Polynomial Regression

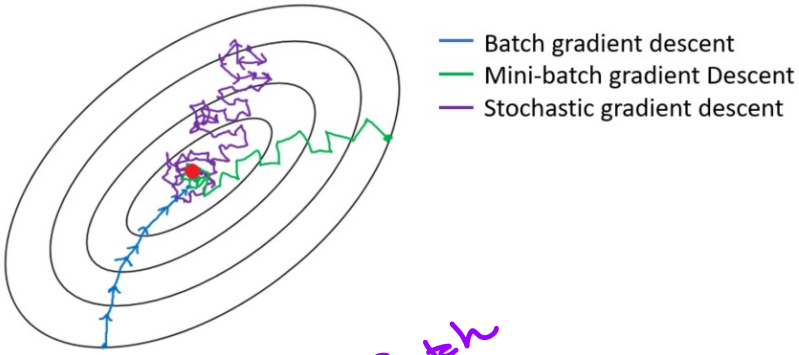
Agenda

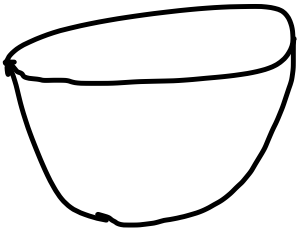
1. Polynomial Regression
2. Generalisation & Occam's Razor
3. Underfitting & Overfitting
4. Bias Variance Tradeoff

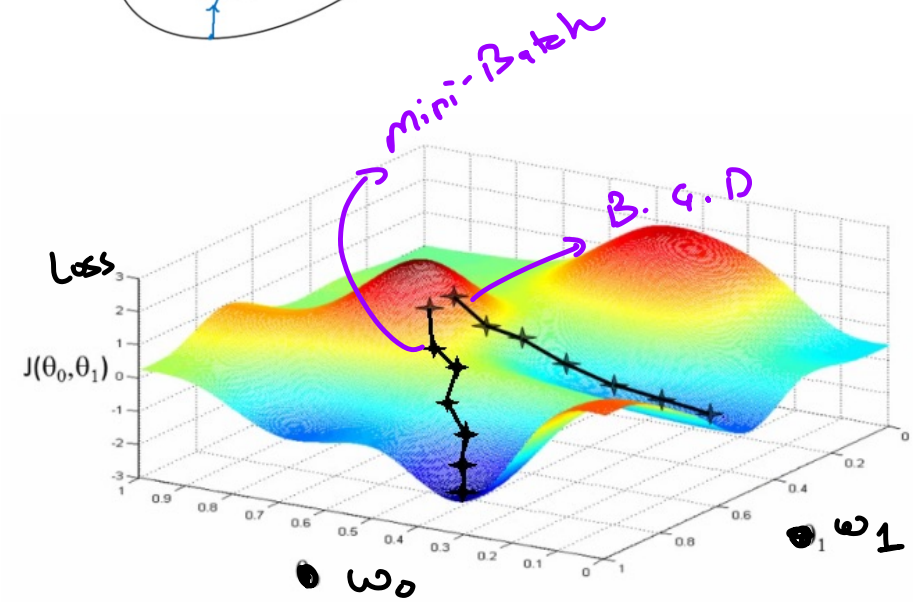
G.D Variants

$$w_j' = w_j - \eta \cdot \frac{\partial L}{\partial w_j}$$

Batch G.D	Mini-Batch G.D (Most Popular)	Stochastic G.D
① iterates all data Points – 'gradients | derivate'	① iterate over 'K' = 32 Batch size data Points.	① gradients calculated using 1 data Point.
② 100% Converge global minima	② in Between B.G.D & S.G.D	② Sometimes stucks in local minima.
③ Speed is slow	③ Speed is moderate	③ Speed is v. fast


Batch gradient descent
Mini-batch gradient Descent
Stochastic gradient descent



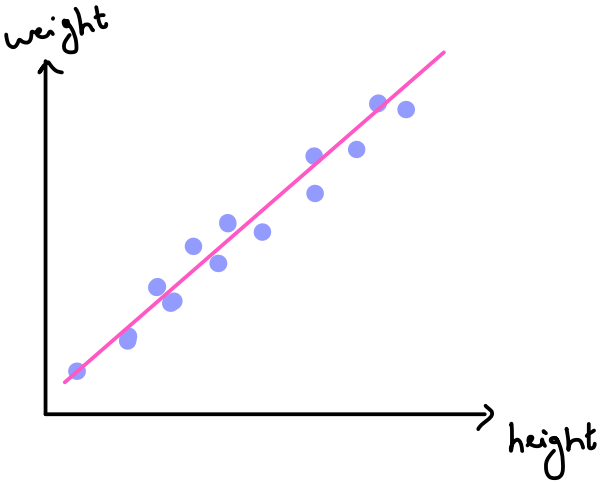

Mini-Batch
B.G.D
Loss
$J(\theta_0,\theta_1)$
ω_0
ω_1

Which statement is true about mini-batch gradient descent?

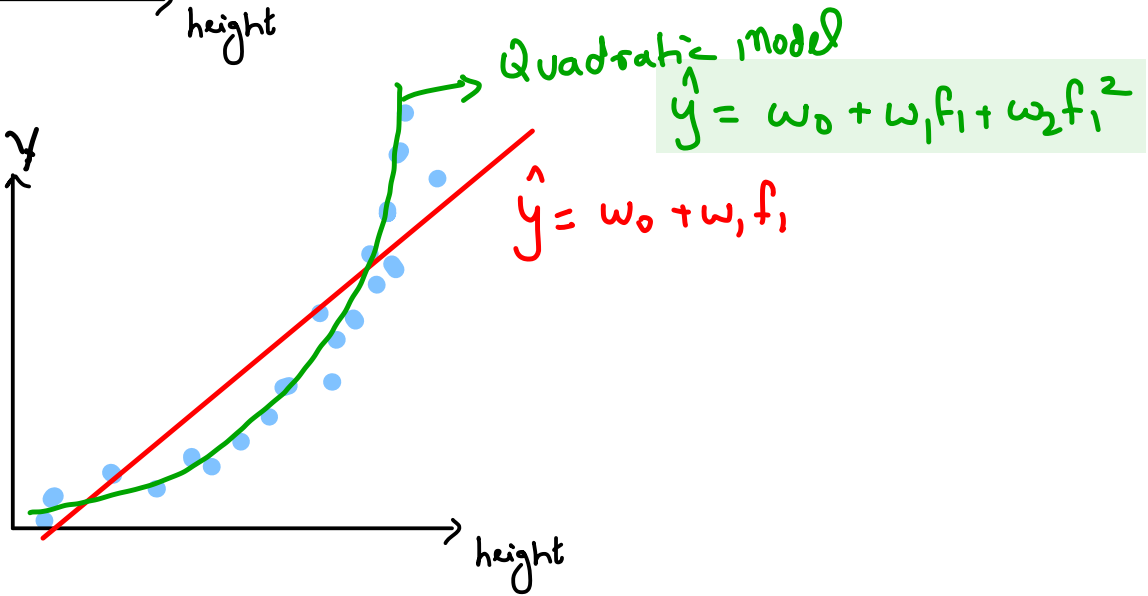
58 users have participated

✗	A	It guarantees convergence to the global minima	36%
✓	B	It may converge to a local minima due to the weight fluctuations	57%
✗	C	It requires a very high learning rate.	2%
✗	D	It is not suitable for large datasets (OPP.)	5%

End Quiz Now

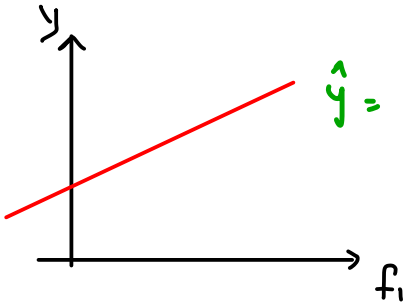

weight
height

f_1	f_1^2	y


y
height
Quadratic model
$\hat{y} = w_0 + w_1 f_1 + w_2 f_1^2$
$\hat{y} = w_0 + w_1 f_1$

f_1	y

$\xrightarrow{L \cdot R}$

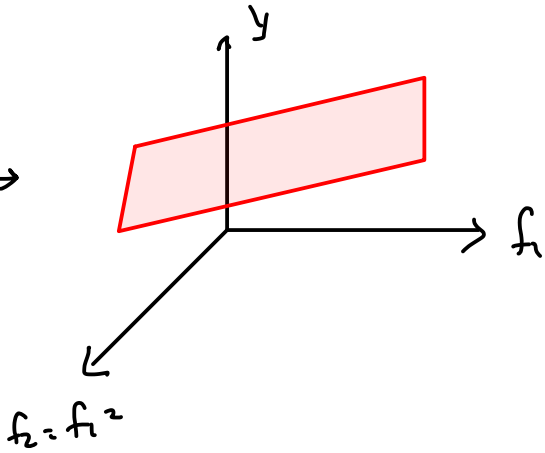


$\hat{y} = \omega_1 f_1 + \omega_0$

$\downarrow$ transform Data

f_1	f_1^2	y

$\xrightarrow{L \cdot R}$



$\hat{y} = \omega_1 f_1 + \omega_2 f_1^2$

Does $f_2 = f_1^2$ cause m.c ?

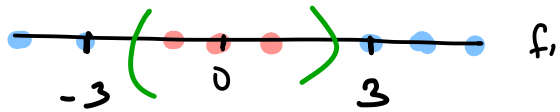
linear comb. of other features.

$f_2 = \alpha \cdot f_1 + \beta \rightarrow$ m.c

$f_2 = \underbrace{f_1 \times f_1}_{\text{Non-linear}} \rightarrow$ No. m.c

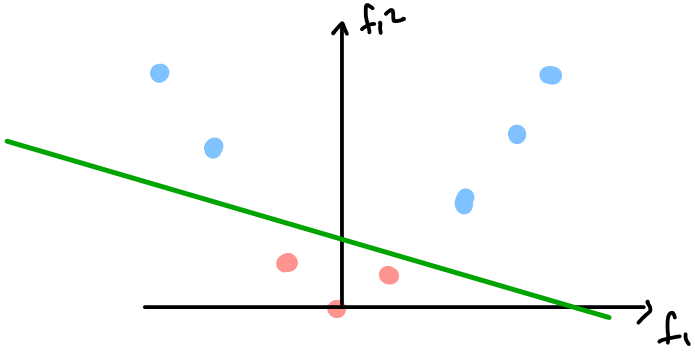
f_1	f_1^2	y

"Create Non-linear features"



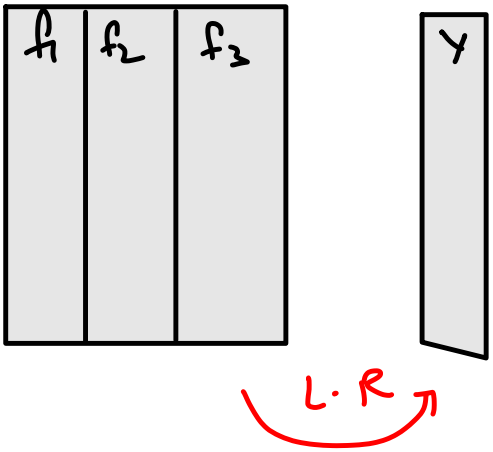
f_1	y

F.E



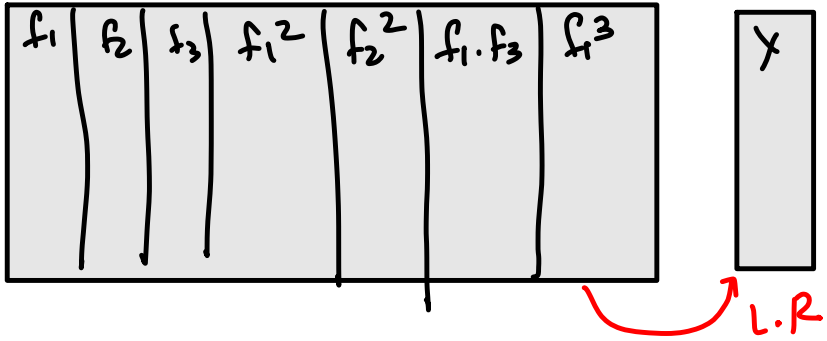
f_1	f_1^2	y
0	0	
1	1	
-1	1	
3	9	
-3	9	

linear data in higher Dim^n = Non-linear data in lower dim^n



$\Rightarrow \quad R_2 = 0.65$

F.E (Non-linear feature) / Poly^n features



$\Rightarrow \quad R_2 = 0.79$

Polynomial Regression

→ Linear Regression + (Non-linear features)

↓

Poly^n features.

$$\hat{y} = w_0 + w_1 f_1 + w_2 f_2 + w_3 f_1^2 + w_4 f_2^2 + w_5 f_1^3 + w_6 f_1^4$$

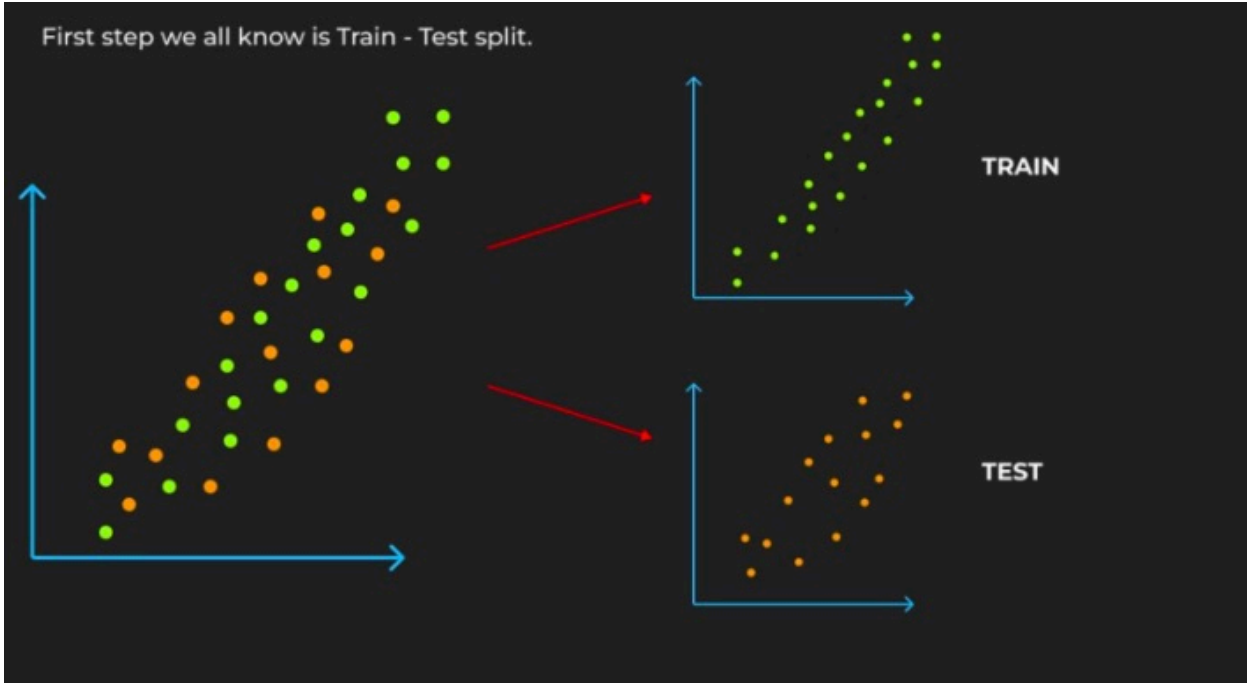
What metric should be used during Polynomial Regression ?

77 users have participated

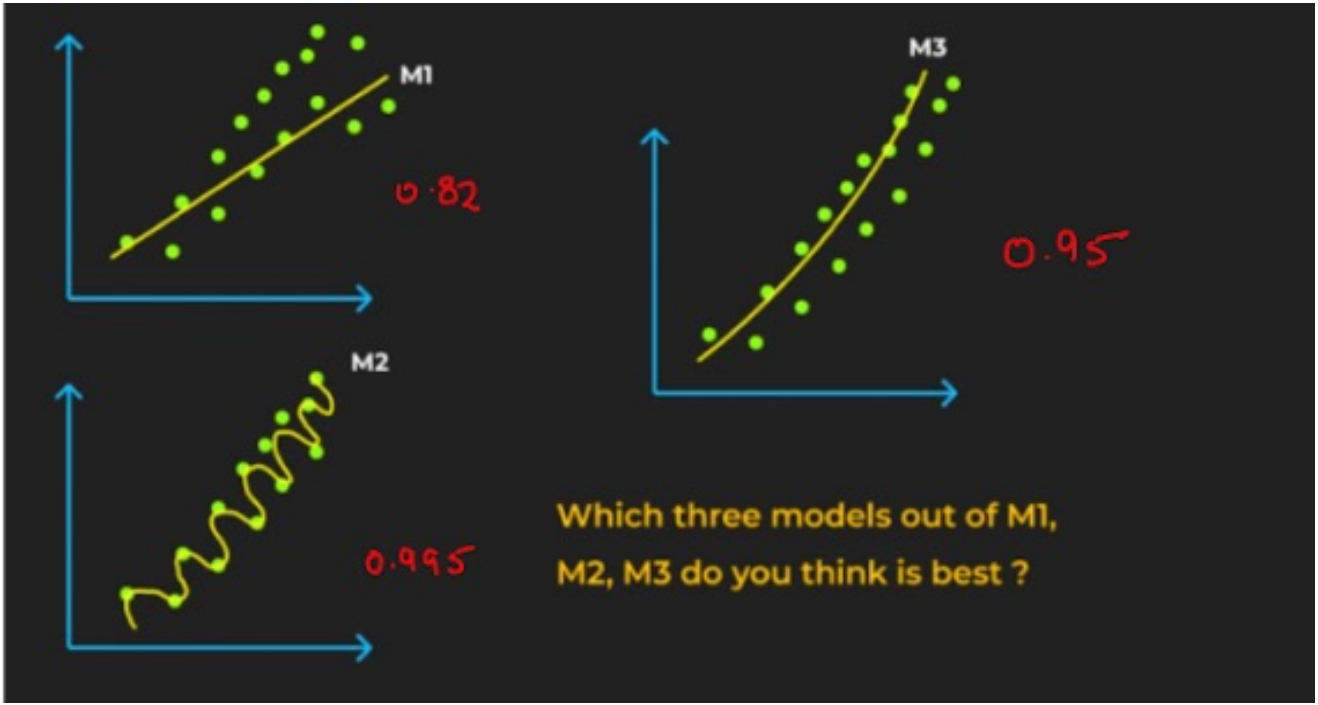
A	R-sq	13%
✓ B	Adj R-sq	84%
C	Doesnt matter	1%
D	Use a different metric	1%

End Quiz Now

Generalisation / Occam Razor



Train


M1
0.82
M3
0.95
M2
0.995
Which three models out of M1, M2, M3 do you think is best ?

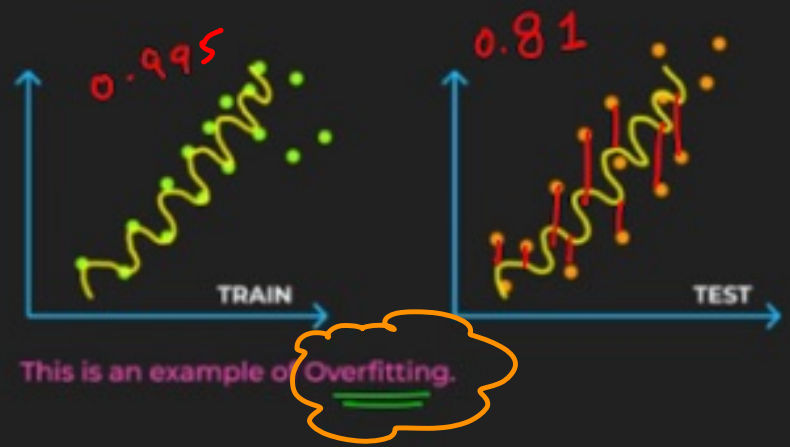
MODEL M1 Clearly M1 model is not fitting well on both training and test data.

Failed to capture patterns.

This situation is called as Underfitting.

- Imagine M1 as a student who did not study at all for exam, so bound to fail.

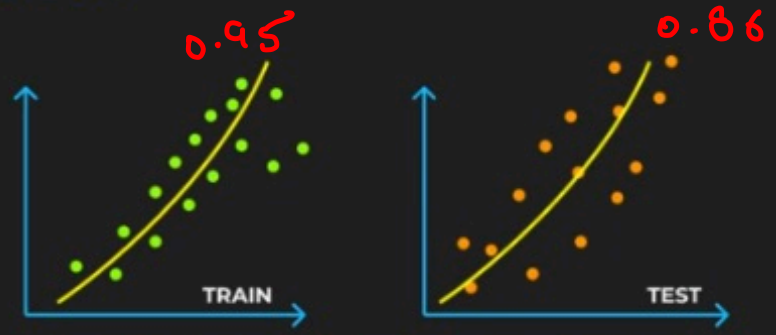
MODEL M2 M2 is performing great on training but only decent on testing.



Overcaptured the pattern due to noise.

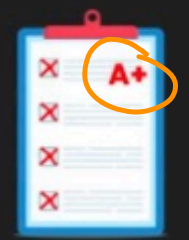
- Imagine M2 as a student who studied hard bt instead of understanding better, just cramped the Q/A

MODEL M3 M2 is performing great on training but only decent on testing.



Balanced between overfitting & underfitting & underfitting

Able to see the generic (Pattern)



- Imagine M3 as a student who studied smartly and understood the concept.

To summarize,

SIMPLEST

↕

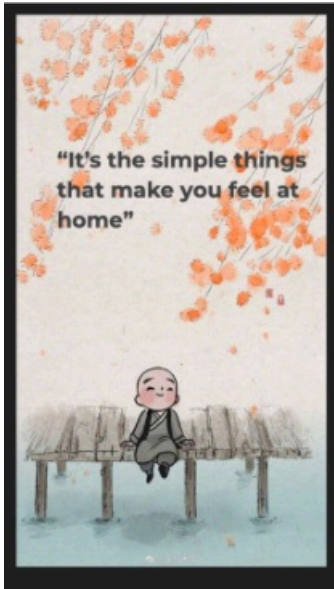
COMPLEX

	TRAINING	TESTING
UNDERFIT	POOR	POOR
PERFECTLY FIT	GREAT	GREAT
OVERFIT	BEST	DECENT

$$y = w_0 + w_1 f_1 + w_2 f_2 + w_3 f_1^2 + w_4 f_2^2 + w_5 f_1 f_2 + w_6 f_1^3 + \cdots$$

If there are several ML model, Generalization states that:

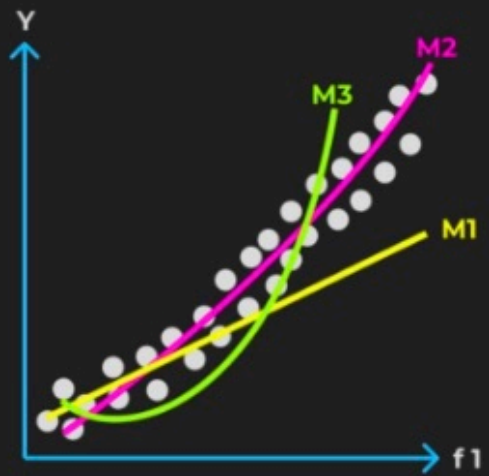
- Always choose the models that learns and understand the data inside out so that it can make good predictions on unseen data.



OCCAM's RAZOR

There is another rule that we follow while choosing best model.

**"There are many solutions to the problem,
always choose the simpler one"**



Now suppose given these 3 models :

M1 Linear: f_1

M2 Quad: f_1, f_1^2

M3 Cubic: f_1, f_1^2, f_1^3

Which one should we choose ?

Error

	M1	M2	M3
Test P[redacted]ce	HIGH	LOW	LOW

Params

w_0, w_1, w_2

Params

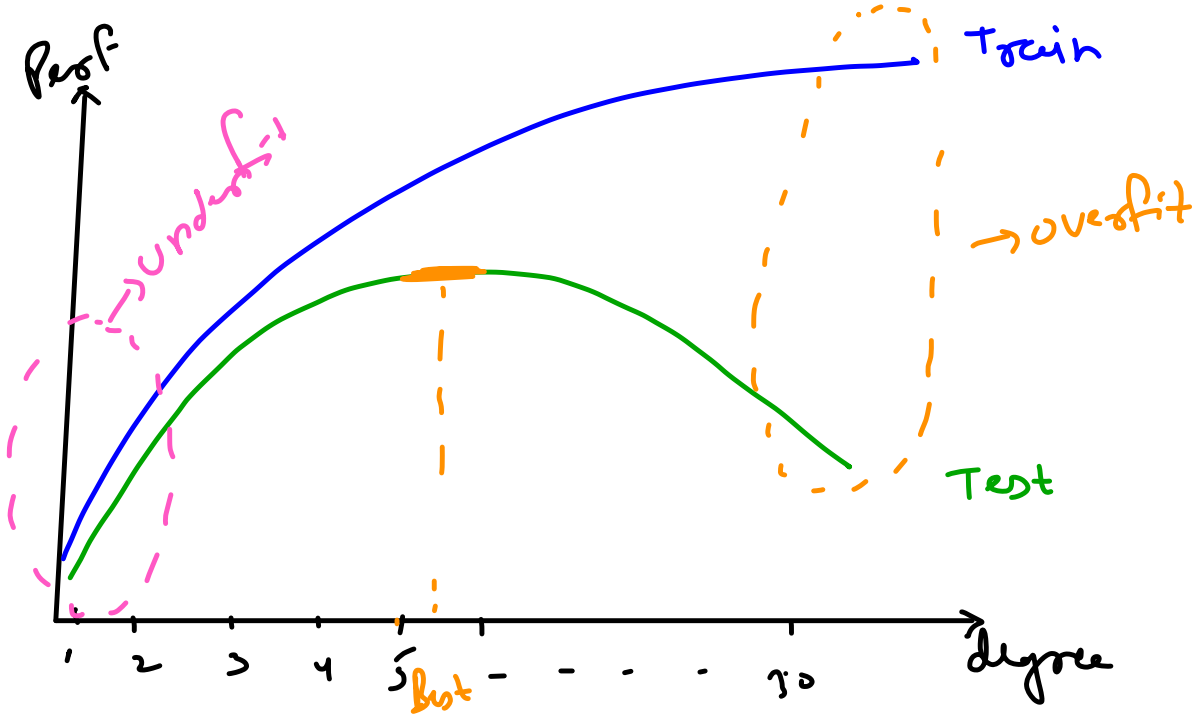
w_0, w_1, w_2, w_3

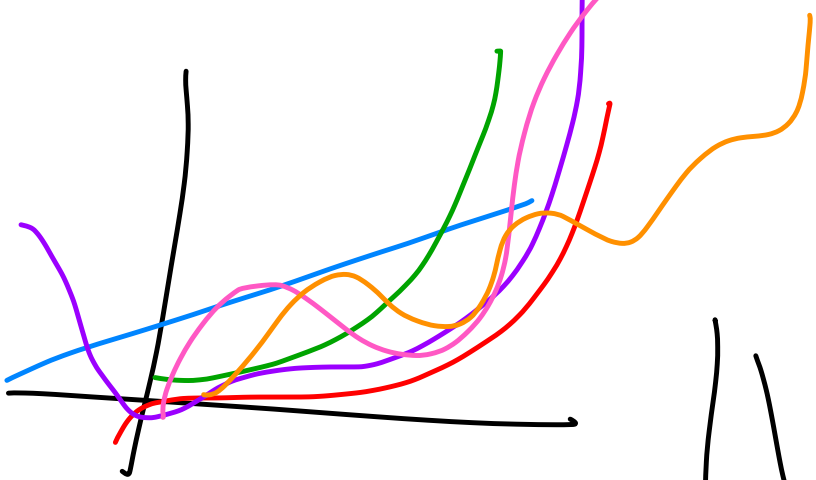
Why is Occam's Razor important in machine learning?

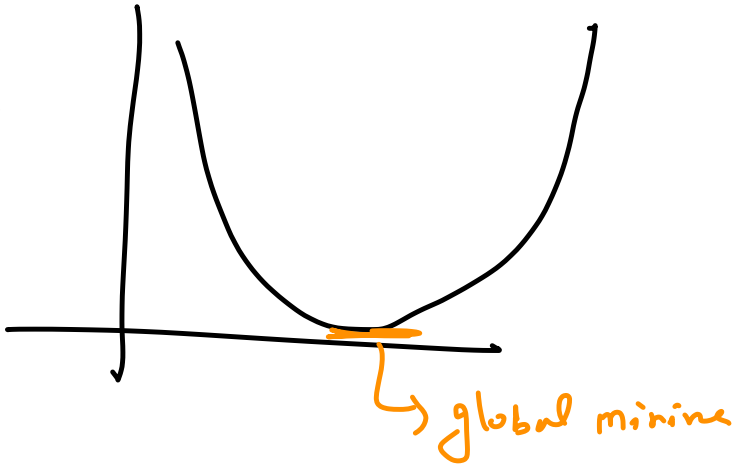
70 users have participated

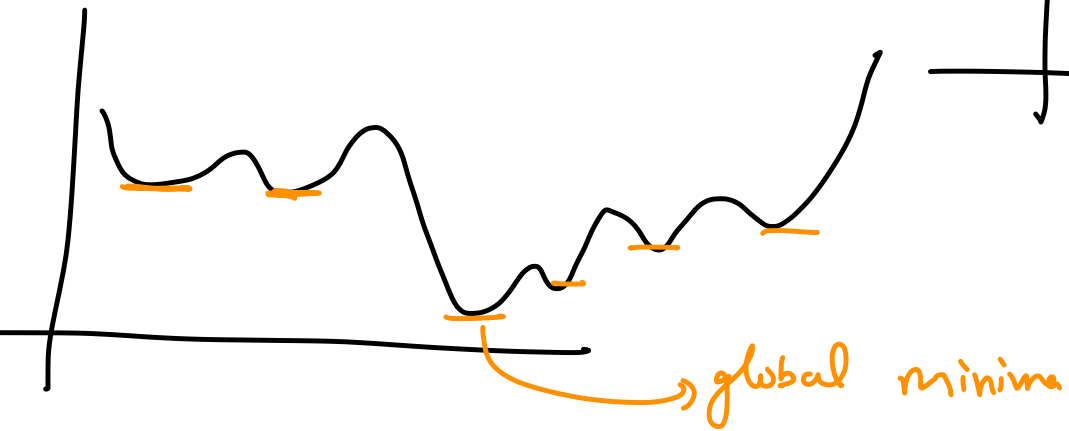
A	It helps in selecting the model that fits the training data perfectly.	21%
B	It encourages the use of complex models.	1%
C	It helps in avoiding overfitting by favoring simpler models.	76%
D	It promotes the use of large datasets for training models.	1%

End Quiz Now


Perf
Train
Test
underfit
overfit
1
2
3
4
5
Best
10
degree




global minima


global minima

Inference.

feature → ML model (Price) → Prediction

Pipeline.

1. Preprocess. → Encoding, Cat → Numeric
2. Scaling

0.9 ↕ 0.6

0.7

0.58

improve overfitting

GDt momentum