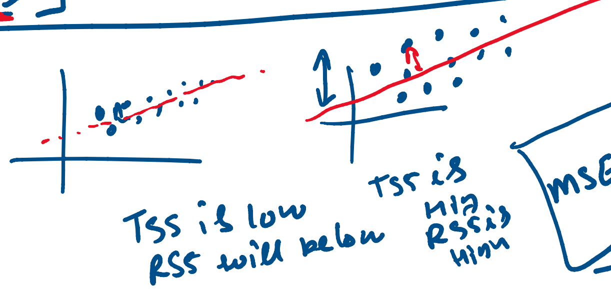
R2-Score

we use this metric to evaluate linear model performance. It is used as accuracy.

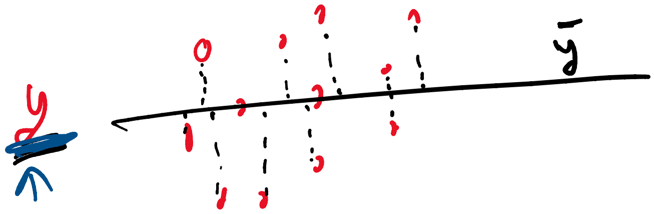
$\boxed{ACC = 1 - Error}$ Error $\in [0,1]$

$$\boxed{\text{Residual Sum of Square Error [RSS]} = \sum_{i=0}^{m} (\hat{y}_i - y_i)^2}$$

Prediction Error $(\hat{y}_i - y_i)$



$\boxed{MSE = \frac{RSS}{m}}$



→ Total Sum of Square Error

$$TSS = \sum_{i=1}^{m} (\bar{y} - y_i)^2$$

$\frac{10}{10} \to \frac{0-1}{1}$

Small

$\bar{y} \to 20$

$$\text{Total Error} = \frac{RSS}{TSS},$$

in general $\boxed{TSS > RSS}$

$TE \in [0,1]$

$\bar{y}$

$\boxed{TSS == RSS}$

25

1

model

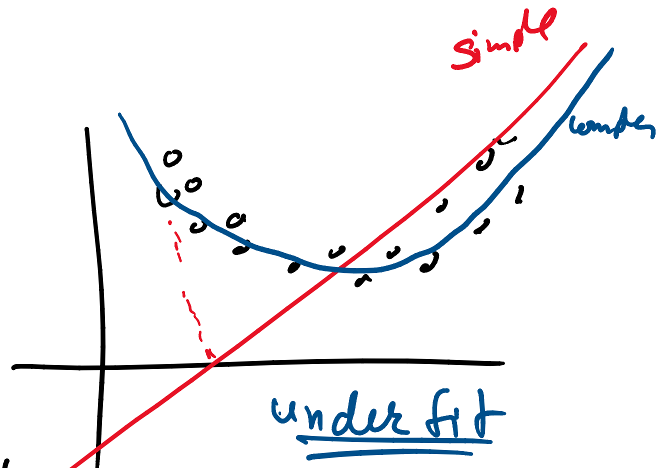
1

$X \to \hat{y} \simeq \bar{y}$

$$\boxed{R2\text{-}score = 1 - \frac{RSS}{TSS}}$$

$$\boxed{R2\_score = 1 - \frac{\sum_{i=1}^{m}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{m}(\bar{y} - y_i)^2}}$$

⇒ **underfitting**



Training Accuracy
and Test Accuracy
Both will be very low

Train Error very High
Test Error very High] → chances of underfitting

$X \to$ poly feature $\quad 2^0$

$$[X, X^2] \xrightarrow{fit} \boxed{\text{model}} \ \ LR$$

$$[X_1 \quad X_2 \quad X_3]$$

Transform 2^0

$$[x_1, x_2, x_3, x_1^2, x_2^2, x_3^2, x_1 \cdot x_2, x_2^2 x_1 \ldots]$$

no of features increase exponentially w.r.t degree of polynomial

$$\text{no of few} = \frac{(n+d)!}{n!\, d!}$$

$n=3$
$d=2$

$$\frac{5!}{3!\,2!} = \frac{5 \times 4 \times 3!}{3! \times 2!} = \boxed{10}$$

$3 \xrightarrow{2^0} 10$

col

no of N > no of m

row

Regularization

Simple → good fit ← Complex

underfit → good fit ← overfit

Test Error 2.

police

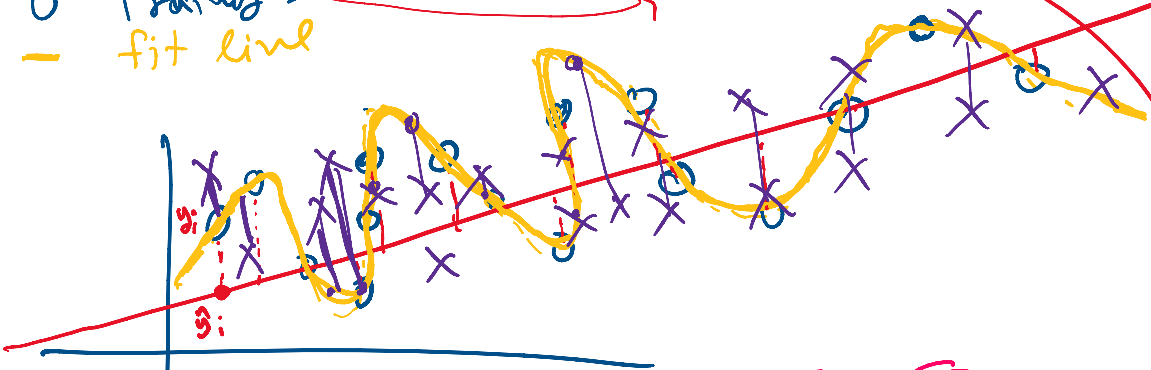
underfit

Train Error - High

Linear model

o Training Data

— fit line



d = 200

poly model

Train Error = 0

misure

overfitting

predict

Train Error ⇒ 0

100%

Training

SPAM

HAM

fit

model

SPAM → SPAM

HAM → Inbox

Test Error

new data

Test data

7 bus

100

Test

SPAM

HAM

X

y

$\hat{y}$ Result

MSE

Simple

Under fitting

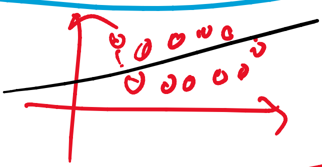
⇑

High Training Error

generalised Error

Test Error / Test Acc

$\theta_0 + \theta_1 X_1 + \theta_2 X_2$



$\hat{\theta}_0$ $\hat{\theta}_1$

Train Error

y

MSE

$\hat{y}$

SGD

θ params

Population

Data Set

Train

X_train

y_train

fit

predict

model

Test

X_test

X_test

y_test

X, y

Sampling

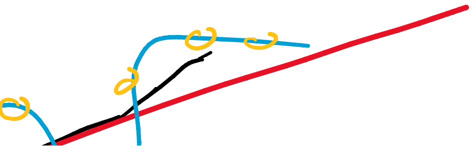
Random

Train Error ≃ Bias

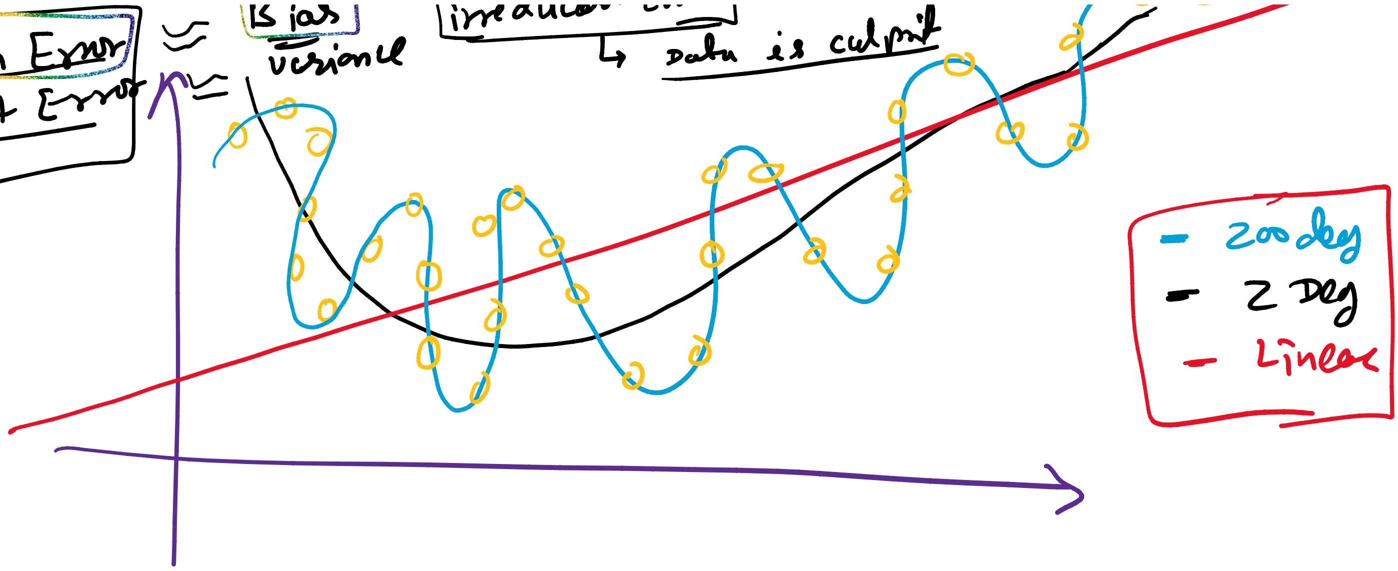
variance

irreducible Error

↳ Data is culprit



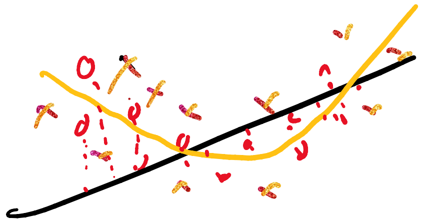
Train Error ≈ Bias
Test Error ≈ Variance
irreducible error
↳ Data is culprit



⇒ Bias - Variance Trade off

Bias → Error in our model due to model ~~simplicity~~
our model is two simple to understand data relationship

High Bias (Train Err) ⇒ Underfit



→ Search

Biased ⇒ Thinks

y Train	Test	$\hat{y}$
70 —	120	80 kg
85 —	125	120 kg
99 —	140	100 kg
100	145	150
123 —	160	120 kg
135	165	190 kg

Variance →

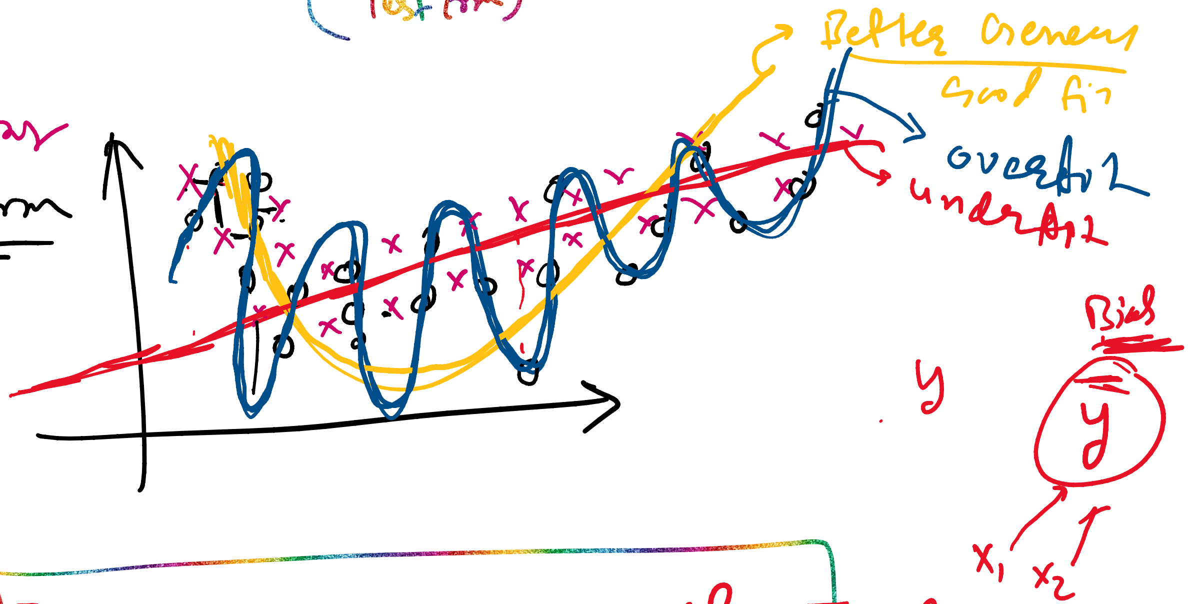
Error due to variability of data.

If your model is too sensitive to training data that model variance will be high.

High Variance ⇒ Overfitting
(Test Error)

Test Error

Train Error

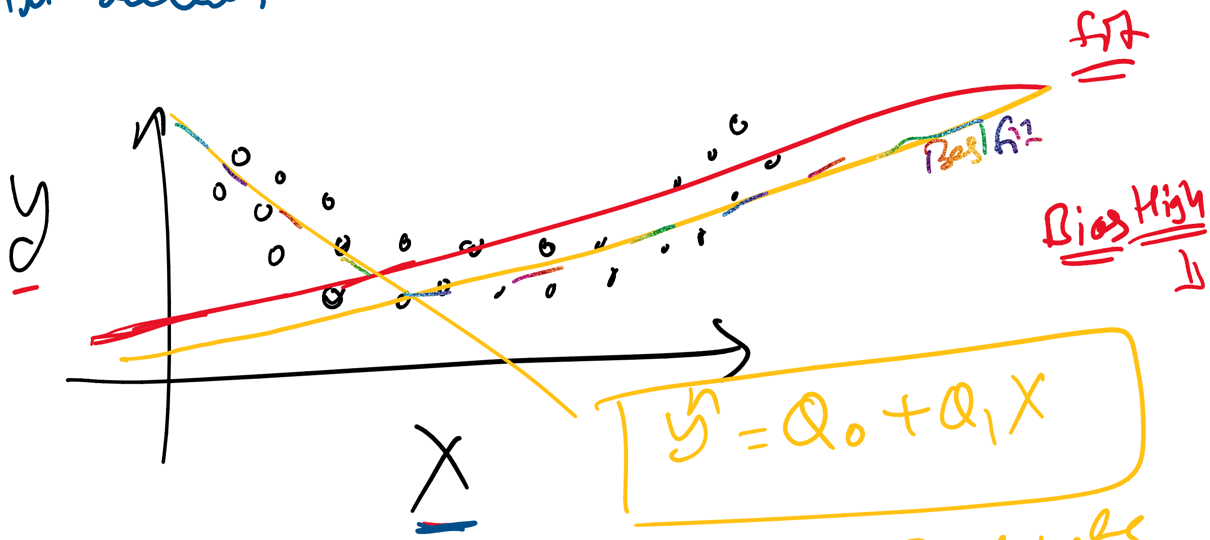


Bias vs Variance Tradeoff

simple complex

(Underfit) If we increase Bias of model then Variance will decrease

(Overfit) If we increase Variance of model then Bias will decrease



$$\boxed{\hat{y} = \theta_0 + \theta_1 X}$$

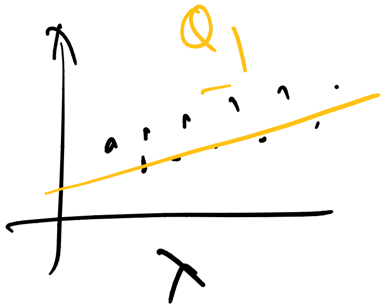
2 parameter

2

2 parame...

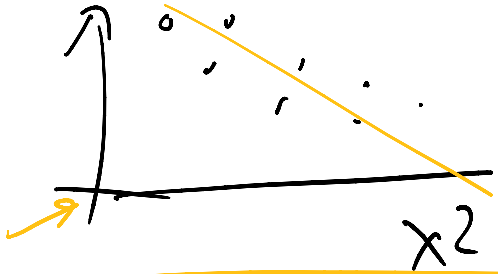
$\boxed{X \mid X^2}$

Q_1



Q_2

Q_0



white Box

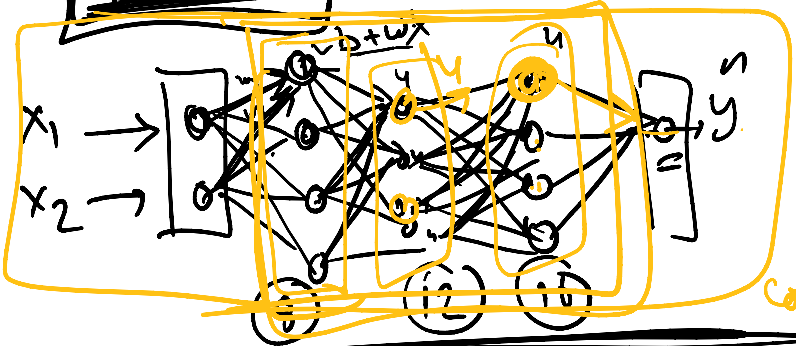
$$\boxed{\hat{y} = \theta_0 + \theta_1 X + \theta_2 X^2}$$

3 parameters

Deep learning

$\boxed{N.N}$

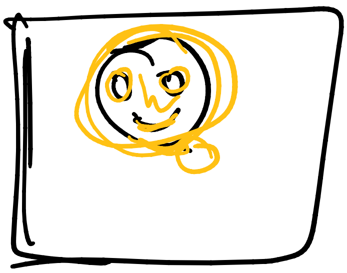
$\boxed{\text{Black Box}}$



Complex

36

CNN

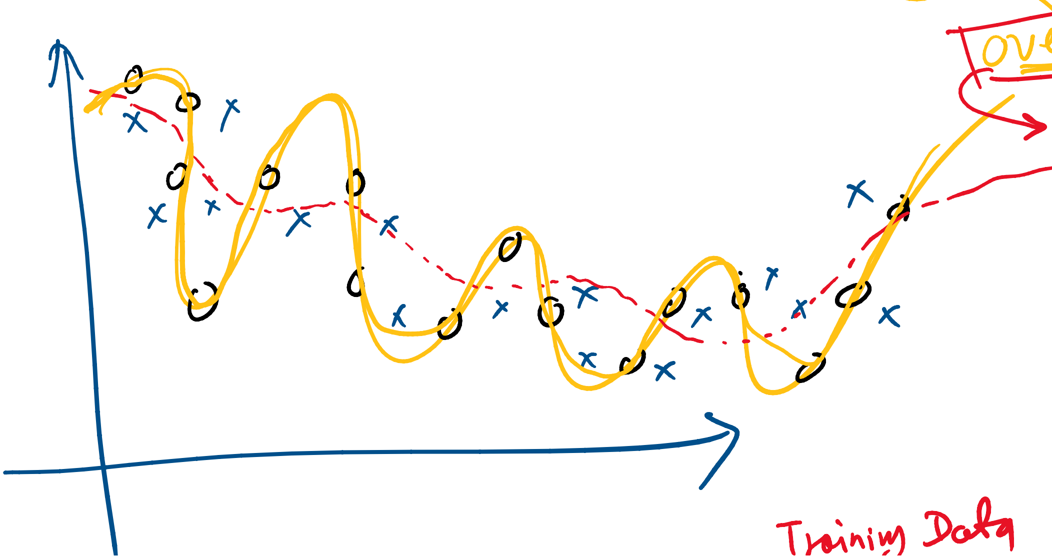


Regularization

Q

$\boxed{\text{overfit}}$

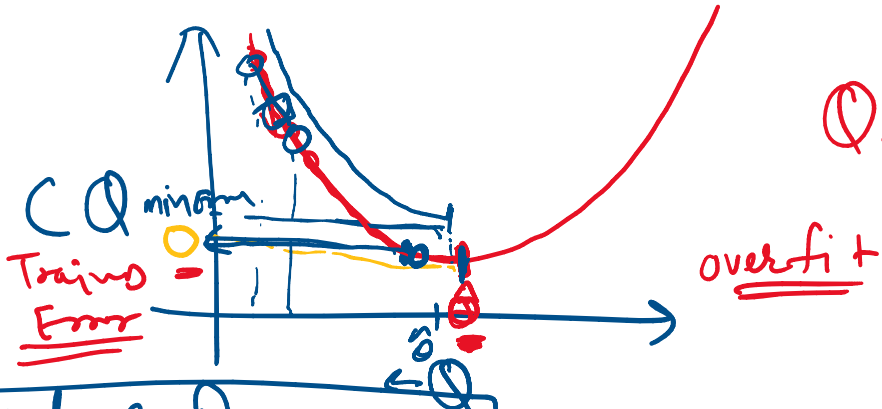
Best fit



Training Data

Training Data

$$\theta_n = \theta - \nabla_\theta \eta \, MSE\theta$$

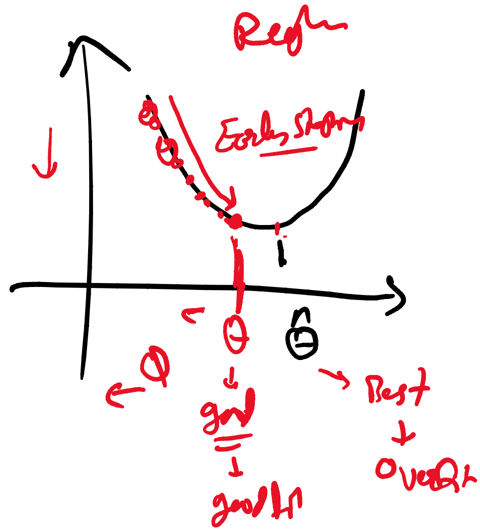


Ridge Regression

$$\Rightarrow \quad J\theta = \left[MSE\theta + \sum_{i=1}^{n} \theta_i^2 \right]$$

$$\theta_{next} = \theta_{pre} - \eta \left(\frac{\nabla_\theta MSE\theta}{5} + \frac{\nabla \theta^2}{2} \right)$$

1. Ridge
2. LASSO
3. Elastic Net
4. Early Stopping



Logistic Regression

Hyper

→ Housing Price → Deploy — 5-?

Logistic

MNIST → logistic

Data pipelines