \sum_{1}^{n} (y'_i - y_i)$$

$$m = m - \frac{2}{n} \alpha \cdot \sum (y' - y) \cdot x$$

$$c = c - \frac{2}{n} \alpha \sum (y' - y)$$

Goodness of fit

$$R^2 = \frac{\sum (y' - \bar{y})^2}{\sum (y_i - \bar{y})^2}$$

→ Test to determine how good the algorithm is

Based on R^2, we can calculate how much x & y are corellated.

Larger values of R^2 means lesser corellation
Ideal value of $R = 1$

Note →

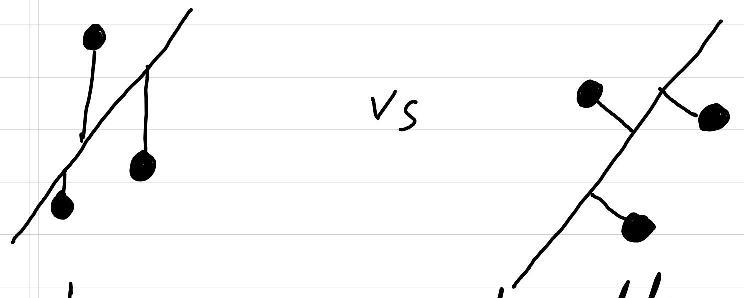
In linear regression we are taking only the vertical distance & not the distance from the line.

This is because of the following assumptions

1) Independant variable x is known & dependant variable y is known

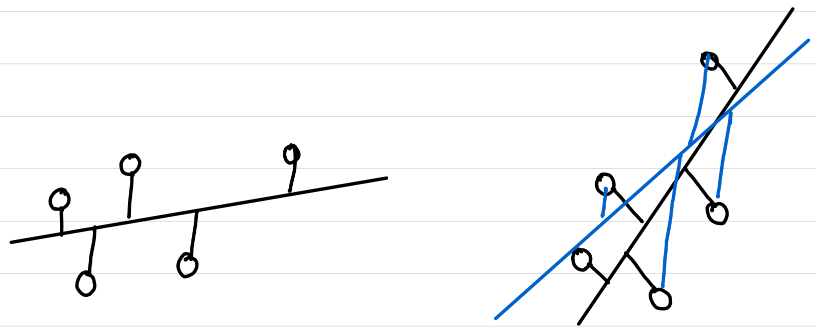
2) We want to minimize error in prediction & not find best fit line.

Hence geometrically speaking linear regression is not line fitting



Linear regression is not rotation invariant for the same reason.

If points are rotated by θ, then line doesnt get rotated by θ



① $x = [1, 2, 3, 4, 5]$
$y = [3, 4, 2, 4, 5]$

$L = 0.0001$

→ Epoch 1

$y' = [0 0 0 0 0]$, $m = 0$, $c = 0$ – Initialize

$$m = m - \alpha \times \frac{2}{n} \Sigma (y' - y) \cdot x$$

$$m = 0 + \frac{2}{5} \times 0.0001 \left[1 \times 3 + 2 \times 4 + 3 \times 2 + 4 \times 4 + 5 \times 5 \right]$$
$$= 0.00232$$

$$c = c - \alpha \frac{2}{n} \Sigma (y' - y)$$

$$= 0 + 0.0001 \times \frac{2}{5} \left[3 + 4 + 2 + 4 + 5 \right]$$

$$= 0.00072$$

Epoch 2

$$y' = mx + c$$

$$= [0.00304 \quad 0.00536 \quad 0.00768 \quad 0.01 \quad 0.01232]$$

$$m = m - \frac{2}{n} \alpha \left[\Sigma (y' - y) \cdot x \right]$$

$$(y' - y) = [2.996 \quad 3.994 \quad 1.992 \quad 3.99 \quad 4.98]$$

$$m = 0.00232 + \frac{2}{5} \times 0.0001 \times [\, 57.82 \,]$$

$$= 0.00463$$

$$c = 0.00072 + \frac{2}{5} \times 0.0001 \times [\, 17.952 \,]$$

$$= 0.00143808$$

Multivariate linear regression

$$x = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_n \end{bmatrix} \qquad x_0 = 1$$

$$\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{bmatrix}$$

$$y_i' = \theta^T X_n^i$$

$i \rightarrow$ data point
$n \rightarrow$ feature

$$M.S.E = \frac{1}{2n} \sum_{i}^{n} \left(y_i' - y_i \right)^2 = J(\theta)$$

$$\frac{\partial J(\theta)}{\partial \theta_i} = \frac{1}{n} \sum_{k=0}^{n} (y_i' - y_i) \, x_k$$

$$\theta = \theta - \alpha \frac{\partial J(\theta)}{\partial \theta_i}$$

Regularization

→ Techniques to caliberate ML algorithms to prevent overfitting & underfitting

Regularization

RIDGE — LASSO

L2 — L1

$$\text{Cost} = \text{Loss} + \lambda \sum \|w\|^2$$

Penalty

slope of curve

$$\text{Cost} = \text{Loss} + \lambda \|w\|$$

Ridge Regression

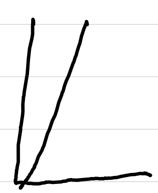
Regularization is used to add a bias in the data

By starting with a slightly worse fit ridge regression can provide better long term results

That is adding bias to reduce variance

Ridge Minimizes $\left(\text{sum of square} + \lambda\, (\text{slope})^2\right)$

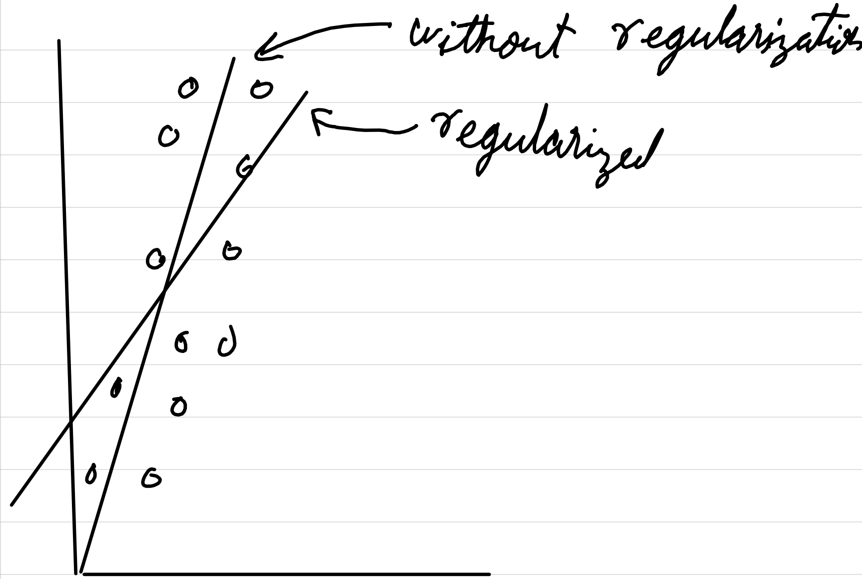
This means that the slope must be as small as possible



High slope

Low slope

When slope of the line is steep then the prediction for class is very sensative to small changes in the input



Q2 Find linear regression

L.R. = 0.001

x	1	3	5	6	7
y	2	3	4	5	6

→ $y = mx + c$

$$\frac{\partial J}{\partial m} = -\frac{2}{n}\sum (y_2 - y_1) \cdot x$$

$$\frac{\partial J}{\partial c} = -\frac{2}{n}\sum (y_2 - y_1).$$

Iteration 1 → $m = 0$, $c = 0$

y' $[0 \quad 0 \quad 0 \quad 0 \quad 0]$

$$\frac{\partial J}{\partial m} = +\frac{2}{5}(2\times1 + 3\times3 + 4\times5 + 5\times6 + 6\times7)$$

$$\frac{\partial J}{\partial m} = +41.2$$

$$m = 0 + 0.001 \times 41.2$$
$$= 0.0412$$

$$\frac{\partial J}{\partial c} = +\frac{2}{5}(2+3+4+5+6)$$

$$= 8$$

$$c = 0 + 0.001 \times 8$$
$$= 0.008$$

Iteration 2 → $m = 0.04$, $c = 0.008$

$y' - y = mx + c - y$

$y' - y = [-1.95, -2.86, -3.786, -4.714, -5.703]$

$$m = 0.0412 + 0.001 \times [97.665] \times \frac{2}{5}$$
$$= 0.0412 + 0.039$$
$$= 0.0802$$

$$c = 0.008 + 0.001 \times 19.013 \times \frac{2}{5}$$
$$= 0.008 \quad 0.0076$$
$$= 0.0156$$