ML cont

Agenda

1. Ridge
2. Lasso
3. elastic net
4. LoR

$$\mathcal{E} = \frac{1}{2m} \sum (y_i - \hat{y}_i)^2$$

$$\mathcal{E} =$$

Adjusted r^2

If we add more columns, even random unrelated col^n, ⇒ variance is gonna increase ⇒ r^2 ↑↑

temp america	study	sleep	score
53F		↓↓	
60F		r^2 ↑↑	
78F ⇒ r^2			

the score get increased or even can stay as it is which is not an accepted behaviour.

Adjusted r^2

$$r^2_{adj} = 1 - \left[\frac{(1-r^2)\ (n-1)}{(n-1-k)} \right]$$

n: no. of rows

k: no. of indep. features

1) irrel. feature like say temp. added

$R \uparrow\uparrow$

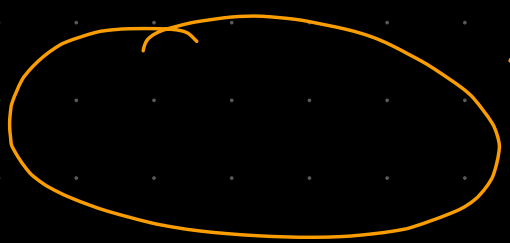
$\Rightarrow (n-1-R) \downarrow\downarrow$ $\quad (n-1)$ const

$R^2 \uparrow$

$\Rightarrow$ const/dec. $\quad \Rightarrow$ const.

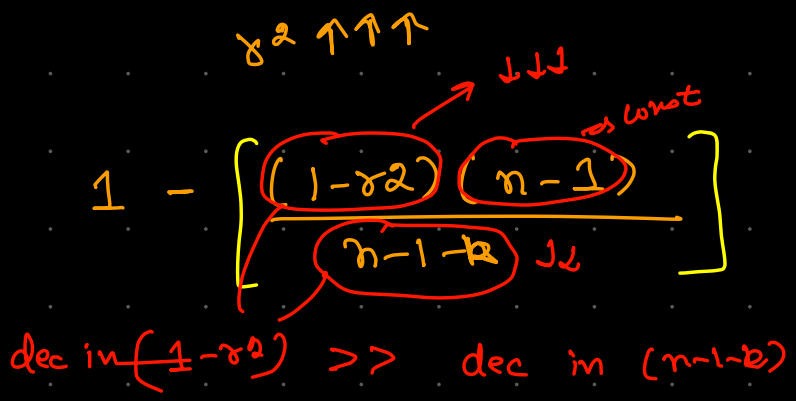
$$1 - \left(\frac{(1-R^2)(n-1)}{n-R-1} \right) \downarrow\downarrow\downarrow$$

$\frac{SS_R}{SS_E} \downarrow\downarrow \Rightarrow \frac{SS_R}{SS_E} \uparrow\uparrow$

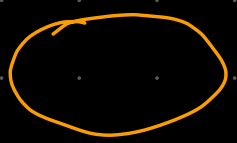
$\Rightarrow 1 -$  $\uparrow\uparrow$

$\Rightarrow$ adjusted R^2 gonna decrease.

2. Some relevant feature say sleep hours



$\frac{12}{4} = 3$ $\quad$ $\frac{8}{3} = 2.666$

$\Rightarrow$ $1 -$  ↓↓↓

$\Rightarrow$ γ^2_{adj} >>

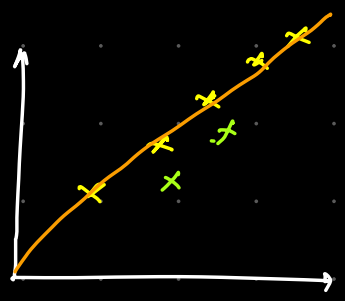
Assumption of LR

1. Linear relation b/w data point 
2. Indep. feature should have normal distribution.
3. Take care of multicollinearity
4. Homo scedasity
5. Heteroscedasity
6. Feature scaling

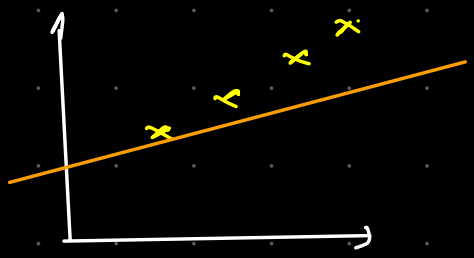
1. Overfitting in Algo :

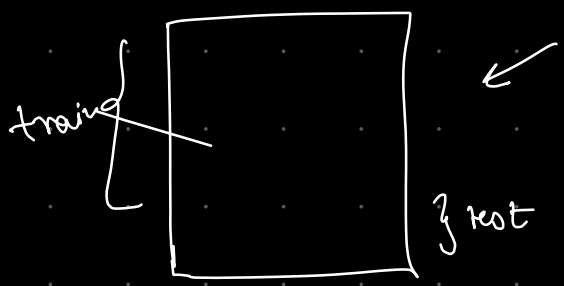
$5 + 4 = 9$ ✓

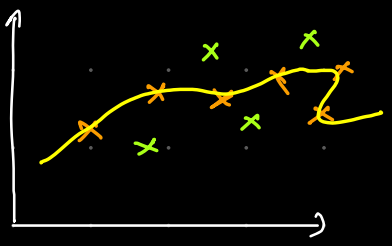
$5 + 2 = 7$ ✗

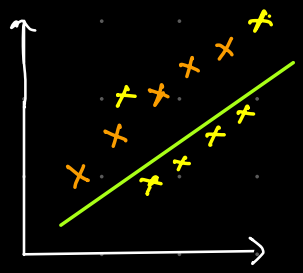


2. Underfitting



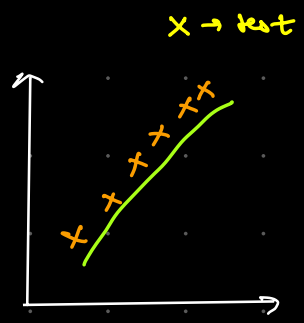




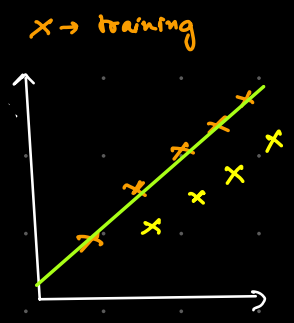


M1

Underfitting



M2



M3

Overfitting

⇒ Bias ⇒ Training data

⇒ Variance ⇒ Test data

Underfitting ⇒ training error ↑↑

⇒ high bias

M1

high error in training data

↓↓

bias

Now for test data, we can have 2 conclusion.

1. low variance
2. high variance

M2

low bias ⇒ training error

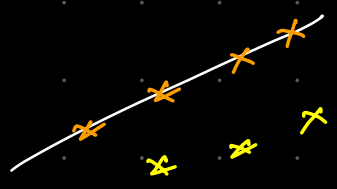
test data ⇒ low variance

no underfitting & no overfitting
⇒ accepted / required model

M3

low bias

error in test ↑↑
⇒ high variance

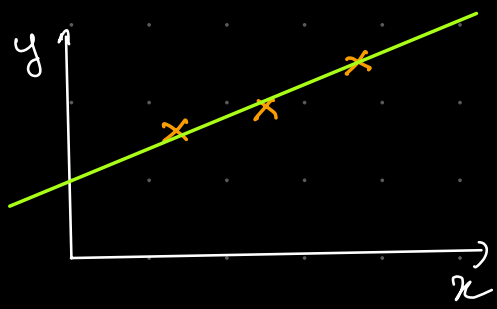


low bias + high variance ✓

So the aim should be to get a model like M2 which is generalized

⇒ low bias + low variance

Ridge regression



$$J/\mathcal{E} = \Sigma (y_i - \hat{y}_i)^2$$

$J \to 0$ in case of overfitting.

Low bias + high variance → Eg: M3

How to fix this?

$$\underset{\theta_0, \theta_1}{\mathcal{E}}(m, c) = \Sigma (y_i - y_i)^2 + \lambda \, (\text{slope})^2$$

We have added a penalty

$\lambda \to$ hyperparameter

slope → coefficient

What exactly will this λ do?

$\lambda = 1$

Our aim is to reduce cost f^n to global minima.

overfitting $\Rightarrow$ $J = 0$

$J \to 0$ $\Rightarrow$ line is fixed

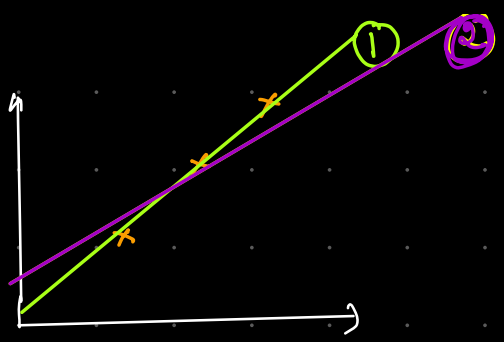
in ridg regression we are adding a compon.

$\lambda (\text{slope})^2$

When error is 0 & say $\lambda = 1$ $m = 15$

$$J = \boxed{0} + \underline{(2.25)}$$

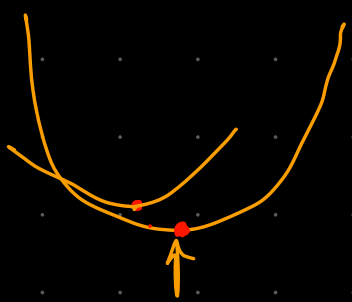
$\Rightarrow$ model will have to train more as we need to reach global min^m

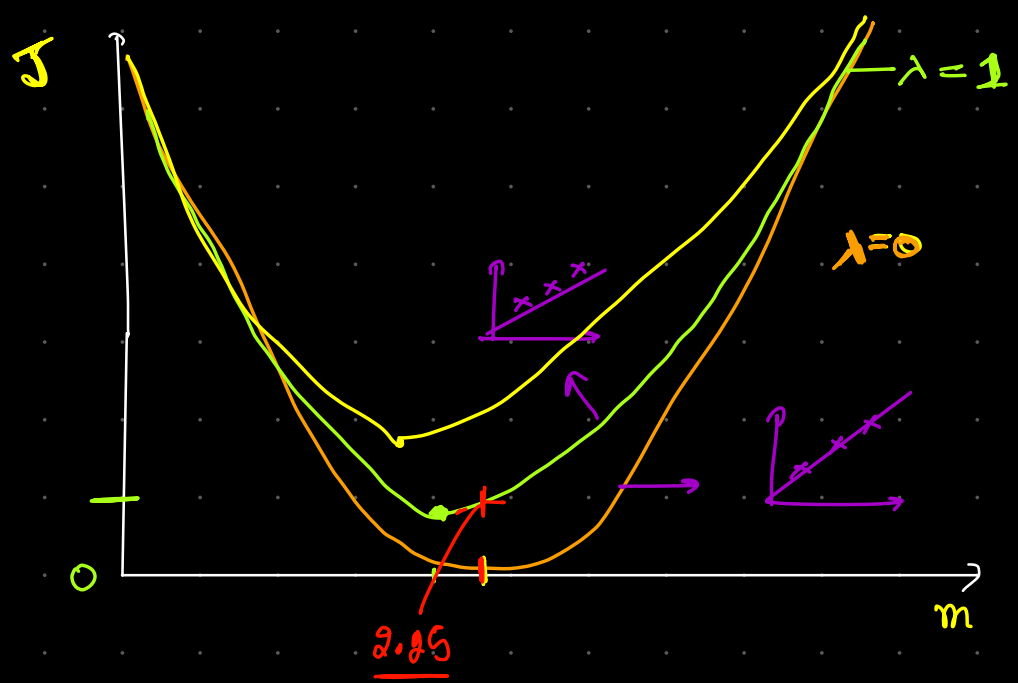


J = (small value) + $1\,(0.8)^2$

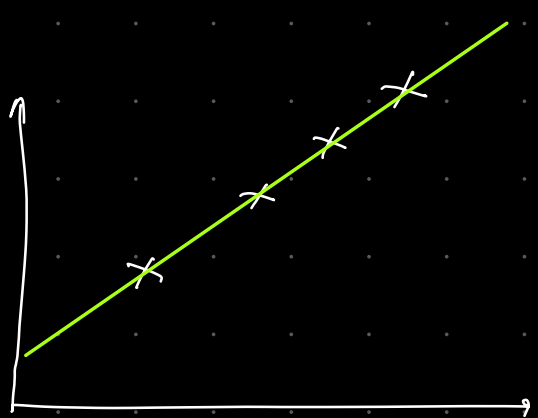
= (small value) + 0.64 < 2.25

↑ actual min





$$\frac{(y_i - \hat{y}_i)^2}{\text{Small value}} + \frac{\lambda}{1}(\text{slope})^2 \rightarrow (0.64 + \text{small}) << 2.25$$



relation b/w λ & slope ?

$\lambda \uparrow$ ⇒ slope $\downarrow\downarrow$

⇒ ridge regression help us in avoiding overfitting

λ ⇒ hyperparameter

Lasso Regression [L1 regularization]

Only cost fⁿ we will be changing

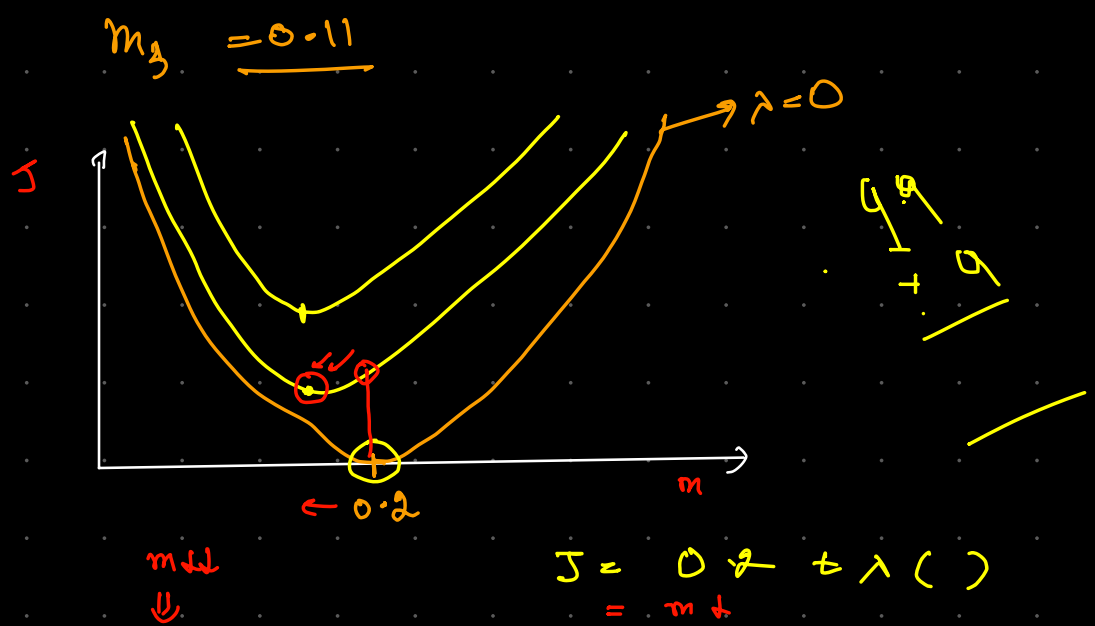
$$\mathcal{E} = \frac{1}{2m} \Sigma (y_i - \hat{y}_i)^2 + \lambda \Sigma (|\text{slope}|)$$

⇒ feature selection

$$y = m_1(x_1) + m_2 x_2 + m_3 x_3 + c$$

$$= m_1(\text{study}) + m_2(\text{sleep}) + \underbrace{m_3}_{\downarrow 0}(\text{temp}) + c$$

$m_3 = 0.11$ ⇒ it means for change of 1 temp just 0.11 unit change



Global minima gets shifted

Ridge ⇒ overfitting

Lasso ⇒ feature selection.

Elastic Net ($l_1 + l_2$)

lasso + ridge

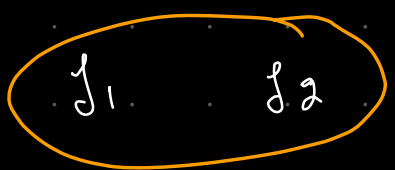
$$E = \sum (y_i - \hat{y}_i)^2 + \underbrace{\lambda_2 (\text{slope})^2}_{\text{ridge}} + \underbrace{\lambda_1 |\text{slope}|}_{\text{lasso}}$$

λ_1 & λ_2 are different

⇒ reducing overfitting & feature selection

When to use ridge and lasso?

Multicollinearity

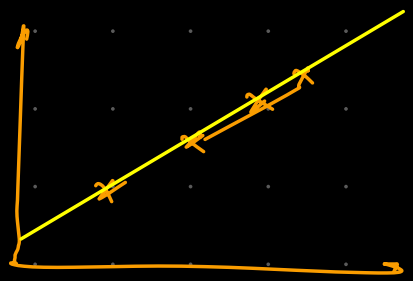


If f_1 & f_2 are highly correlated.

↙ feature selection (l_1)

↘ lot of similar feature ⇒ overfitting.

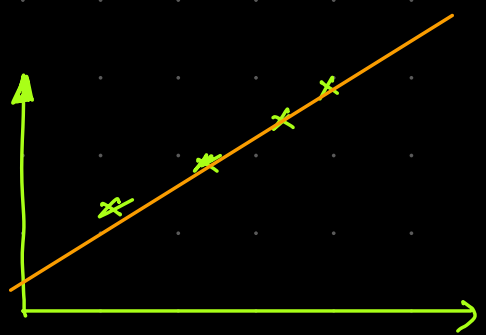
800 +

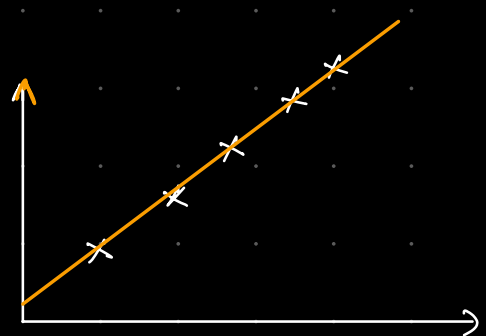


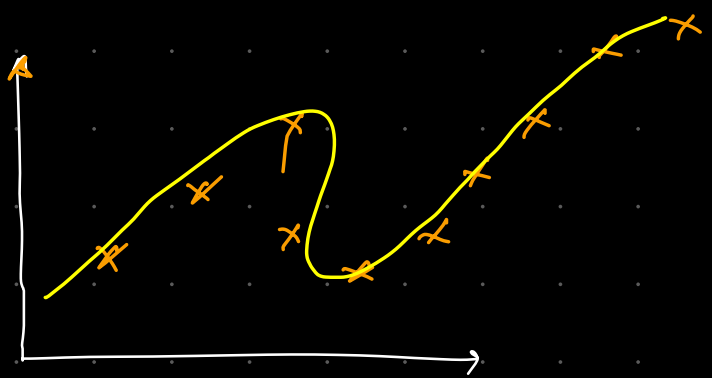
avoid overfitting
↓
l_2

feature selection
$\Rightarrow l_1$

$\Rightarrow$ elastic net



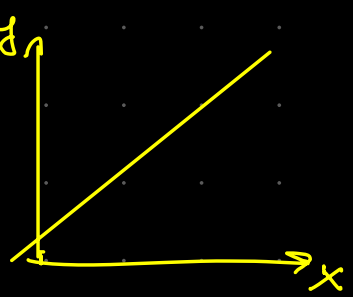




$x_1^2 \leftarrow x_1 \; x_2$ ⇒ pred

y ⇒ result

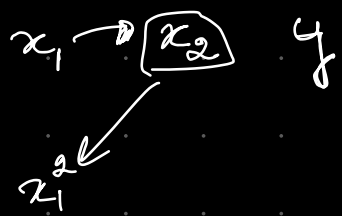
x_3



$$y = m_1 x_1 + m_2 x_2 + c$$

$$= m_1 x_1 + m_2 x_2 + m_3 x_3 + c$$

$$= m_1 x_1 + m_2 x_1^2 + c + m_3 x_3$$



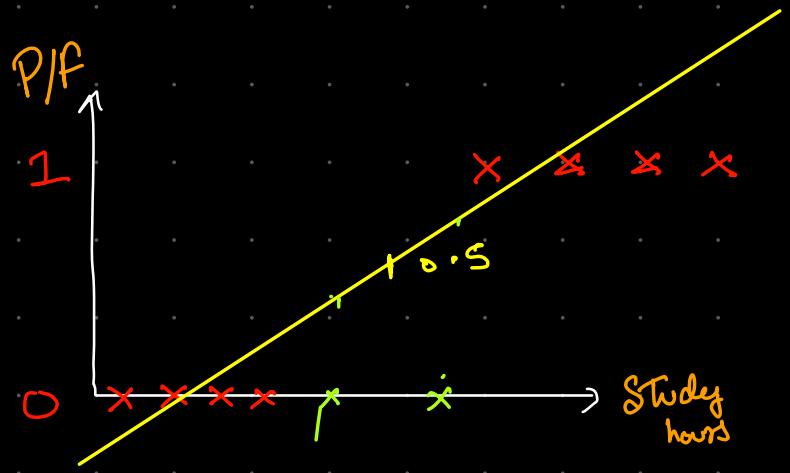
$$y = mx + c$$

$$y = m_1 x_1 + m_2 x_2 + c$$

$$y = m_1 x_1 + m_2 x_1^2 + c$$

Logistic Regression [Classification]

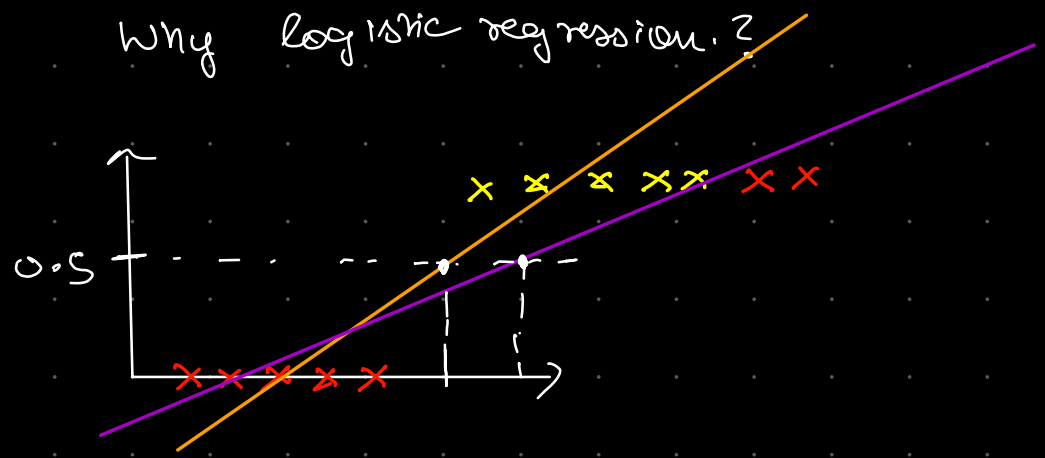
Binary Cl



$0.5 <$ ⇒ 0

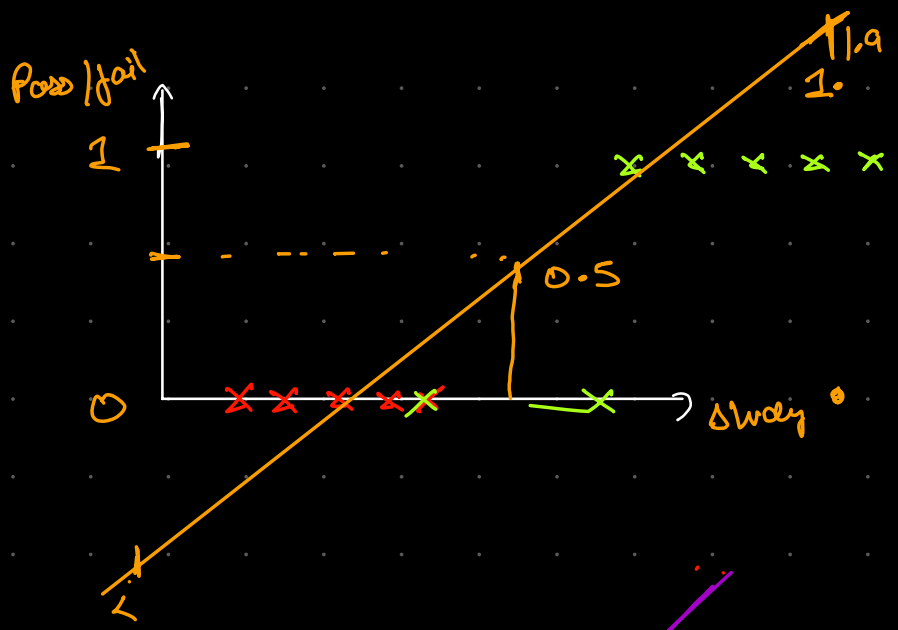
> 0.5 ⇒ 1

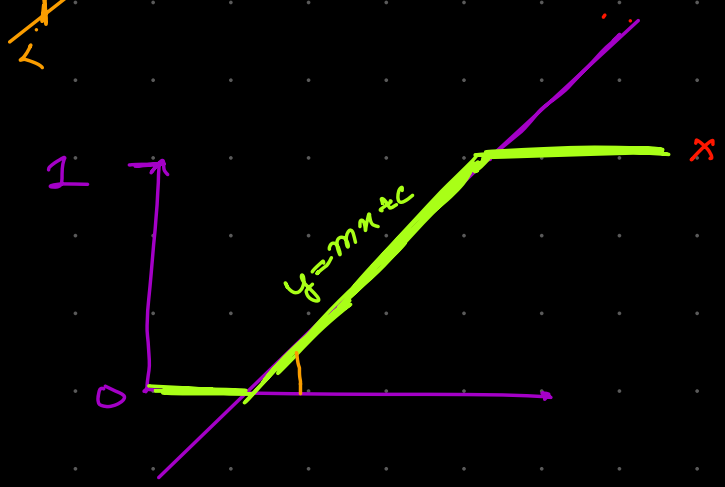
Why logistic regression.?



in training data

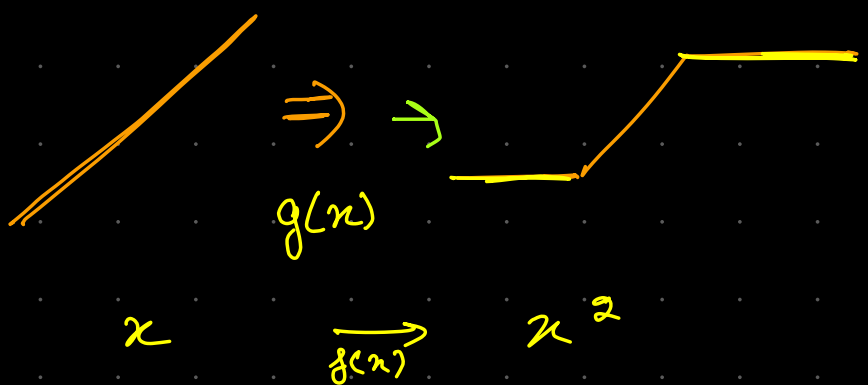
outlier, moved our line a lot





Our line equation is gonna be still

$$y = mx + c$$



$$f(x) = y = (\underbrace{mx + c}_{z})$$

$$f(x) = g(z)$$

$g(z)$ = sigmoid or activation f^n

$$g(z) = \frac{1}{1+e^{-z}} \qquad z = mx+c$$

activation f^n

linear regression

$y = mx + c$

$\varepsilon = (\quad)^2$

$$y = \frac{1}{1+e^{-(mx+c)}}$$

$\varepsilon =$

$y = mx + c$ ✗✗

$$y = \frac{1}{1 + e^{-(mx+c)}}$$

$$y(\underline{m}, \underline{c}) = \frac{1}{1 + e^{-(\underline{m}x + \underline{c})}}$$

$y = \{0, 1\}$