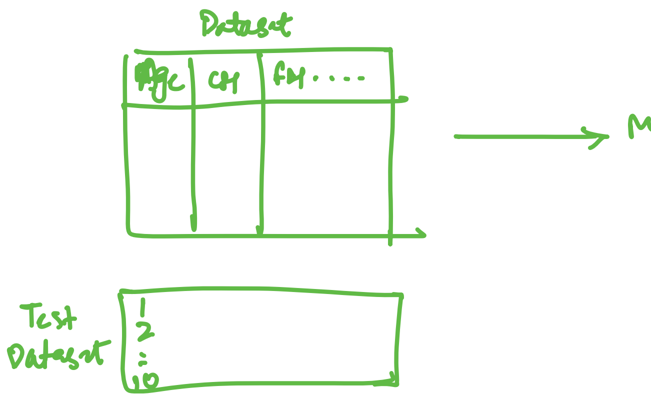


M1 →
M2
M3
:
M75

10

Actual	M1 Result
0	0
1	1
1	0
0	1
0	1
0	0
	:

False Negative (Type II error)

		Predict	
		detect Cancer (1)	Not Detected (0)
Actual	Has Cancer (1)	1000 TP	200 FN
	No Cancer (1)	800 FP	8000 TN

	detect Cancer (1)	Not Detected (0)
Has Cancer (1)	1000	500
No Cancer (1)	500	8000

Actual Cancer → Predict Not Cancer } Avoid, minimize

$$\text{Recall} = \frac{TP}{TP+FN} = \frac{\text{Correct}}{\text{Actual}=1}$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad \frac{\text{Correct}}{\text{Predicted}=1}$$

$$\text{Recall} = \frac{1000}{1200} > \text{Recall} = \frac{1000}{1500}$$

$$\text{Recall} = \frac{TP}{TP+FN} \qquad \text{Precision} = \frac{TP}{TP+FP}$$

want : high precision & high recall

Precision = 1
FP = 0
T I

Recall = 1
FN = 0
T II

	1	0
1	TP	FN
0	FP	TN

f1 Score:

→ Combination of Precision & Recall

Simple Idea:

$$\frac{P+R}{2}$$

$$\boxed{\frac{2PR}{P+R}}$$

Case 1: P = 0 R = 100

Simple Idea

$$\frac{P+R}{2}$$

$$\frac{0+100}{2} = 50$$

$$\frac{2PR}{P+R}$$

0

Case 2: P = 60 R = 100

$$\frac{60+100}{2} = 80$$

$$\frac{\cancel{2} \times 6\cancel{0} \times 100}{\cancel{160}_{\,8\cancel{0}}} = 75$$

Metrics:
- → Acc
- → Precision
- → Recall
- → f1

Predicted

Actual

	Dog	Cat	Horse	Total
Dog	25	5	10	40
Cat	0	30	4	34
Horse	4	10	20	34
Total	29	45	34	108

$P_D = \frac{25}{29} = 0.86$ $\quad P_C = \frac{30}{45} = 0.66$ $\quad P_H = \frac{20}{34} = 0.58$

- Macro - Precision $\rightarrow \frac{0.86 + 0.66 + 0.58}{3} = 0.70$

- Weighted Precision $\rightarrow \frac{40}{108} * 0.86 + \frac{34}{108} * 0.66 + \frac{34}{108} * 0.58 = 0.71$

$R_D = \frac{25}{40} = 0.62$ $\quad R_C = \frac{30}{34} = 0.88$ $\quad R_H = \frac{20}{34} = 0.58$

- macro Recall
- weighted Recall

Predicted

Actual	Dog	Cat	Horse	Total	Recall
Dog	25	5	10	40	0.62
Cat	0	30	4	34	0.88
Horse	4	10	20	34	0.58
Total	29	45	34	108	
Precision	0.86	0.66	0.58		

$F1_D = \frac{2 P_D R_D}{P_D + R_D} = \frac{2 \times 0.86 \times 0.62}{0.86 + 0.62}$

$F1_C = \frac{2 P_C R_C}{P_C + R_C}$

$F1_H = \frac{2 P_H R_H}{P_H + R_H}$

- Macro F1

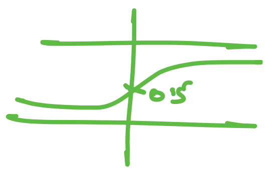
← weighted F1

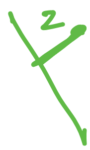
ROC Curve (Receiver Operating Characteristics)

→ Threshold find out
→ which model is better?

Classification: Log. Regression

iq	cgpa	place?
		Y, N?





hypothesis?

$\sigma(z) \rightarrow 0$ to 1 threshold set

$\sigma(z) \geq 0.5 \longrightarrow$ Yes
$\sigma(z) < 0.5 \longrightarrow$ No

Find out the best threshold?

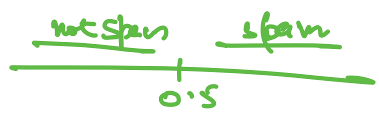
Threshold: 0.5   Spam Classification

2 mistakes

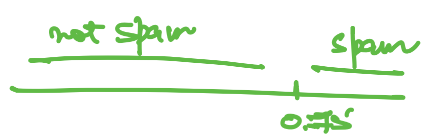
A: spam
P: not spam

A: not spam
P: spam

threshold 0.5



threshold 0.75



ROC Curve: 0.35   0.5   0.7   0.9 ?
find out
help

True Positive Rate (Benefit)

		Predicted 1	Predicted 0
Actual	1	TP	FN
	0	FP	TN

$$TPR = \frac{TP}{TP + FN} \rightarrow \text{(Recall)}$$

The task for which model was designed how good you are able to do?

Email Classifier (Spam, Not spam) : Among all the mails which are actually spam, how many your model was able to detect?

Churn Rate predict: Among all the people who were going to leave the platform, how many you were able to predict correctly.

		Pred. 1	Pred. 0
Act.	1	80	20
	0		

$$TPR = \frac{80}{100} = 80\%$$

False Positive Rate (Cost)

		Predict 1	Predict 0
Act.	1	TP	FN
	0	FP	TN

$$FPR = \frac{FP}{FP + TN}$$

Email: mails were actually valid but you said it is spam.

Churn Rate: people were not leaving your platform, but you gave them discounts. Company: cost.

$$TPR = \frac{TP}{TP + FN}$$

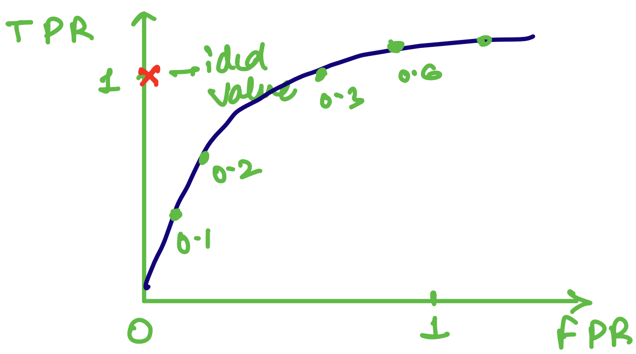
max benefit $= 1$ $\quad FN = 0$

$$FPR = \frac{FP}{FP + TN}$$

min cost $= 0$ $\quad FP = 0$

	0
0	

ROC Curve:



Threshold:

0.3 , 0.5 , 0.6 , 0.7 , 0.8

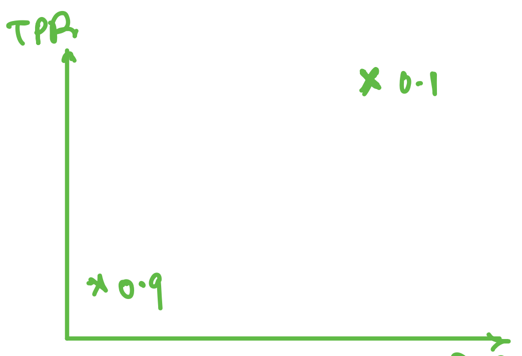
	Predict
<0.3 : 0	N
≥0.3 : 1	Y

		P. 1	0
A.	1	TP	FN
	0	FP	TN

TPR , FPR

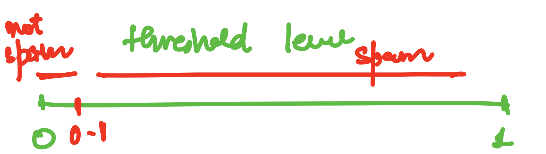
Optimal threshold: Closest to your ideal x

Impact of threshold on TPR & FPR:



		Predicted 1	0
Actual	1	TP ↑	FN ↓
	0	FP ↑	TN ↓

small: 0.1 ≈ 0

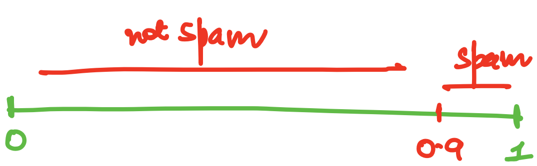


$$TPR = \frac{TP}{TP+FN}$$

= increase

$$FPR = \frac{FP}{FP+TN}$$

= Increase



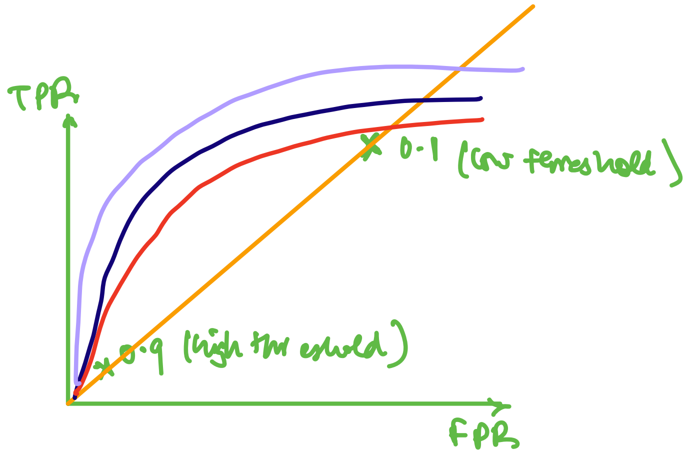
		Predicted 1	0
Actual	1	TP ↓	FN ↑

0 | FP ↓ | TN ↑

large: 0.9 ≈ 1

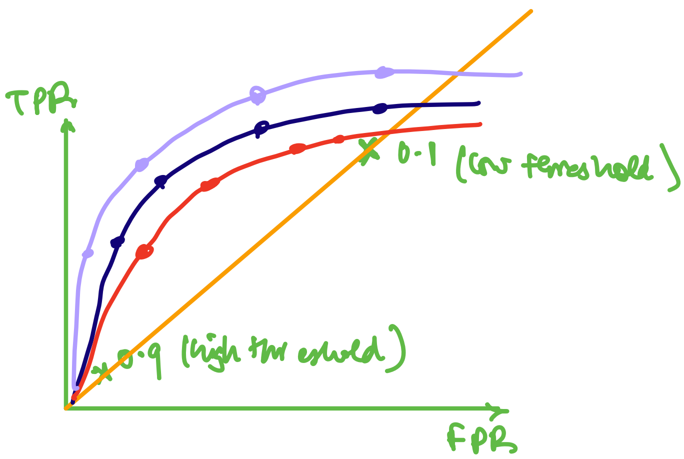
$$TPR = \frac{TP}{TP+FN} \qquad FPR = \frac{FP}{FP+TN}$$

= decrease   = decrease



AUC → Area Under Curve

ROC curve



NB: 0.2 0.6 0.8 0.9

Log. Reg: 0.2 0.6 0.8 0.9

SVM : 0.2 0.6 0.8 0.9

Decide which model is better?

Ideal value of AUC = 1 → close to 1.

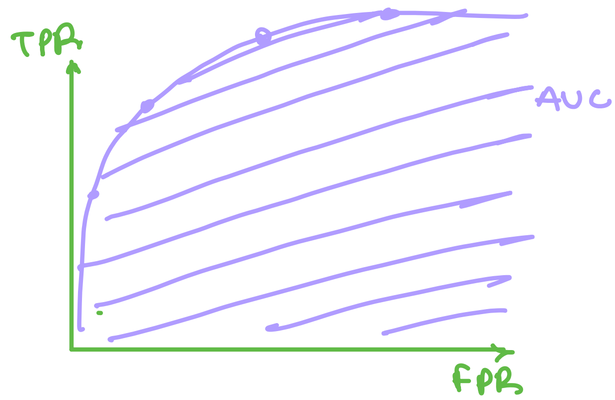
AUC: 1 : model is perfect

AUC: 0.5 : Random guesses

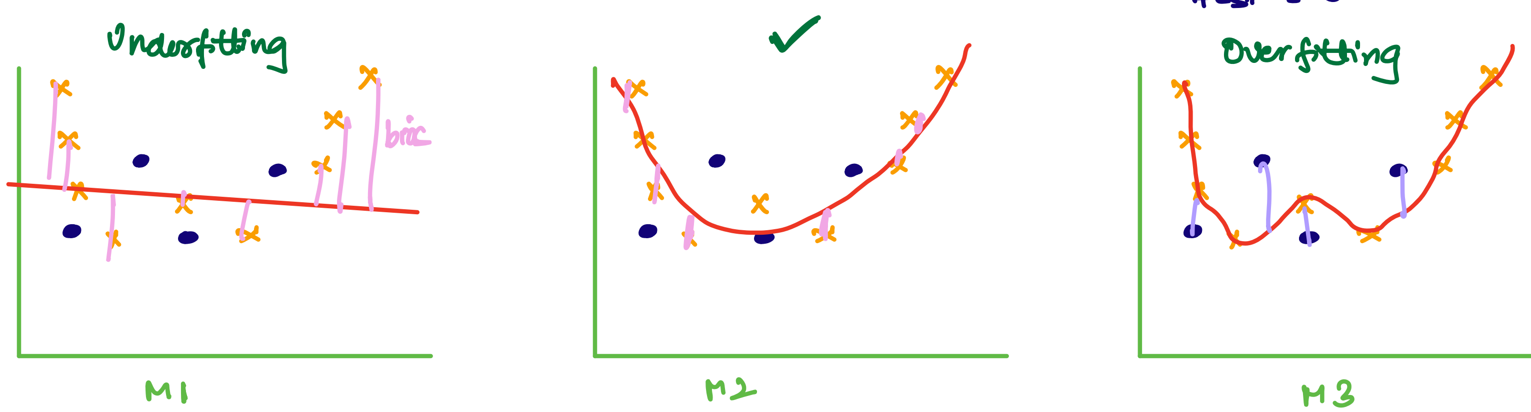
AUC: 0 : completely wrong

— : 0.85
— : 0.70
— : 0.65

0.85 is best.



Bias Variance Tradeoff:



High Bias
low variance

Low bias
High variance

Bias: Inability of your ML model to capture the relationship in training data.

Variance: Training set : low error } high variance
Test set : high error

0.01 } high variance
1000

1000 } low variance
999

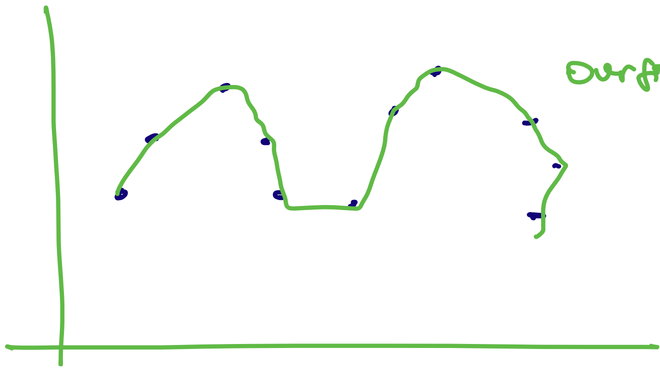
Ideal setup: low Bias } M2 is best
low Variance

Techniques to avoid overfitting:

- Regularisation
- Bagging
- Boosting

Overfitting:

Model performs very well on training data but doesnot perform good on test data.



Overfitted model

Solution:

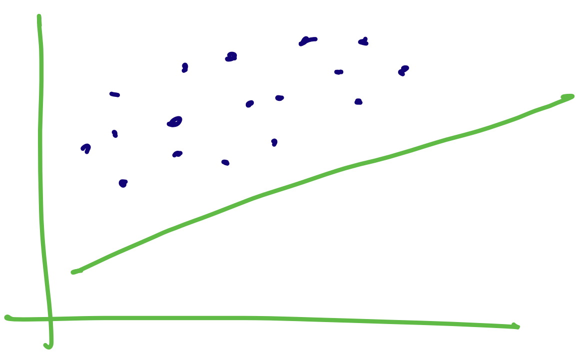
- → K fold cross validation
- → Sufficient Data
- → Ensembling techniques

  ↳ KNN → 0
  SVM → 1
  Log. R → 1
  DT → 1

  } Majority

Underfitting:

Model is not learning pattern from training set.



Underfit model will give bad performance on both training & test dataset.

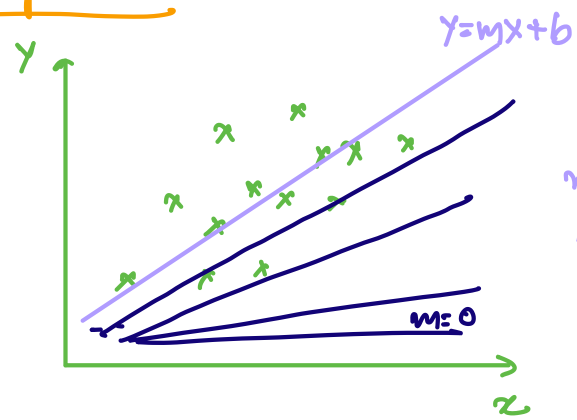
Solutions:

- → No. of features increase
- → Models Complex
- → Reduce noise
- → Inc. the duration of training

Regularization:

Ridge (L2)	Lasso (L1)	Elastic Net ($L_1 + L_2$)

Linear Regression



m = wt assign x

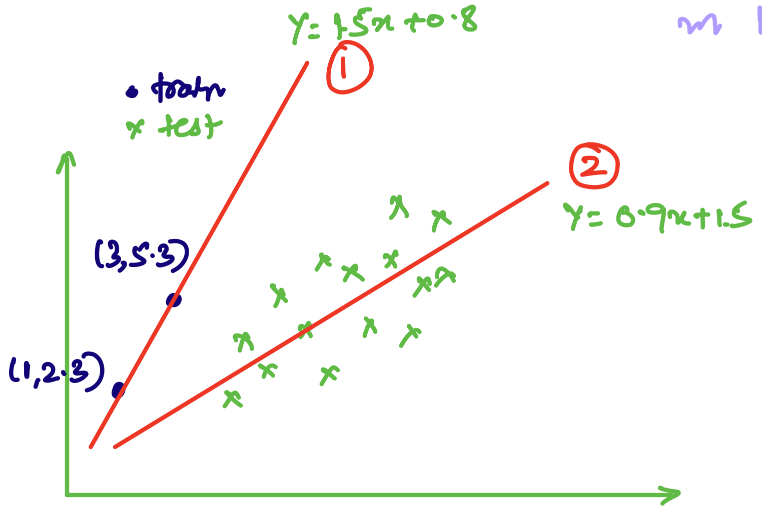
b = bias

m reduce → less importance to the feature x

m high → more importance to feature

No of hours x	Marks y

$y \in \mathbb{R}$



$$L = \frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)} - \hat{y}^{(i)}\right)^2 \quad \text{MSE}$$

$$+ \lambda (m^2)$$

1

$\lambda = 1$

Training loss $= 0 + (1.5)^2$

Training loss $= 2.25$

2

$\lambda = 1$

$$(2.3 - 0.9 - 1.5)^2 +$$

$$(5.3 - 2.7 - 1.5)^2 + (0.9)^2$$

$$= (0.1)^2 + (1.1)^2 + (0.9)^2$$

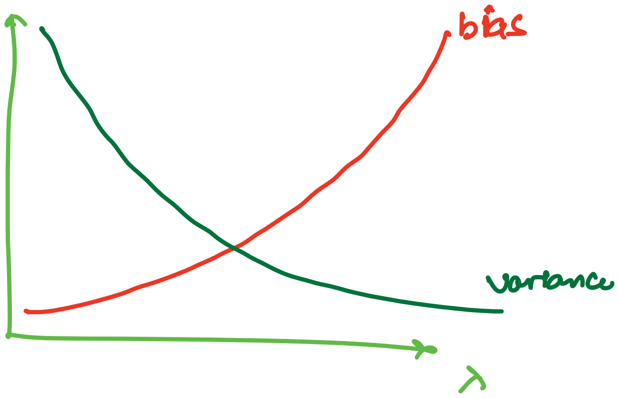
$$= 2.03$$

$$\begin{array}{ccc} x_1 & x_2 & x_3 \\ \downarrow & \downarrow & \downarrow \\ w_1 & w_2 & w_3 \end{array}$$

$$L = \sum_{i=1}^{m}\left(y^{(i)} - \hat{y}^{(i)}\right)^2 + \lambda \|w\|^2$$

Ridge Regularization

$\lambda(\omega_1^2 + \omega_2^2 + \omega_3^2 \cdots \omega_n^2)$



λ inc → weights less importance

→ high bias

Lasso Regularisation:

$$L = \sum_{i=1}^{m} (y^{(i)} - \hat{y}^{(i)})^2 + \lambda \|\omega\|$$

Lasso Regularisation

$|\omega_1| + |\omega_2| + |\omega_3| \ldots |\omega_n|$

Ridge

$\lambda \uparrow \quad \omega \downarrow$

(close to 0 never to 0)

Lasso

$\lambda \uparrow \quad \omega = 0$

feature selection

Elastic Net:

$$L = \frac{1}{m} \sum (y^{(i)} - \hat{y}^{(i)})^2 + a\|\omega\|^2 + b\|\omega\|$$

$\lambda = a + b$

$l1\_ratio = \frac{a}{a+b}$

$\lambda = 1$ $\quad l1\_ratio = 0.5 \Rightarrow a = 0.5 \quad b = 0.5$

$l1\_ratio : 0.9 \Rightarrow \underline{\underline{a = 0.9}} \quad b = 0.1$

more importance