Regression Analysis (predictive Data analysis)
predict real valued output for given input.

i/p	o/p
Amount of spend for Ad	sales (profit)
Amount of carbon-14	Age of fossil
Rainfall in previous month	Rainfall in current
previous stock prices	Future stock prices
No of cigarettes	lung capacity
size of house, area, no. of rooms location	House price

Types of Regression
Linear Regression :-
(1) simple linear Regression (Univariate)
(2) Multiple linear Regression (Multivariate)
Polynomial Regression

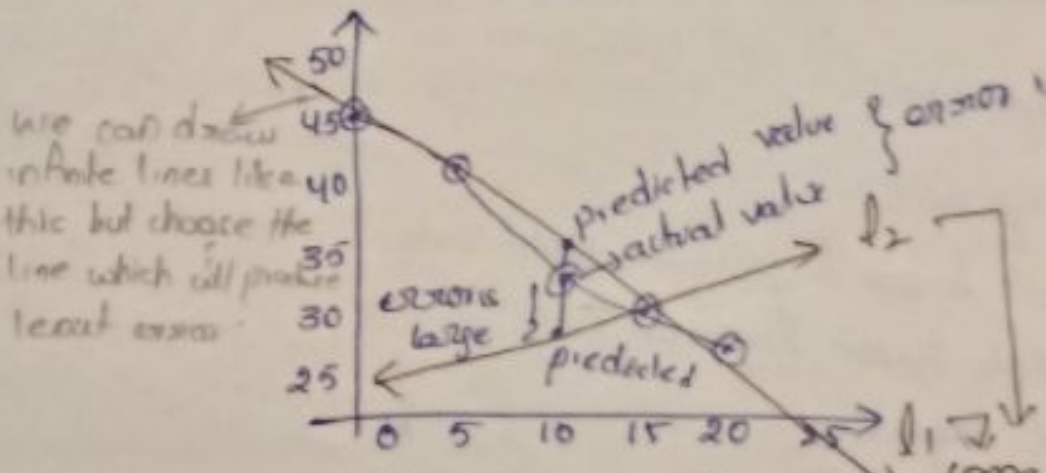
(1) Simple linear Regression (one i/p, one o/p)
Data :- pairs of variables
$D = \{(x_i, y_i) / i = 1 \text{ to } n\}$
x_i = Today's temperature
y_i = Tomorrow's Rainfall
x_i is independent Variable
y_i is dependent Variable.

No of cigarettes (x_i)	lung capacity (y_i)
0	45
5	42
10	33
15	31
20	29
25	?

Task:

predict real valued y for given real valued x using a regression model f.

Model Structure :- $f(x) = \omega_0 + \omega_1 x$

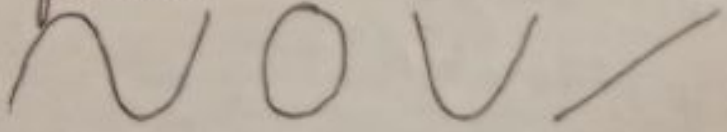


We can draw infinite lines like this but choose the line which will produce least error.

some function which is slightly varied from original

We can use lineplot (or) scatter plot

functions :-



If line slope $\Rightarrow y = mx + c$

m = slope

c = intercept.

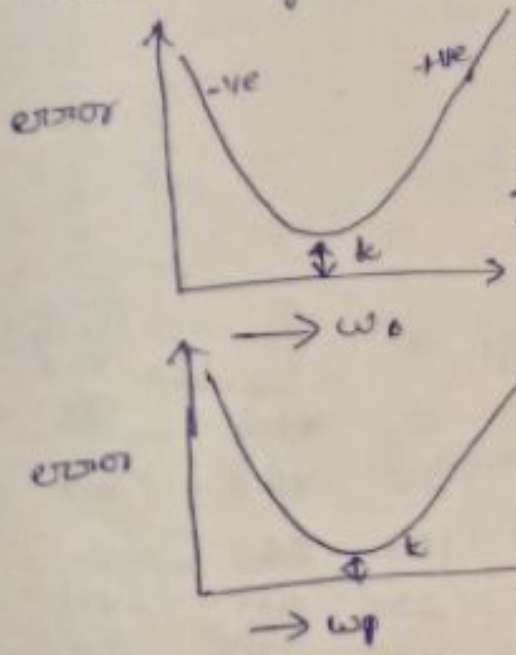
If m or c change line may change but x is constant it will not go beyond line.

Model Parameter :- $\theta = \{w_0, w_1\}$.

Error function :- sum Absolute Error = $\sum_{i=1}^{n} |y_i - f(x)|$

sum Squared Error = $\sum_{i=1}^{n} (y_i - f(x_i))^2$

Training :- given a training set find the value of regression parameters such that model best fits the for training data



$\frac{dy}{dx}$

$\frac{dE}{dw_0} = 0$

From these a choose the one which has least error (k).

Ex: w_1

$\frac{dE}{dw_1} = 0$ → select if zero. or else select w_1.

E = Sum squared error = $\sum_{i=1}^{n} (y_i - f(x_i))^2 = \sum_{i=1}^{n} (y_i - w_0 - w_1 x_i)^2$

[derivative]

$$\frac{\partial E}{\partial w_0} = -2 \sum_{i=1}^{n} (y_i - w_0 - w_1 x_i) = 0$$

$$\Rightarrow \Sigma y_i - \Sigma w_0 - w_1 \Sigma x_i = 0$$

$$\Rightarrow \Sigma w_0 = \Sigma y_i - w_1 \Sigma x_i$$

$$= n w_0 = \Sigma y_i - w_1 \Sigma x_i$$

$$\therefore w_0 = \frac{\Sigma y_i - w_1 \Sigma x_i}{n}$$

$$\boxed{w_0 = \bar{y} - w_1 \bar{x}}$$

$$\frac{\partial E}{\partial w_1} = -2 \sum_{i=1}^{n} (y_i - w_0 - w_1 x_i) x_i = 0$$

$$\Rightarrow \sum x_i y_i - w_0 \sum x_i - w_1 \sum x_i^2 = 0$$

$$\Rightarrow \sum x_i y_i - (\bar{y} - w_1 \bar{x}) n\bar{x} - w_1 \sum x_i^2 = 0$$

$$\Rightarrow \sum x_i y_i - n\bar{x}\bar{y} - n\bar{x}^2 w_1 - w_1 \sum x_i^2 = 0$$

$$\Rightarrow \sum x_i y_i - n\bar{x}\bar{y} - n\bar{x}^2 w_1 - w_1 n^2 x_i^2$$

$$\boxed{w_1 = \frac{\sum x_i y_i - n\bar{x}\bar{y}}{\sum x_i^2 - n\bar{x}^2}}$$

$\bar{x} = \frac{\sum x_i}{n}$

25/1/19

x_i No of cigars	y_i Wng capacity	$x_i y_i$	x_i^2	$\hat{y}_i = w_0 - w_1 x$	$(\hat{y} - \bar{y})$	$(\hat{y} - \bar{y})^2$
0	45	0	0	44.6	8.6	73.96
5	42	210	25	40.3	4.3	18.49
10	33	330	100	36	0	0
15	31	465	225	31.7	-4.3	18.49
20	29	580	400	27.4	-8.6	73.96
		1585	750			

$$\bar{x} \Rightarrow \frac{0+5+10+15+20}{5}$$

$$\Rightarrow \frac{50}{5} = 10$$

$$\bar{y} = \frac{45+42+33+31+29}{5}$$

$$= \frac{180}{5}$$

$$= 36$$

$$\frac{\partial E}{\partial \omega_1} = -2 \sum_{i=1}^{n} (y_i - \omega_0 - \omega_1 x_i) \cdot x_i = 0$$

$$\Rightarrow \sum x_i y_i - \omega_0 \sum x_i - \omega_1 \sum x_i^2 = 0$$

$$\Rightarrow \sum x_i y_i - (\bar{y} - \omega_1 \bar{x}) n\bar{x} - \omega_1 \sum x_i^2 = 0$$

$$\Rightarrow \sum x_i y_i - n\bar{x}\bar{y} - n\bar{x}^2 \omega_1 - \omega_1 \sum x_i^2 = 0$$

$$\Rightarrow \sum x_i y_i - n\bar{x}\bar{y} - n\bar{x}^2 \omega_1 - \omega_1 n^2 x_i^2$$

$$\boxed{\omega_1 = \frac{\sum x_i y_i - n\bar{x}\bar{y}}{\sum x_i^2 - n\bar{x}^2}}$$

$\bar{x} = \frac{\sum x_i}{n}$

25/1/19

x_i No of cigars	y_i Wng capacity	$x_i y_i$	x_i^2	$\hat{y}_i = \omega_0 - \omega_1 x$	$(\hat{y} - \bar{y})$	$(\hat{y} - \bar{y})$
0	45	0	0	44.6	8.6	73.96
5	42	210	25	40.3	4.3	18.49
10	33	330	100	36	0	0
15	31	465	225	31.7	-4.3	18.49
20	29	580	400	27.4	-8.6	73.96
		1585	750			

$$\bar{x} \Rightarrow \frac{0+5+10+15+20}{5}$$

$$\Rightarrow \frac{50}{5} = 10$$

$$\bar{y} = \frac{45+42+33+31+29}{5}$$

$$= \frac{180}{5}$$

$$= 156 \; 8 \; 36$$

$w_0 = 36 - (w_1 \times 10)$

$w_1 = \frac{1585 - 1800}{750 - 500}$

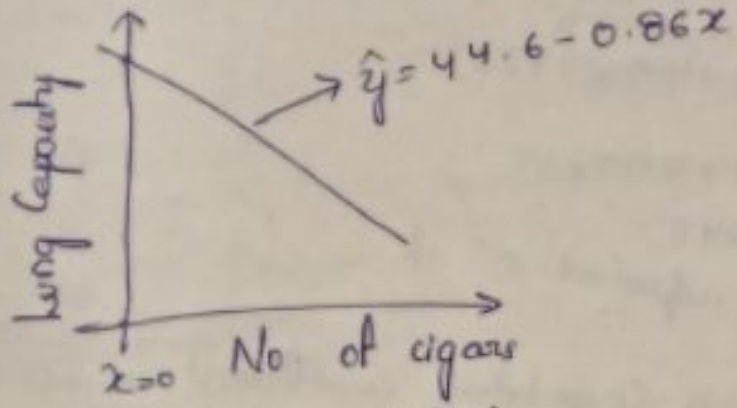
$\Rightarrow \frac{-215}{250}$

$= -0.86$

$w_0 = 36 - (-0.86)(10)$

$= 36 + 8.6$

$= 44.6$



$x = 0$ then $y = 44.6$.

$(y_i - \bar{y})^2$

81
36
25
49

Now we have to look whether the model is good enough or not can be done by coefficient of Determination (R^2)

R^2 measure % of variation in y explained by the model

$R^2 = \frac{\text{Explained Variation}}{\text{Total Variation}} = \frac{SSM}{SST}$

$= \frac{\sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2} = \frac{184.9}{200} \Rightarrow 0.9245$

92% variation is explained by the model.

100% variation will not be explained and there will be noise for all the time.

$$\text{Adjusted } R^2 = 1 - \left(\frac{N-1}{N-d}\right)(1-R^2)$$

N = no. of data points = 5

d = no of features = 1 [(xi) input]

$$1 - \left(\frac{5-1}{5-1}\right)(1-(0.9245)^2)$$

$$1 - (1 - 0.85470025)$$

$$1 - (0.14529975)$$

$$= 0.85470025$$

$$\Rightarrow 0.9245$$

Features ↑ → adjusted R^2 ↓

r = pearson correlation coefficient = -0.96151

$r^2 = (-0.96151) = 0.9245$

This will work for 2 variables only.

14/19

Multiple linear Regression

i/p: TV Ad, Internet Ad, Newspaper Ad — o/p: sales

Data $\{x_{11}, x_{12}, \ldots x_{1d}, y_1\}$

$\{x_{21}, x_{22} \ldots x_{2d}, y_2\}$

$\{x_{n1}, x_{n2} \ldots x_{nd}, y_n\}$

$D = \{(x_{i1}, x_{i2}, \ldots x_{id}, y_i) / i = 1, 2, 3 \ldots n\}$

Task :- predicting the values of y

Model Structure :

$\hat{y} = f_y = f(x_1, x_2 \ldots x_d)$

$= w_0 + w_1 x_1 + w_2 x_2 + \ldots + w_0 x_n$

↳ not polynomials (power 1), so variables

$= w_0 + \sum_{i=1}^{d} w_i x_i$

Parameters :- $d+1$ parameters

$\theta = \{w_0, w_1, \ldots, w_d\}$

↓

we have to find these parameters such that error function is minimum

Error function :-

$E = SSE = \sum_{i=1}^{n} (y_i - f(x_{i1}, x_{i2} \ldots x_{id}))^2$

Training :-

$d+1$ parameters $\Rightarrow$ $d+1$ equations

Matrix Notation is useful to find the Parameters $w_0, w_1, \ldots w_d$.

$3x + 4y = 6$

$9x + 22y = 26$

$\begin{bmatrix} 3 & 4 \\ 9 & 22 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 6 \\ 26 \end{bmatrix}$.

$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 3 & 4 \\ 9 & 22 \end{bmatrix}^{-1} \begin{bmatrix} 6 \\ 26 \end{bmatrix}$

Normal Equation (for Simple Linear Regression)

$$\frac{\partial E}{\partial \omega_0} = -2 \sum_{i=1}^{n} (y_i - \omega_0 - \omega_1 x_i) = 0$$

$$\frac{\partial E}{\partial \omega_1} = -2 \sum_{i=1}^{n} (y_i - \omega_0 - \omega_1 x_i) x_i = 0$$

$$X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \qquad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

$$\begin{bmatrix} n & \Sigma x_i \\ \Sigma x_i & \Sigma x_i^2 \end{bmatrix} \begin{bmatrix} \omega_0 \\ \omega_1 \end{bmatrix} = \begin{bmatrix} \Sigma y_i \\ \Sigma x_i y_i \end{bmatrix}$$

↓ Above equation And matrix format are equal.

$$\begin{bmatrix} \omega_0 \\ \omega_1 \end{bmatrix} = \begin{bmatrix} n & \Sigma x_i \\ \Sigma x_i & \Sigma x_i^2 \end{bmatrix}^{-1} \begin{bmatrix} \Sigma y_i \\ \Sigma x_i y_i \end{bmatrix}$$

$$X^T X = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ x_1 & x_2 & x_3 & \dots & x_n \end{bmatrix} \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}$$

$$= \begin{bmatrix} n & \Sigma x_i \\ \Sigma x_i & \Sigma x_i^2 \end{bmatrix}$$

$$X^T y = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ x_1 & x_2 & x_3 & \dots & x_n \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix}$$

$$= \begin{bmatrix} \Sigma y_i \\ \Sigma x_i y_i \end{bmatrix}$$

$$\boxed{W = [X^T X]^{-1} [X^T y].}$$

Normal Equation (for Multiple linear Regression)

$$X = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1d} \\ 1 & x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & & & & \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nd} \end{bmatrix}_{n \times (d+1)} \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}_{n \times 1}$$

$$W = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_d \end{bmatrix}_{(d+1) \times 1}$$

$$\boxed{W = (X^T X)^{-1} (X^T y)} \Rightarrow \underline{(d+1) \times (d+1)}$$

square matrix and symmetric

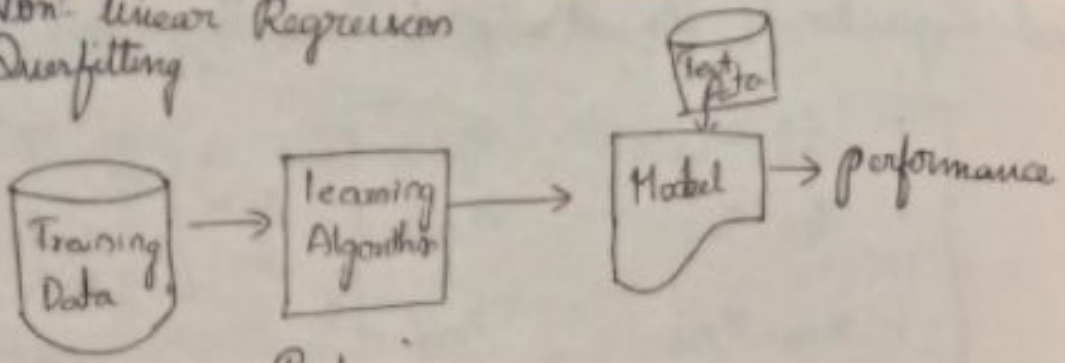
Every square symmetric matrix is invertible

Limitation of linear Regression

~~The relationship~~

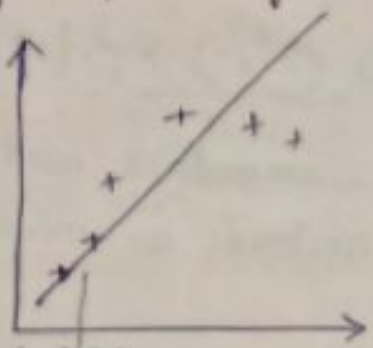
* True Relationship of x and y might be non linear
* complexity of the Algorithm $O(Nd^2 + d^3)$
  - N = no. of datapoints.
  - d = no. of features
* It includes all variables in the model. but all variables may not be related to y
* This approach may fail if two variables are Correlated namely multi correlation
* correlation / colinearity among x variables can cause numerical instability

solving Non-linear Regression
Overfitting

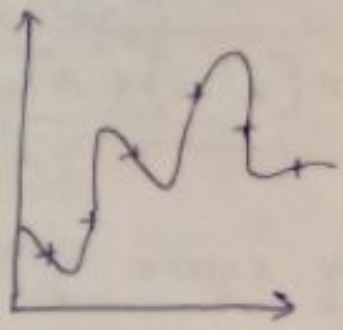


Reducing training Error

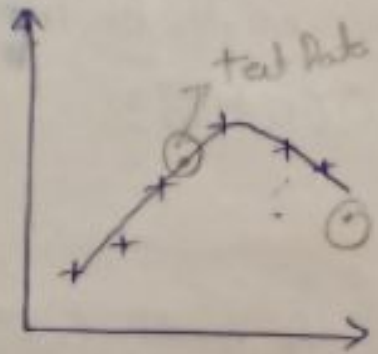
Complexity Vs Goodness of fit



more training exam and it is time

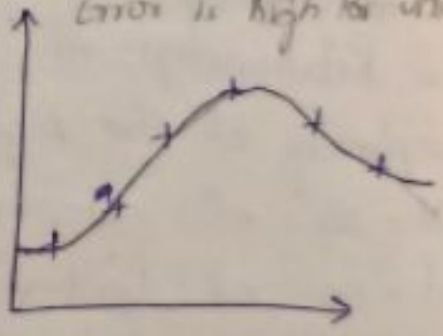


* Very Complex, less train
$y = w_0 + \sum^{20} w_i x$
Error is high for unknown



$y = w_0 + w_1 x + w_2 x^2$

* some training exp
* works well for unknown data

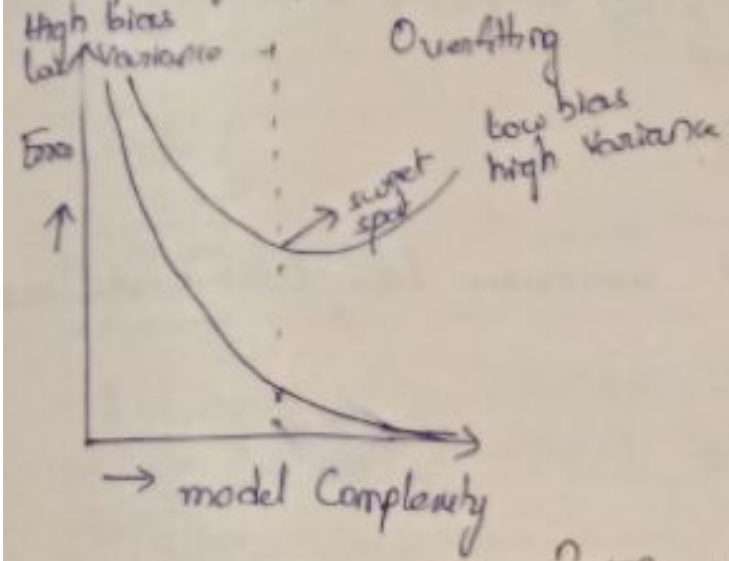


$y = w_0 + w_1 x + w_2 x^2$

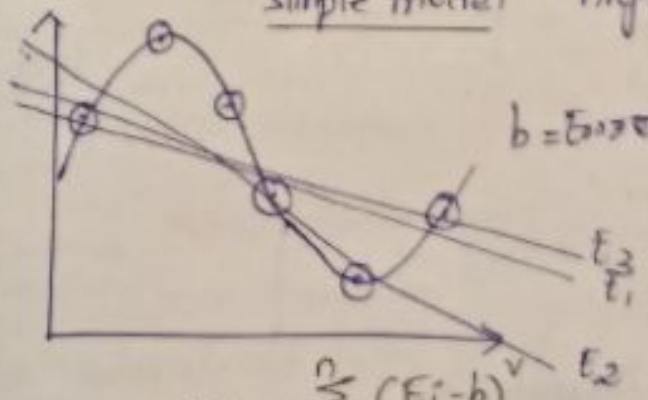
Occam's Razor Principle:-
simpler solutions are more likely to be correct than the complex one. ~~use the si~~

Use the simplest model which gives acceptable accuracy on training set

Complexity of model Vs generalization

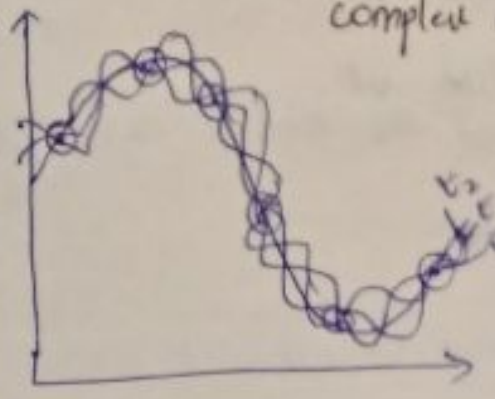


① Bias of a model = $\frac{\sum_{i=1}^{n} E_i}{\text{Simple model}} = b$ low Variance high bias.



$b = \text{Error} = \frac{E_1+E_3+E_2}{3}$

② Variance model = $\frac{\sum_{i=1}^{n} (E_i - b)^2}{n-1}$ complex model



$b = \frac{E_1+E_2+E_3}{3}$

$V = \frac{\sum_{i=1}^{3} (E_i - b)^2}{2}$

low bias
high variance

Bias :-

Bias of a model is the average error it makes on various training sets

$$B = \left(\sum_{i=1}^{n} E_i\right)/n$$

Variance :-

Variation of ~~a model~~ error it makes on different training sets $V = \frac{\sum_{i=1}^{n} (E_i - b)^2}{n}$

we need to find a

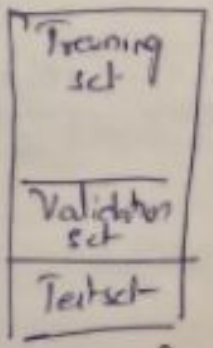
Find a model that minimizes both bias and variance

Regularization :- L_1

L_2

Model Validation :-

① Holdout method :-



purpose of training set

use this data to train each model

Purpose of Validation set

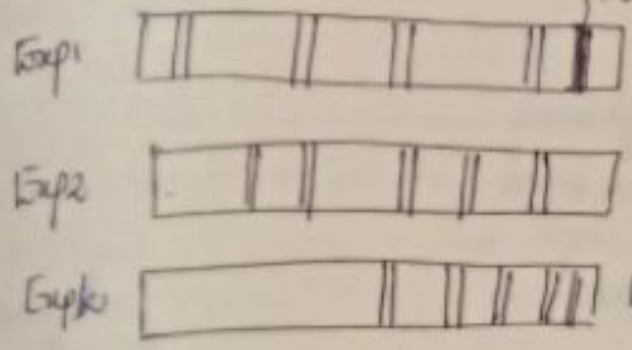
use this data to test the models and select the model with minimum error.

Purpose of Test set

used this data to calculate unbiased error for the selected model

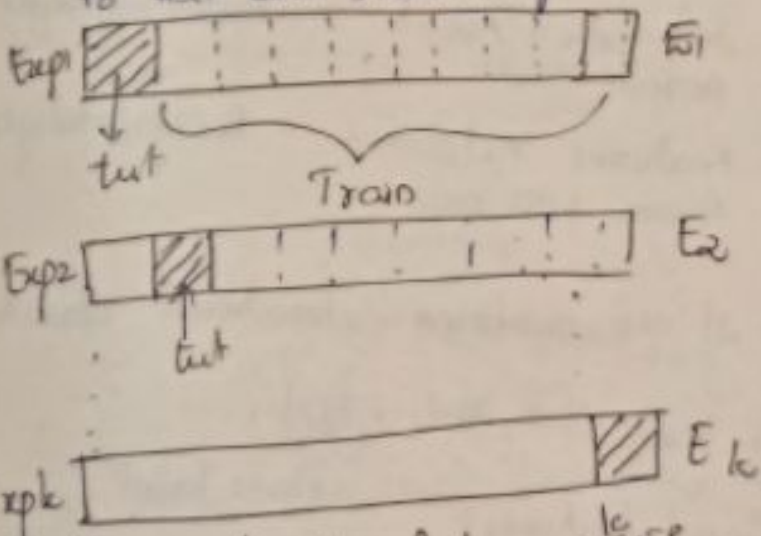
③ Cross Validation:-

ⓐ Random Subsampling: Randomly selecting a test set and test data remaining for training



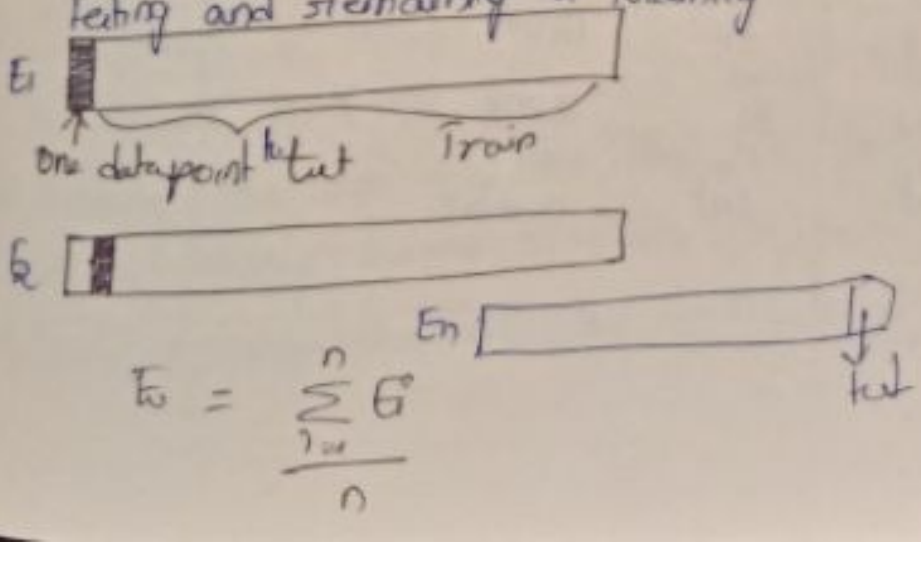
True estimate of error $E = \frac{\sum_{i=1}^{k} E_i}{k}$

ⓑ k-fold Cross validation: use a fold of dataset to test and remaining for training.



True estimate of $k = \frac{\sum_{i=1}^{k} E_i}{k}$

ⓒ Leave one out: Each time selecting a datapoint for testing and remaining for training



$$E_0 = \frac{\sum_{i=1}^{n} E_i}{n}$$

11/2/19

Classification - supervised learning

- Decision Trees
- Baye's Classifier
- k Nearest Neighbours
- support vector machine
- ANN
- Ensemble method

Task	i/p	o/p (categorical)
① Categorizing Email	Features Extracted from email header and Content	spam / not spam
② Loan Approval	Information About person	Approve / Reject
③ Identifying Tumour cell	Features extracted from MRI scan	Benign / Maligant

Decision Tree :- it is a tree structured classifier.

Data :- $\{x_{11}, x_{12}, \ldots x_{1d_1}, y_1\}$

($x_{11}, x_{12}, \ldots x_{1d_1}$: input features; y_1 : class label)

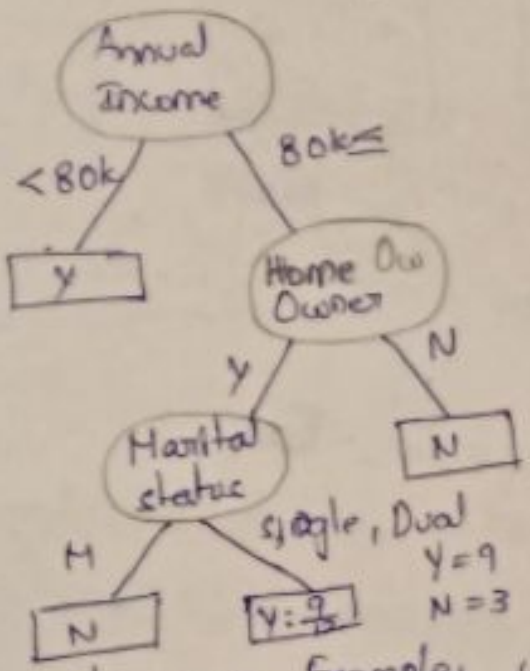
ID	Home owner	Marital status	Annual Income	Defaulter
1	Y	M	125k	N
2	N	N	220k	Y
	↑ cat	↑ cat	↑ number	

Task:- The task is to learn a model that maps each attribute set x into one of the predefined class label y

N	S	180k	?

Model structure :- - A tree
- Two types of nodes
  - Decision Node - Internal Node - Attribute
  - Consequence Node - Leaf Node - class

Eg:-

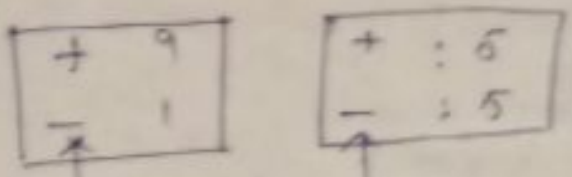


*given some training Examples we have to generate Decision Tree
- many Decision Trees are possible
- we prefer a smaller Tree
  means less no of nodes and low depth
* finding the smallest Tree that fits the data is computationally hard problem
* Greedy Algorithm - will be used (local optimization)

Parameters of model
- which Attribute to select for split ?
- when to stop ?

① All attributes belongs to the same class
② All Attributes are already used (no more features)
③ Too few examples to make Informative split

Error Function: Measure of Node impurity.



more pure less pure

① Gini index :-
② Entropy
③ Misclassification Error

above 3 are used for calculating information gain

Training will done by below Algorithms

- Hunt's Algorithm
- CART
- ID3, C45
- SLIQ, SPRINT

Basic outline of ID3 Algorithm : Top down Approach

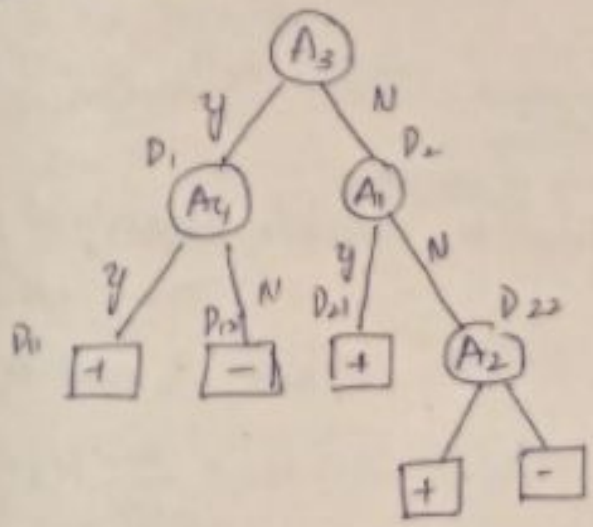
A = { A1, A2, A3, A4 }

① Select best Attribute among all selected for Root node
② Assign the selected Attribute as Decision Attribute for the Node

③ For each value of the Attribute create new descendent

④ sort the training Examples ~~to~~ to leaf nodes according attribute value

⑤ If all training Examples are perfectly classified then stop else iterate over new leaf Nodes.



4/2/19

which attribute to select ?

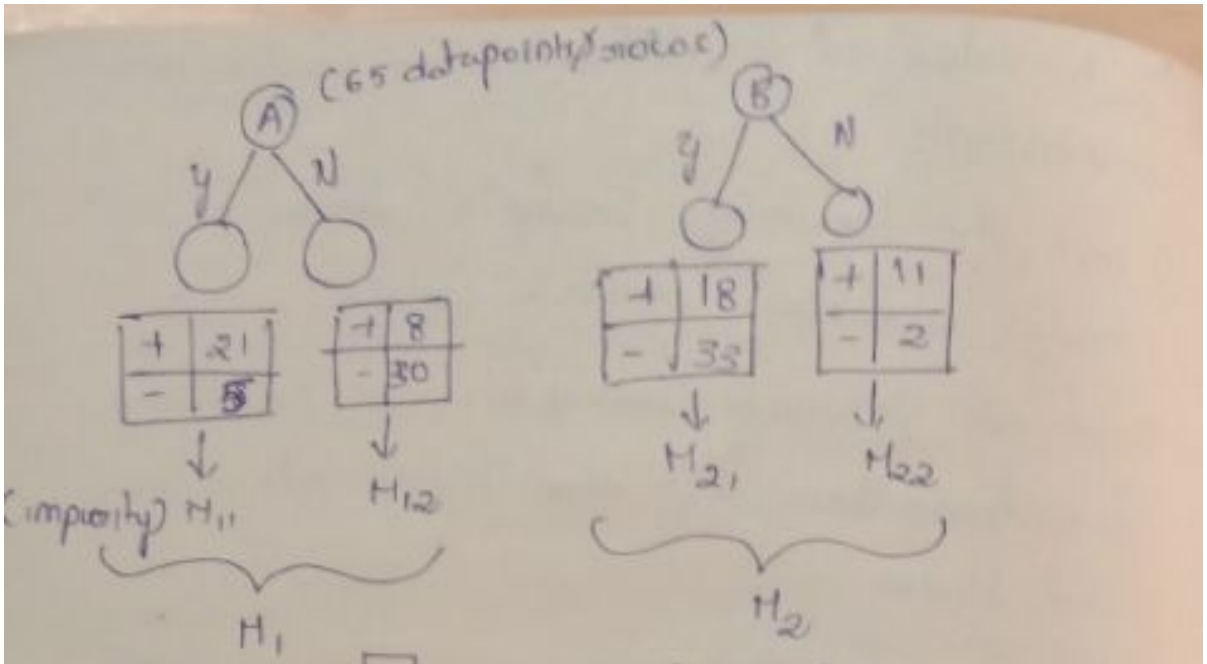
- which attribute gives smallest errors
- greedy Approach
- Measure of Impurity
  - Entropy
  - Gini
  - Misclassification Error

} Information gain

+	29
-	35

Attribute = {A, B}

P = Impurity



Gain = P - H_1 (impurity of children)

↳ impurity of parent

Gain = P - H_2

we don't know how to calculate impurity of parent (P)

" " " " " " " children (H_1)

" " " " combine H_{11} and H_{12}

<u>combining H_{11} and H_{12}</u> :- $H_1 = \frac{26}{64} H_{11} + \frac{38}{64} H_{12}$

① finding Best Attribute :- $H_2 = \frac{n_1}{n} H_{21} + \frac{n_2}{n} H_{22}$

-1 Compute the impurity measure p before splitting

-2 compute the impurity measure after splitting

step-A: compute the impurity measure for each child node

Measure of Impurity

Entropy :-

Expectation : It is the fancy name for Average

Team (x) :	A	B	C	D
Probability of wins	0.3	0.4	0.2	0.1
Reward (v(x)) :	10k	5k	10k	-30k

what is the Expected Reward?

$$\text{Expected Reward} = \frac{3\times10 + 4\times5 + 2\times10 + 1\times(-30)}{10}$$

$$= \frac{3}{10}\times10 + \frac{4}{10}\times5 + \frac{2}{10}\times10 + \frac{1}{10}\times(-30)$$

$$= 0.3\times10 + 0.4\times5 + 0.2\times10 + 0.1\times-30$$

$$= \sum \text{probability of } (v(x)) * \text{value of } x$$

For any Random Variable :-

$$E(X) = \sum_{i \in A,B,C,D} p(x) \cdot v(x)$$

Information content :-

$$IC(A) \propto \frac{1}{p(A)}$$

$$IC(A) = f(p(A)) = \log\left(\frac{1}{p(A)}\right) = -\log(p(A))$$

$IC(A) = -\log(p(A))$

Information content of Independent Events:

X = which team is going to win

Y = lunch is going to be good in mess

$$IC(x \cap y) = IC(x) + IC(y)$$

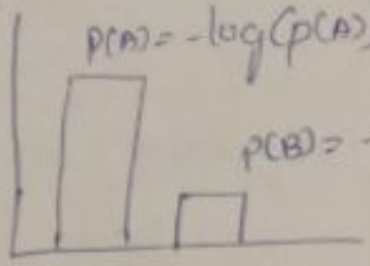
$$f(p(x \cap y)) = f(p(x)) + f(p(x))$$

$$f(p(x)) \cdot f(p(x)) = f(p(x)) + f(p(x))$$

$$\log(p(x) \cdot p(y)) = \log(p(x)) + \log(p(y))$$

[x, y are Independent

Entropy:-



Expected IC = P(A) I(A) + P(B) I(B)

$= -p(A)\log_2(p(A)) - p(B)\cdot\log_2(p(B))$

= Entropy

+	29
-	35

29/64

Calculable Entropy

35/64

$$\text{Entropy} = \frac{-29}{64}\log_2\left(\frac{29}{64}\right) - \frac{35}{64}\log_2\left(\frac{35}{64}\right)$$

$$= -\frac{29}{64}\log_2\left(\frac{29}{64}\right) \Rightarrow \begin{matrix}+0.155776209 \\ -0.143342461\end{matrix}$$

$$\frac{35}{64}\log_2\left(\frac{35}{64}\right)$$

+	10
-	0

0

E(0) = 0.99

+	0
-	10

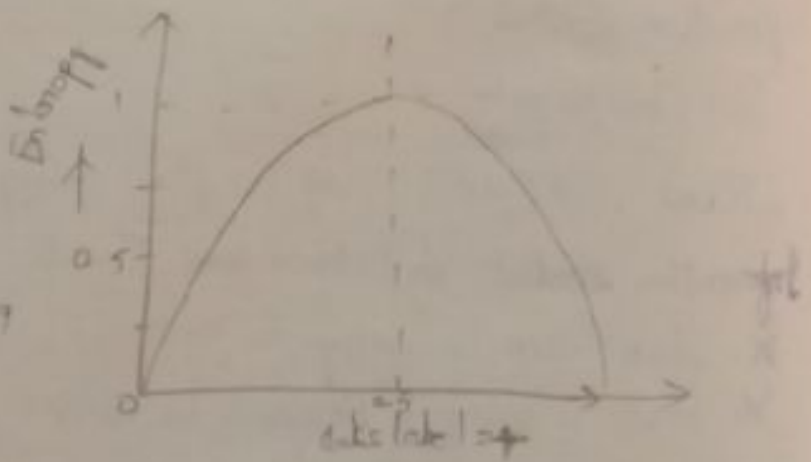
0

+	5
-	5

1

+	9
-	1

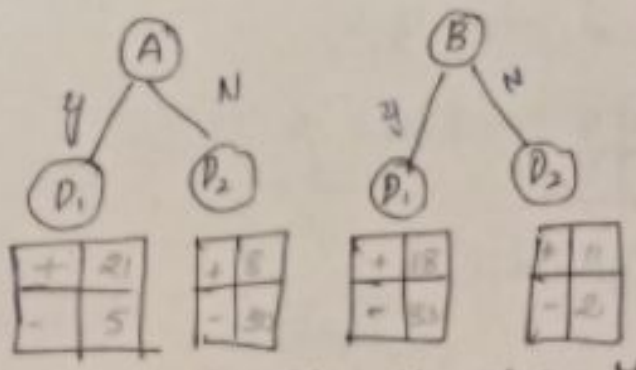
0.469



5/2/19

D = | + | 29 |
    | - | 35 |

E(D) = 0.99



$H_{11} = E(D_1)$ = 0.71 $H_{12} = E(D_2)$ = 0.74 $H_{21} = E(D_1)$ = 0.94 $H_{22} = E(D_2)$ = 0.62

play or not ?

Day	outlook	Temp	Humidity	wind	play
1	sunny	Hot	High	weak	No
2	sunny	Hot	High	strong	No
3	overcast	Hot	High	w	yes
4	Rainy	Mild	High	w	yes
5	Rainy	cool	Normal	w	yes
6	Rainy	cool	Normal	s	no
7	overcast	cool	Normal	s	yes
8	sunny	Mild	High	w	no
9	sunny	cool	Normal	w	yes
10	Rainy	mild	Normal	w	yes
11	sunny	mild	Normal	s	yes
12	overcast	mild	High	s	yes
13	overcast	Hot	Normal	w	yes
14	Rainy	Mild	High	s	NO

(A) $Gain = 0.99 - \left(\frac{26}{64} \times 0.71 + \frac{38}{64} \times 0.74\right)$

$= 0.99 - 0.73$

$= 0.26$

(B) $gain = 0.99 - \left(\frac{51}{64} \times 0.94 + \frac{13}{64} \times 0.62\right)$

$\Rightarrow 0.99 - 0.62$

Reduction in impurity = 0.37. 0.115

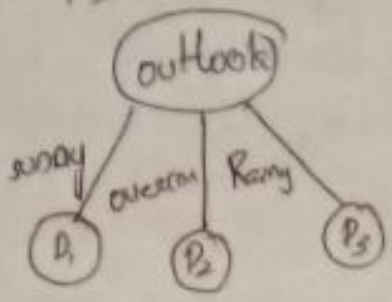
select A for split since high gain

Entropy of root node

D = | yes | 9 |
| --- | --- |
| NO | 5 |

$E(D) = 0.94$

Gain (Outlook, D)



class	count		
	S	O	R
yes	2	4	3
NO	3	0	2

$E(D_1) = 0.97 \Rightarrow -\frac{2}{5}\log_2(\frac{2}{5}) - \frac{3}{5}\log_2(\frac{3}{5})$

$E(D_2) = 0$

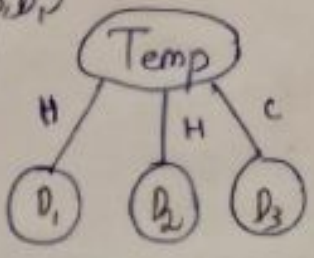
$E(D_3) = 0.97$

gain (Outlook, D) = $0.94 - (0.97 \times \frac{5}{14} + 0 \times \frac{4}{14} + 0.97 \times \frac{5}{14})$

$\Rightarrow 0.94 - (2(0.97 \times \frac{5}{14}))$

$\Rightarrow 0.25$

Gain (temp, D_1)

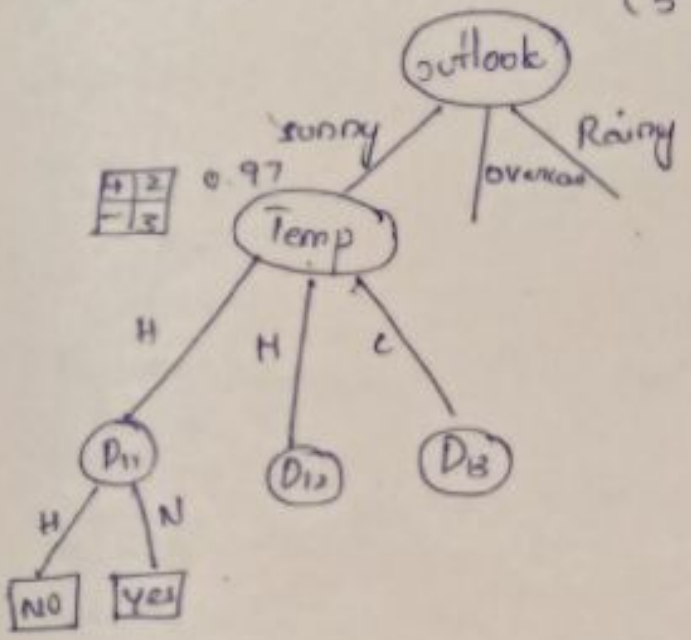


class	count		
	Hot	mild	cool
yes	2	4	3
NO	2	2	1

$E(D_1) = 1$ $\quad E(D_2) = 0$

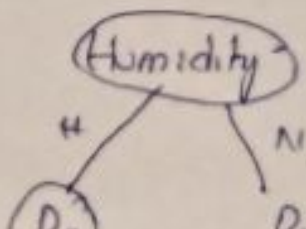
$E(D_3) = 1$

Gain(Temp, D_1) = $0.97 - \left(\frac{2}{5} \times 0 + \frac{2}{5} \times 1 + \frac{1}{5} \times$



~~0.97~~

= 0.57



Gain(Humidity, D_1) = 0.97 - 0

= 0.97

Gain(wind, D_1) = $0.97 - \left(\frac{3}{5} \times 0.918 + \frac{2}{5} \times 1\right)$

= 0.02

Gini index

$$Gini(t) = 1 - \sum_{i=0}^{c-1} [p(i/t)]^2$$

+	5 → $P = \frac{1}{2}$
−	5 → $P = \frac{1}{2}$

$$1 - \left[\left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2\right] = 0.5$$

+	10	1
−	0	0

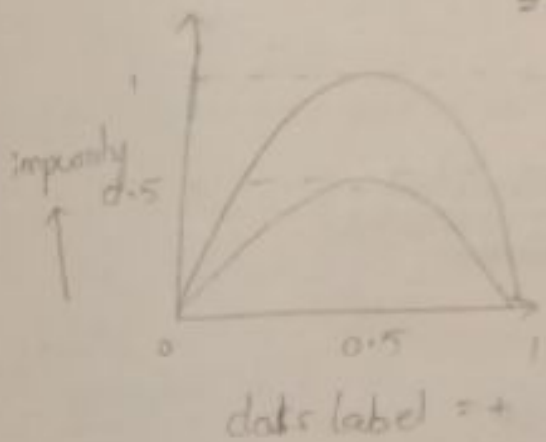
$Gini(t) = 1 - [(1)^2 + (0)^2] = 0$

+	10	0
−	10	1

$Gini(t) = 1 - [0^2 + (1)^2] = 0$

+	8	0.8
−	2	0.2

$$Gini(t) = 1 - [(0.8)^2 + (0.2)^2]$$
$$= 1 - [0.64 + 0.04]$$
$$= 0.32$$



Metrics to evaluate model performance (classification) :-

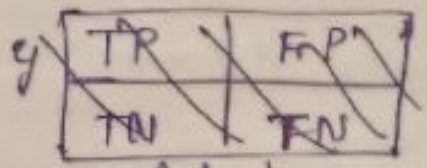
play ?

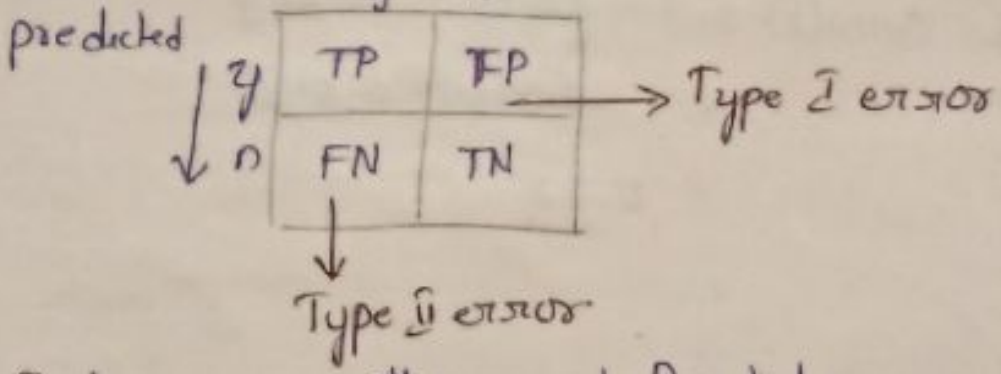
+	−
yes	no

data labels.

True label	predicted label	Result Type
yes	yes	True +ve (TP)
yes	no	True -ve (TN)
no	yes	false +ve (FP)
no	no	false -ve (FN)

Confusion Matrix :-





① Accuracy = $\frac{\#\text{ correct Prediction}}{\#\text{ total prediction}}$

$= \frac{TP + TN}{(TP + TN + FN + FP)}$

② Error Rate = $\frac{\#\text{ incorrect prediction}}{\#\text{ total prediction}}$

$= \frac{FP + FN}{TP + TN + FN + FP}$

Eg:-

True	predicted
y	N
N	N
N	N
y	y
y	y
n	n
y	n
y	y
n	n
N	y

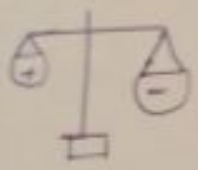
predicted \ Actual	y	N
y	TP 3	1
N	2	4

$$\text{Accuracy} = \frac{3+4}{10} = 0.7$$

$$\text{Error Rate} = \frac{2+1}{10} = 0.3$$

Limitation with Accuracy :-

- class imbalance problem.



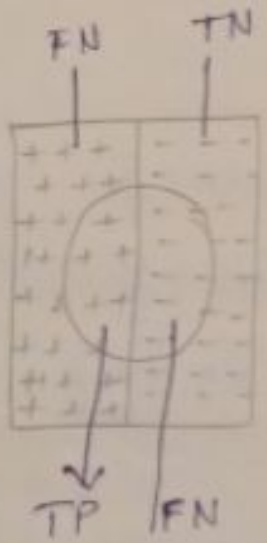
- very few + examples
- most are -ve Examples

Eg: 3% are cancer, 97% are not cancer

y	n
0	0
3	97

12/2/19

Precision And Recall



$$\text{precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP+FN}$$

Precision = $\frac{\text{O}}{\text{V}}$

→ measure of exactness

Recall = $\frac{\text{O}}{\square}$

→ measure of Completeness

F-Measure:- combines precision & Recall

$$F_1 = 2 * \frac{\text{Precision} \times \text{Recall}}{\text{precision} + \text{Recall}}$$

$0 \leq F_1 \leq 1$

1 = perfect classification

F_1 = evenly Weighted

F_2 = Weight recall more

F_3 = Weight precision more

0	0
3	97

Accuracy = 0.97 precision = 0, Recall = 0

$F_1 = 0$

Predicted \ Actual	cat	Dog	Rabbit
cat	5	2	0
Dog	3	3	2
Rabbit	0	1	11

cat ?

	+	−
+	5	2
−	3	

Dog ?

	+	−
+	3	5
−	3	

Rabbit ?

	+	−
+	11	1
−	2	

Precision = $\frac{5}{7}$ precision = $\frac{3}{8}$ Precision = $\frac{11}{12}$

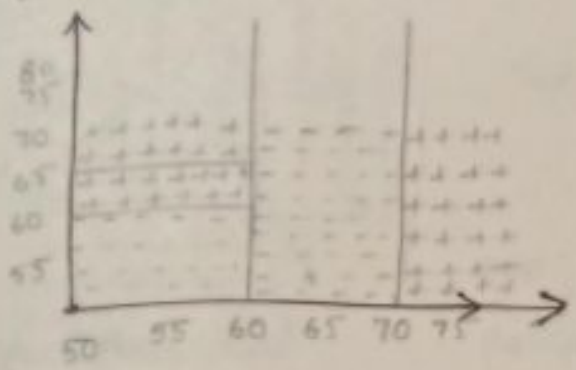
Recall = $\frac{5}{8}$ Recall = $\frac{3}{6}$ Recall = $\frac{11}{13}$

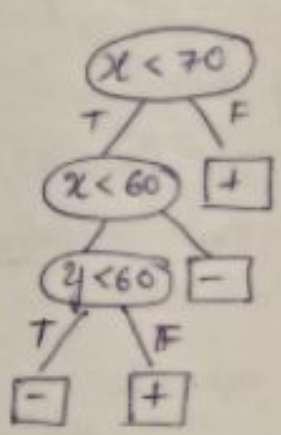
$F_1 = 2 \times \frac{\frac{5}{7} \times \frac{5}{8}}{\frac{5}{7} + \frac{5}{8}}$ $F_1 = 2 \times \frac{\frac{3}{8} \times \frac{3}{6}}{\frac{3}{8} + \frac{3}{6}}$ $F_1 = 2 \times \frac{\frac{11}{12} * \frac{11}{13}}{\frac{11}{12} + \frac{11}{13}}$

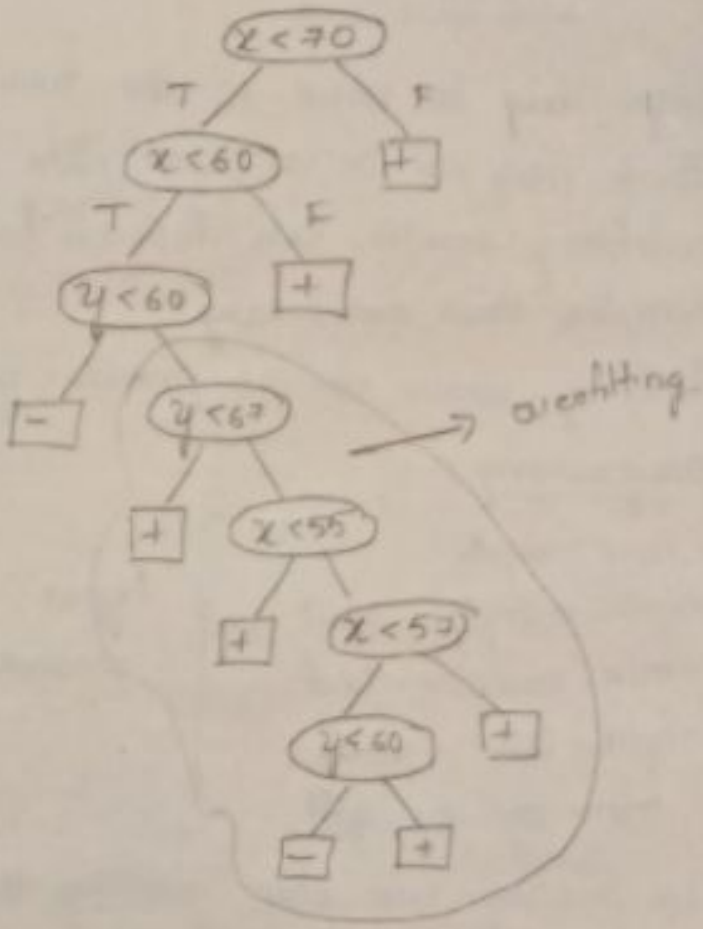
$\rightarrow 0.67$ $= 0.42$ $= 0.88$

$$F = \frac{0.67 + 0.42 + 0.88}{3} = 0.65$$

overfitting

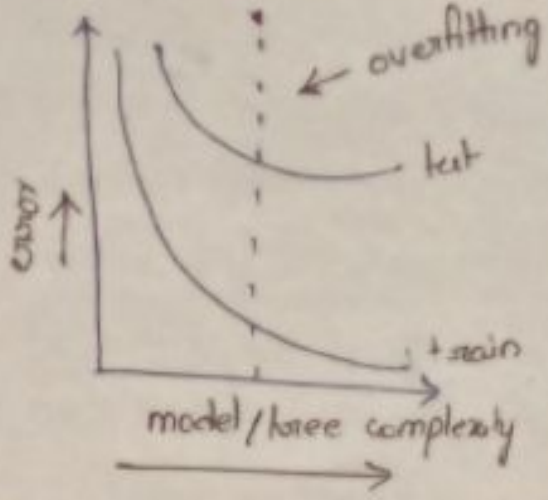


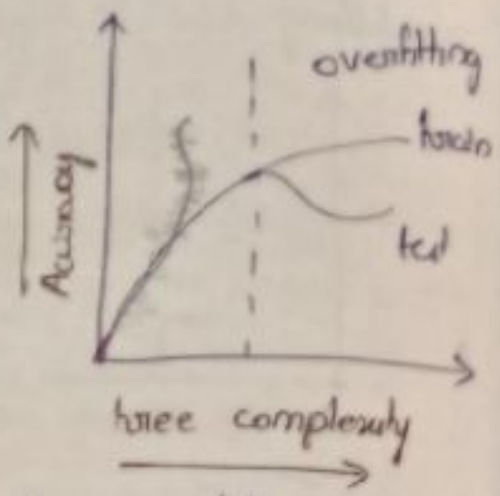




11th feb '19

A model m is said to overfit the training data if there is another model m', such that m' has more error than m on training data but m' has less error than m on test data.





- There may be noise in the training data
- There may not be sufficient data to take decision
- overfitting results in a decision tree that is more complex than necessary
- Training errors doesn't provide good estimate

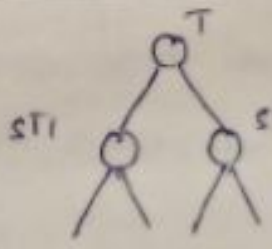
Pre pruning :-

- max-depth
- min-samples-split
- min-samples-leaf
- max-leaf-nodes
- min-impurity-split

} hyper parameters.

In pruning, we stop growing the tree

Cross Validation:-



validation Error

E(T)

E(T - ST1), E(T - ST2)

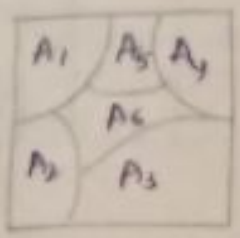
select the node whose removal gives least validation error

15th Feb 19

Baye's Classifier

probability crash Course

Axioms:-

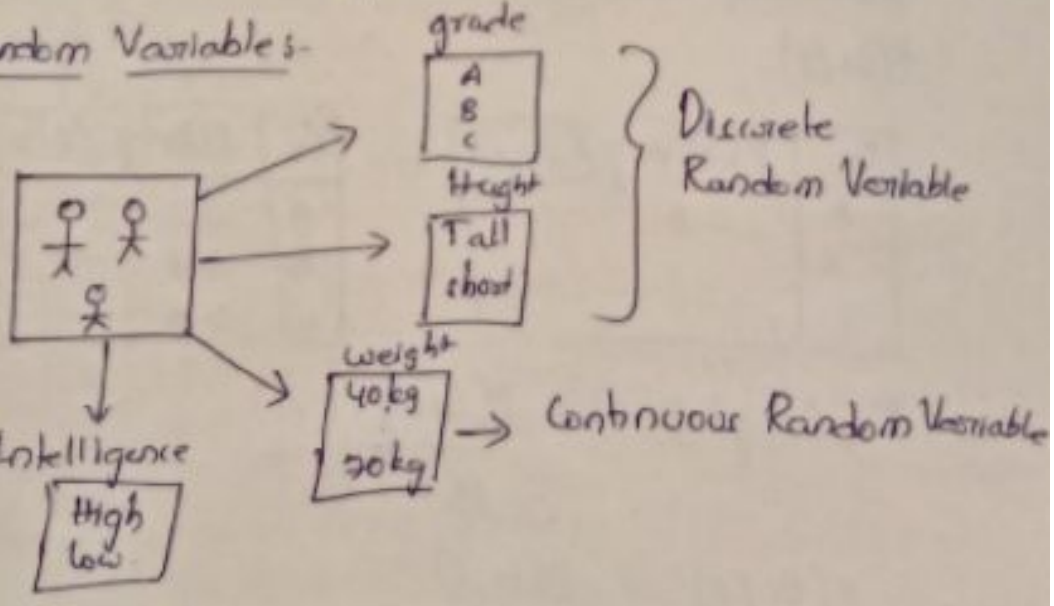


- For any event A, $p(A) \geq 0$
- $p(\cup A_i) = \sum_i (p(A_i)) = p(\Omega)$

Disjoint Events:- $A_i \cap A_j = \phi$

$\forall i \neq j$

Random Variables:-



Marginal Distribution:

A	40
B	30
C	30

G	$p(G=g)$
A	0.4
B	0.3
C	0.3

$p(G=g) \; \forall \; g \in A, B,$

$p(G)$

joint Distribution

G and I

G	I	$P(G=g, I=i)$
A	H	0.3
A	L	0.1
B	H	0.15
B	L	0.15
C	H	0.05
C	L	0.25

	A	B	C	
High	30	15	5	
low	10	15	25	
				100

$P(G=g, I=i) \; \forall \; (g,i) \in \{A,B,C\} \times \{H,L\}$

$P(G, I)$

Conditional Distribution

$P(G/I)$

G	$P(G=g / I=H)$
A	0.6
B	
C	

G	$P(G=g / I=L)$
A	0.2
B	0.3
C	0.5

$$P(G=g / I=i) = \frac{P(G=g, I=i)}{P(I=i)}$$

$$P(G/I) = \frac{P(G,I)}{P(I)}$$

$$P(G, I) = P(G/I) \times P(I)$$

↓ joint Distribution ↳ conditional ↳ marginal

Joint Distribution of Random Variable

x_1	x_2	x_3	. . . . .	x_n	$p(x_1, x_2, x_3, x_4 \ldots x_n)$

2^n ; Assuming each random Variables take 2 values

$p(x_1, x_2, \ldots x_n) = p((x_2, x_3, x_4 \ldots x_n)/x_1) \cdot p(x_1)$

$= p((x_3, x_4, \ldots x_n)/x_1, x_2) \cdot p(x_2/x_1) p(x_1)$

$= p((x_4 \ldots x_n)/x_1, x_2, x_3) \cdot p(x_3/x_1, x_2) \frac{p(x_2, x_3)}{p(x_1)}$

General Equation :-

$$p(x_i)_{i=2}^{n} \prod p(x_i^o / x_1 \ldots x_{i-1}^o)$$

Calculating Marginal distribution from joint Distribution

G	I	p(G, I)	
A	H	0.3	} → 0.4
A	L	0.1	
B	H	0.15	} 0.3
B	L	0.15	
C	H	0.05	} 0.3
C	L	0.05	

$$p(G = g) = \sum_{\forall i} p(G = g, I = i)$$

$$p(G = g) = \sum_{\forall i} p(G = g, I = i^o) \quad p(I = i) = \sum_{\forall g} p(G = g, I = i)$$

$$p(G) = \sum_{I} p(G, I) \quad p(I) = \sum_{G} p(G, I)$$

n- random Variables :-

$$p(x_1) = \sum_{x_2, x_3 \ldots x_n} p(x_1, x_2, \ldots x_n)$$

x_1	x_2	x_3	x_4	y
outlook	Temp	Humdty	wind	play
3	2	2	2	2

$P(y, x_1, x_2, x_3, x_4)$

$$P(y/x_1, x_2 x_3, x_4) = \frac{P(y, x_1, x_2, x_3, x_4)}{\sum_{x_1, x_2, x_3, x_4} P(y, x_1, x_2, x_3, x_4)}$$

6 Feb 14

Baye's classification :-

$P(G)$, $P(I)$ → marginal

$P(G, I)$ → joint

$P(G/I)$ → conditional

$P(G, I) = P(I/G)\, P(y)$

$\frac{R(x,y)}{P(x)} = P(x/y)\, P(y)$

$P(x, y) = P(x/y) \cdot P(y)$

$$\Rightarrow \frac{P(x,y)}{P(x)} = \frac{P(x/y) \cdot P(y)}{P(x)}$$

$$P(y/x) = \frac{P(x/y) \cdot P(y)}{P(x)}$$

$P(x/y)$ → likelihood probability

$P(y)$ → prior probability

$P(y/x)$ → posterior probability

⇒ 3 i/p variables x_1, x_2, x_3 and one o/p variable y

$$P(y/x_1, x_2, x_3) = \frac{P(y, x_1, x_2, x_3)}{P(x_1, x_2, x_3)}$$

$2^n - 1 = 2^4 - 1 = 15$

$$= \frac{P(x_1, x_2, x_3/y)\, P(y)}{P(x_1, x_2, x_3)}$$

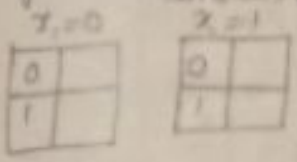
$$= \frac{p(x_2, x_3 | y\, x_1)\, p(x_1 | y)\, p(y)}{p(x_1, x_2, x_3)}$$

$$= \frac{p(x_3 | y\, x_1, x_2)\, p(x_2 | y\, x_1)\, p(x_1 | y)\, p(y)}{p(x_1, x_2, x_3)}$$

$$= \frac{p(x_3 | y)\, p(x_2 | y) \cdot p(x_1 | y)\, p(y)}{p(x_1, x_2, x_3)}$$

↳ using Naive Bayes classifier

y=0 : constant factor



Two Approaches:-

① Naive Bayes

② Bayes Belief N/w.

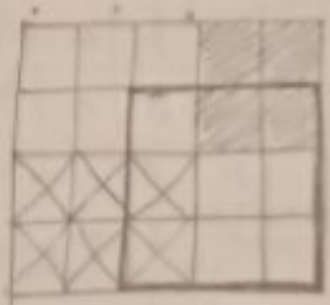
① Naive Bayes classifier

→ conditional independence

intelligence

Grade

Grade



⊠ = 6/20 ▨ = 4/20

$P(C_{IS} / I, G_R) = P(C_{IS} / I)$

↳ Naive Assumption

$P(x_2 / y\, x_1) = P(x_2 / y)$

↳ x_2 and x_1 are independent from each other in the class variable y (assumption)

P(⊠ / inner square) = 2/9

P(▨ / inner squares) = 2/9

The conditional independent Assumption states that the features are independent of each other given class label

Eg: day = {R, C, H, S} play = ? y-yes ; n-no

$$P(y/x) = \frac{P(\text{outlook}/y)\, P(\text{Temp}/y)\, P(\text{Humi}/y)\, P(\text{wind}/y)\, P(y)}{P(\text{outlook, Temp, Humi, wind})}$$

$$P(N/x) = \frac{P(\text{out}/n)\, P(T/n)\, P(H/n)\, P(w/n)\, P(n)}{P(O, T, H, W)}$$

likelihood tables :-

outlook:

	Y	N	P(y)	P(n)
S	2	3	2/9	3/5
O	4	0	4/9	0/5
R	3	2	3/9	2/5
total	9	5	1	1

Temperature:

	Y	N	P(y)	P(n)
H	2	2	2/9	2/5
M	4	2	4/9	2/5
C	3	1	3/9	1/5
total	9	5	1	1

Humidity:

	Y	N	P(y)	P(n)
H	3	4	3/9	4/5
N	6	1	6/9	1/5
total	9	5	1	1

wind

	Y	N	P(y)	P(n)
S	3	3	3/9	3/5
W	6	2	6/9	2/5
total	9	5	1	1

prior table:

class	Freq	prob
Y	9	9/14
N	5	5/14
Total	14	1

$$P(y/x) = \frac{P(R/y)\, P(C/y)\, P(H/y)\, P(c/y)\, P(y)}{P(x)}$$

$$= \frac{\left(\frac{3}{9}\right)\left(\frac{3}{9}\right)\left(\frac{3}{9}\right)\left(\frac{3}{9}\right)\left(\frac{9}{14}\right)}{\alpha} = \frac{0.00793}{\alpha}$$

$$P(w/x) = \frac{P(R/N)\,P(C/N)\,P(H/N)\,P(S/N)\,P(N)}{\alpha}$$

$$= \frac{\left(\frac{2}{5}\right)\left(\frac{1}{5}\right)\left(\frac{4}{5}\right)\left(\frac{3}{5}\right)\left(\frac{5}{14}\right)}{\alpha} = \frac{0.013714285}{\alpha}$$

Eg day = {S, H, N, W}

$$P(y/x) = \frac{P(\text{outlook}/y)\,P(T/y)\,P(H/y)\,P(w/y)\,P(y)}{P(O, T, H, w)}$$

$$P(N/x) = \frac{P(O/N)\,P(T/N)\,P(H/N)\,P(w/N)\,P(N)}{P(O, T, H, w)}$$

likelihood tables :-

outlook:

	y	n	P(y)	P(n)
S	2	3	2/9	3/5
O	4	0	4/9	0/5
R	3	2	3/9	2/5
Total	9	5	1	1

Temp:

	y	n	P(y)	P(n)
H	2	2	2/9	2/5
M	4	2	4/9	2/5
C	3	1	3/9	1/5
Total	9	5	1	1

Humidity:

	y	n	P(y)	P(n)
H	3	4	3/9	4/5
N	6	1	6/9	1/5
Total	9	5	1	1

Wind:

	y	n	P(y)	P(n)
S	3	3	3/9	3/5
W	6	2	6/9	2/5
Total	9	5	1	1

prior table :-

class	F	P
y	9	9/14
n	5	5/14
total	14	1

$$P(y/x) = \frac{P(c/y)\,P(H/y)\,P(N/y)\,P(w/y)\,P(y)}{P(c,H,N,w)}$$

$$= \frac{\left(\frac{2}{9}\right)\left(\frac{3}{9}\right)\left(\frac{6}{9}\right)\left(\frac{6}{9}\right)}{\alpha} = \frac{0.0219}{\alpha}$$

$$P(N/x) = \frac{P(c/n)\,P(H/n)\,P(N/n)\,P(w/n)\,P(n)}{P(c,H,N,w)}$$

$$= \frac{\left(\frac{3}{5}\right)\left(\frac{2}{5}\right)\left(\frac{1}{5}\right)\left(\frac{2}{5}\right)}{\alpha} = \frac{0.0192}{\alpha}$$

19th Feb 2019

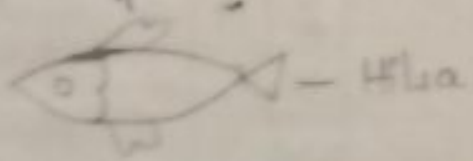
$P(y/x) = \frac{P(x/y)\,P(y)}{P(x)}$ → Likelihood / class condition; prior / Apriori → Bayes Theorem.

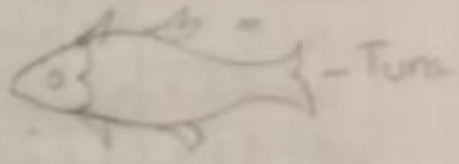
posterior / posterior → Naive Bayes Theorem

$$P(y/x_1,x_2) = \frac{P(x_2/y)\,P(x_1/y)\,P(y)}{P(x)}$$

Data : $\{\underbrace{x_{i1}, x_{i2}, \ldots x_{in}}_{i/p \text{ features}}, y_i\}$ ← class label.

Eg

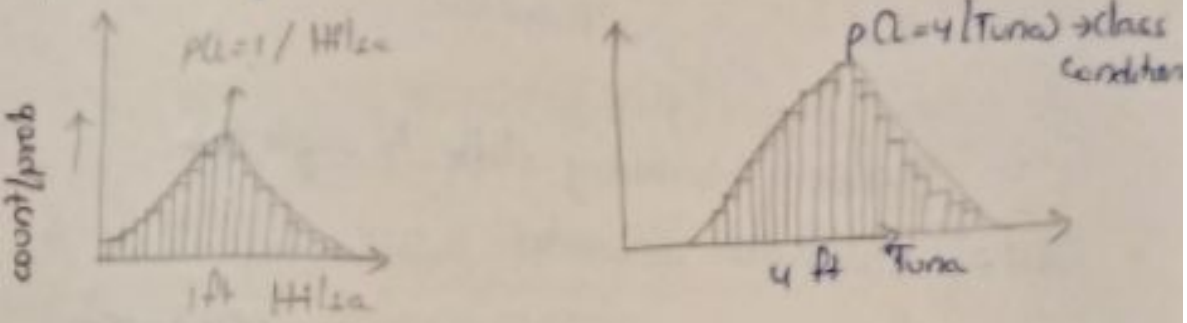


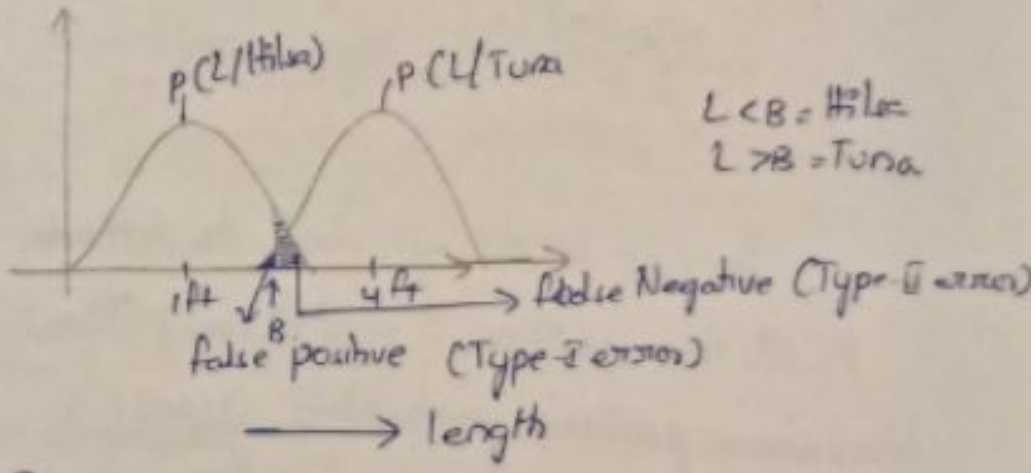


length (x1)	class
1.5	H
3	T
1	H
4.5	T

Task :- classify the fishes into 2 classes based on length
find the decision boundary - B*

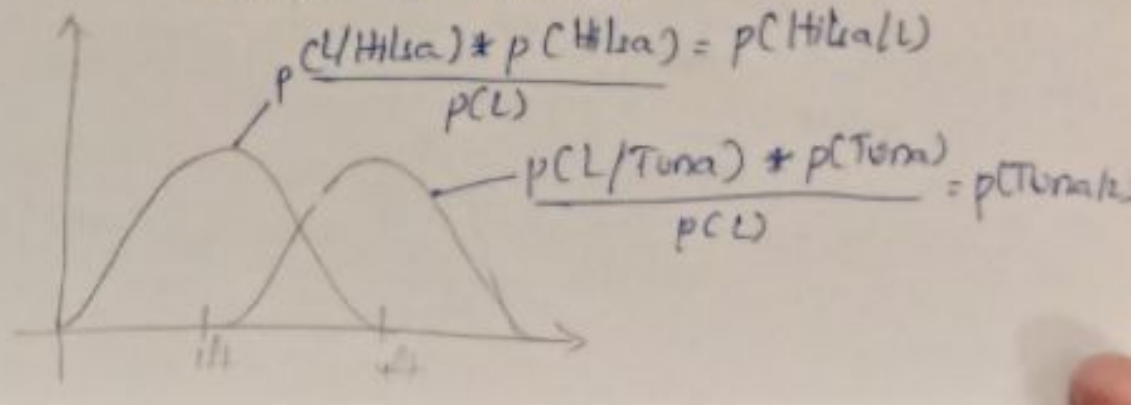
Model :- 1000 - Hilsa 1000 - Tuna
plot Histogram





Hilsa ?
Error
Type - I error : actually hilsa but classified as Tuna
Type - II error : actually Tuna but classified as Hilsa
Error : Type I + Type II

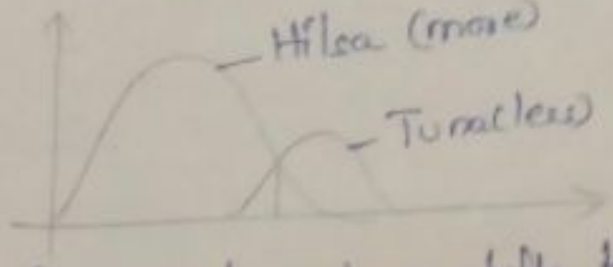
Training :- find optimal decision boundary (B*) such that error is minimum



$P(B^*/Hilsa) = P(B^*/Tuna)$

Case 1: kolkata at Monsoon:

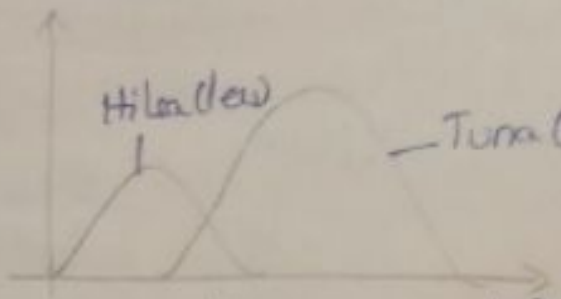
$P(H) > P(T)$



Decision boundary shifts to right side

Case 2. California in winter

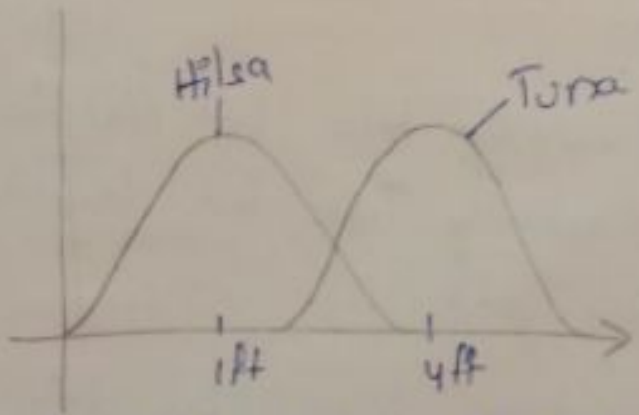
$P(H) < P(T)$



Decision boundary shifts to left side

$P(B^*/Hilsa)p(Hilsa) = p(B^*/Tuna)\,p(Tuna)$

Maximum posterior probability (MAP):



If $p(Hilsa/L) < p(Tuna/L)$

Tuna

else

Hilsa

21st Feb 19

day = {overcast, mild, normal, strong}

play = ?

$$P(y/x) = \frac{P(o/y)P(m/y)P(n/y)P(s/y)P(y)}{P(o,m,n,s)}$$

$$= \frac{\left(\frac{4}{9}\right)\left(\frac{4}{9}\right)\left(\frac{6}{9}\right)\left(\frac{3}{9}\right)\left(\frac{9}{14}\right)}{\alpha} = \frac{0.282}{\alpha}$$

$$P(n/x) = \frac{P(o/n)P(m/n)P(n/n)P(s/n)P(n)}{P(o,m,n,s)}$$

$$= \frac{\left(\frac{0}{5}\right)\left(\frac{2}{5}\right)\left(\frac{1}{5}\right)\left(\frac{3}{5}\right)\left(\frac{5}{14}\right)}{\alpha} = \frac{0}{\alpha}$$

If one of the conditional probability is '0', then entire exp becomes '0' which we don't want.

original : $P(x_i/y) = \frac{N x_i y}{N y}$

Laplace Smoothing :-

$$P(x_i/y) = \frac{N x_i y + 1}{N y + c \text{ (or } m)}$$

c = no. of classes

m = no. of features

$$\therefore P(o/N) = \frac{0+1}{5+2} = \frac{1}{7} \; ; \; P(m/N) = \frac{2+1}{5+2} = \frac{3}{7}$$

$$P(n/o) = \frac{1+1}{5+2} = \frac{2}{7} \; ; \; P(s/N) = \frac{3+1}{5+2} = \frac{4}{7}$$

$$P(No/x) = \frac{\left(\frac{1}{7}\right)\left(\frac{3}{7}\right)\left(\frac{2}{7}\right)\left(\frac{4}{7}\right)\left(\frac{5}{14}\right)}{\alpha} = \frac{0.00356}{\alpha}$$

$$P(Yes/x) = \frac{\left(\frac{5}{11}\right)\left(\frac{5}{11}\right)\left(\frac{7}{11}\right)\left(\frac{4}{11}\right)\left(\frac{5}{14}\right)}{\alpha} = \frac{0.031}{\alpha}$$

Text Classification:-

	Doc	Words	class
Train	0	India Delhi India	I
	1	India India Hyderabad	I
	2	India Mumbai	I
	3	Beijing china India	C
Test	4	India, India, India, Beijing china	?

mail - spam / not spam

Reviews - +ve / -ve

News - sports / politics / Entertainment

Bag of words Representation

Documents	0	1	2	3	4	5
0	2	1	0	0	0	0
1	2	0	1	0	0	0
2	1	0	0	1	0	0
3	1	0	0	0	1	1
4	3	0	0	0	1	1

Train → Multinomial Naive Bayes

words

Vocabulary = { India, Delhi, Hyd, mumbai, china, Beij
0 1 2 3 4 5

	0	1	2	3	4	5
0	1	1	0	0	0	0
1	1	0	1	0	0	0
2	1	0	0	1	0	0
3	1	0	0	0	1	1
4	1	0	0	0	1	1

Bernoulli Naive Bayes

$$P(3|0) = \frac{P(w_1|3)\,P(w_2|3)\,P(w_3|3)\,P(3)}{P(w_1, w_2, w_3)}$$

$$P(w|c) = \frac{N_{w,c} + 1}{N_c + |V|}$$

N_c : total no. of words in the documents of class 'c'

$N_{w,c}$: total no. of times word w appears in the documents of class 'c'

$|V|$: size of the vocabulary

$$P(\text{India}|3) = \frac{5+1}{8+6} = \frac{6}{14} = \frac{3}{7}$$

$$P(3|0) = \frac{P(\text{India}|3)^3\, P(\text{Beijing}|3)\, P(\text{china}|3)\, P(3)}{P(0)}$$

$$= \frac{\left(\frac{3}{7}\right)^3 \times \frac{1}{14} \times \frac{1}{14} \times \frac{3}{4}}{P(0)} = \frac{0.000301}{P(0)}$$

$$P(c|0) = \frac{P(\text{china}|c)\, P(\text{India}|c)\, P(\text{Beijing}|c)\, P(c)}{P(0)}$$

$$= \frac{\left(\frac{2}{9}\right)\left(\frac{2}{9}\right)\left(\frac{2}{9}\right)\left(\frac{1}{4}\right)}{P(0)} = \frac{0.0027}{P(0)}$$

22nd Feb 19

Advantages :-

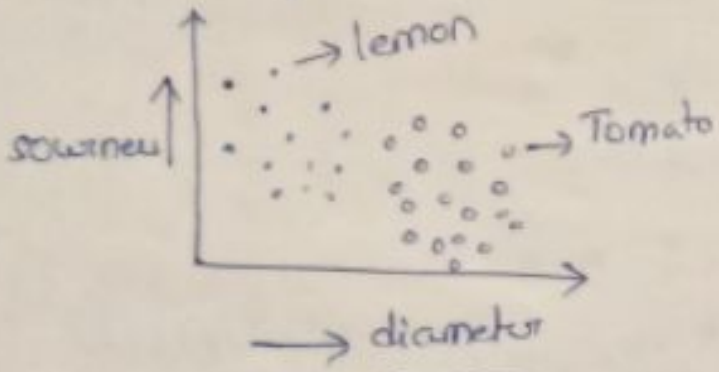
(1) Very fast, requires less storage
(2) It is robust to noise and irrelevant to features
(3) It is very good / cool in a domain with many imp features
(4) It is optimal when the independent assumptions hold

Disadvantages :-

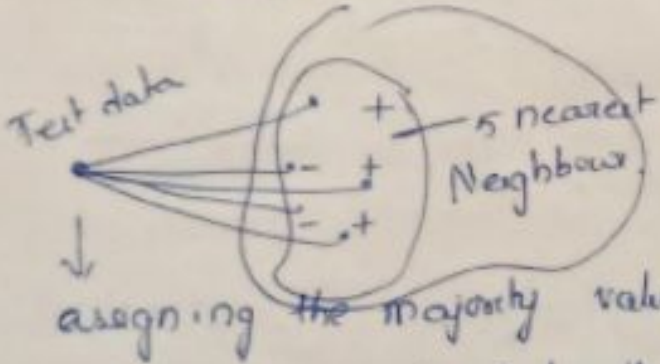
(1) Independence assumption may not hold in reality
(2) (The distribution) Because of finite dataset, we may not be able to estimate the distribution accurately.

k-Nearest Neighbour Classifier :-

- If it walks like a duck and quacks like a duck then it is probably a duck



1-NN ⇒ lemon
3-NN ⇒ lemon
5-NN ⇒ Tomato



Requires :- (a) set of levelled data
(b) distance metric (to compute distance b/w
(c) value of 'k'

To classify an unknown record :-

- compute distances to other records
- identify k nearest neighbours
- assign majority class level of the nearest neighbours to the test record

Issues :-

- The value of k (deciding)
- choice of distance metric
- computation of complex unit

* If k is too small, then the classifier is sensitive to noise
* If k is very / too large, the neighbourhood may include points from other classes
* 'k' is a hyper parameter

Rule of thumb $k = \sqrt{n}$; N is no. of datapoints.

Distance Metrics :-

- Euclidean Distance
- Manhattan Distance / City Block
- correlation
- Canberra
- Mahalnobis
- Quadratic

} Numerical

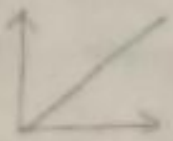
- Rank correlation
- chi-square
- value difference metric
- HDVM → Heterogeneous

} → categorical

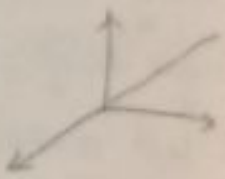
25/2/2019

Distance Metrices :- - simple attribute (single attribute)
- Vectors (multiple Attributes)

x	y



x	y	z



Simple Attribute :-

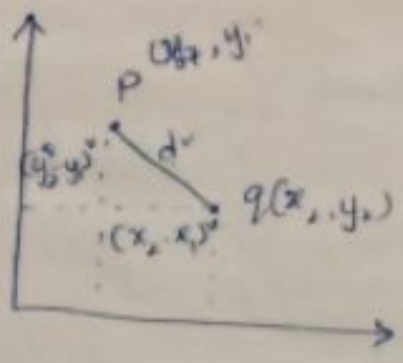
Attribute Type	Dissimilarity	similarity
Nominal	$d = \begin{cases} 0 & \text{if } p = q \\ 1 & \text{if } p \neq q \end{cases}$	$s = \begin{cases} 1 & \text{if } p = q \\ 0 & \text{if } p \neq q \end{cases}$
Ordinal	$d = \frac{\lvert x - y \rvert}{n-1}$ values are mapped to 0 to n-1	$s = 1 - d$
Interval/Ratio	$d = \lvert x - y \rvert$	$s = -d$ $s = \frac{1}{1+d}$ $s = e^{-d}$

A - 0
B - 1
C - 2
D - 3
E - 4

$d(B,E) = \frac{\lvert 4-1 \rvert}{4} = \frac{3}{4}$

Vectors :-

① Euclidean Distance



$d(p,q) = (x_2 - x_1)^2 + (y_2 - y_1)^2$

$d(p,q,r) = (x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2$

$$d(p,q) = \sqrt{\sum_{k=1}^{n} (p_k - q_k)^2}$$

n = no. of dimensions (Attributes)

p_k and q_k are both attributes

④ Minkowski Distance

Generalization of Euclidean Distance

$$d(p,q) = \left(\sum_{k=1}^{n} |p_k - q_k|^r\right)^{1/r}$$

r = parameter chosen by user

r = 2 ⇒ Euclidean Distance

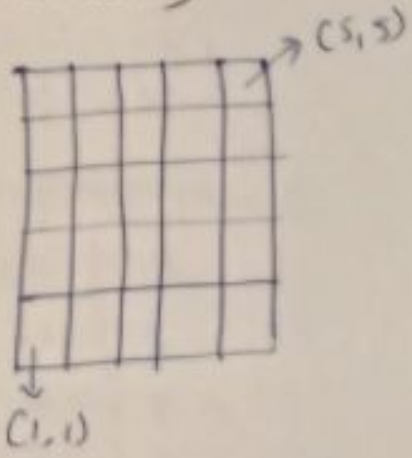
r = 1 ⇒ CityBlock / Manhattan Distance

③ CityBlock Distance (Manhattan Distance)

$$d(p,q) = \sum_{k=1}^{n} |p_k - q_k|$$

$$= |5-1| + |5-1|$$

$$\Rightarrow 4+4 = 8$$



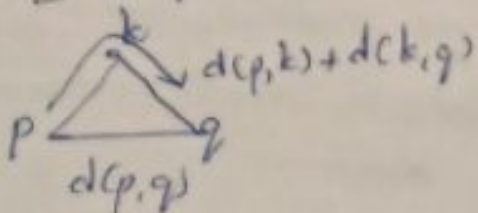
Euclidean properties:-

① $d(p,q) \geq 0$ (distance is always +ve)

② $d(p,p) = 0$ (distance at the particular point is 0)

③ $d(p,q) = d(q,p)$ (distance from p → q is equal to q → p)

④ $d(p,q) \leq d(p,k) + d(k,q)$ [Triangular Inequality] property



Issues with Euclidean Distance :-

① scale Effect

Income (1000)	House size (100 sq.ft)
75	19
52	17

d = 23.07

Income Rupees	House size (100 sq.ft)
75000	19
52000	17

d = 23000

Income 1000	House size (sq.ft)
75	1900
52	1700

d = 201.

* Different features may have different measurement scales.
* Features with larger value will have greater impact on Distance - It may bias the performance of classifier.

② sparsity in the High Dimensional Data

1 1 1 1 1 1 1 1 1 1 1 0 } d = 1.4142 → more similar
0 1 1 1 1 1 1 1 1 1 1 1 1

1 0 0 0 0 0 0 0 0 0 0 0 } d = 1.4142 → more similar
0 0 0 0 0 0 0 0 0 0 0 0 1

③ Collinearity Issue :-

X_1	X_2	X_3
2	1	1
4	6	3
6	8	5

If variables are colinear then co-related variable will have more impact on distance measure than independent variables

Scale effects - Handling :-

Feature Scaling :- ① Normalization - Minmax Scaling
② Standardization

① Normalization :- Data is scaled in the range of 0

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

- problem is Normalization suppresses the outliers

② Standardization :-

$$z = \frac{X - \mu}{\sigma} = \frac{X - \bar{x}}{S}$$

$\bar{x}$ = mean of the Data
S = standard deviation of Data

Eg:-

Marks	Z-score	Normalisation
90	-0.0823	0.08
95	0.7438	0.74
92	0.2479	0.24
93	0.41322	0.41
98	1.2397	1.24
95	0.7438	0.74
98	1.2397	1.24
90	-0.0826	-0.08

(c) $\bar{x} = 90.5$

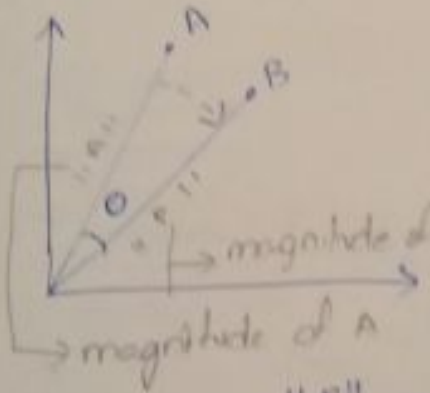
$S = 6.05$

$Z \sim N(0,1)$ ← Normalisation.

↳ Mean

Sparsity in High Dimensional Data:— Handling :

(1) Cosine Similarity :-



$\cos 0^\circ = 1$ (most similar)

$\cos 90^\circ = 0$

$\cos 180^\circ = -1$ (least similar)

$$\cos\theta = \frac{\|P\|}{\|A\|} = \frac{\vec{A}\cdot\vec{B}}{\|\vec{A}\|\cdot\|\vec{B}\|}$$

$$\|P\| = \frac{\vec{A}\cdot\vec{B}}{\|\vec{B}\|} = \frac{\sum_{i=1}^{n} A_i \cdot B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}} = \frac{5}{\sqrt{42}\sqrt{6}}$$

$$= \frac{5}{15.8745} = 0.3149$$

$\vec{A} = [3, 2, 0, 5, 0, 0, 0, 2, 0, 0]$

$\vec{B} = [1, 0, 0, 0, 0, 0, 0, 1, 0, 2]$

26-2-19
(5) Mahalanobis Distance :-

$$d(p,q) = \sqrt{(p-q)^T \Sigma^{-1} (p-q)}$$

Σ = covariance matrix of Data

$$p = [\,]_{n \times 1}$$

$$q = [\,]_{n \times 1}$$

$p - q = n \times 1$

$[p-q]^T = 1 \times n$

$\Sigma^{-1} = n \times n$

$$= \sqrt{\underset{1\times n}{(p-q)^T} \; \underset{n\times n}{\Sigma^{-1}} \; \underset{n\times 1}{(p-q)}}$$

$$\Sigma_{j,k} = \frac{1}{n-1} \sum_{i=1}^{n} (x_{ij} - \bar{x})(x_{ik} - \bar{x}_k)$$

$$\boxed{d(p,q) = \sqrt{(p-q)^T (p-q)}}$$

Σ = identity Matrix, MD = ED

Binary Attributes :-

p = 0 1 0 0 1 1 0

q = 1 0 1 0 1 0 0

Jaikcard Coefficient :

	1	0
1	1	2
0	2	2

M_{00} = # of p = 0 q = 0

M_{01} = # of p = 0 q = 1

M_{10} = # of p = 1 q = 0

M_{11} = # of p = 1 q = 1

J = measure of similarity in presence $= \frac{M_{11}}{M_{01} + M_{10} + M_{11}} \Rightarrow \frac{2}{2+2+1} = \frac{1}{5}$

0 - absent

1 - present

so M_{00} doesn't have importance.

Asymetoric Binary Attribute ⇒ gives more importance to presence (1)

Symmetoric Binary Attribute ⇒ Both outcomes are equally important

Simple Binary Coefficient = measure of dissimilarity -

$$= \frac{M_{01} + M_{10}}{M_{00} + M_{01} + M_{10} + M_{11}}$$

$$= \frac{4}{7}$$

Nominal Attribute :-

simple matching coefficient = $\frac{mm}{n}$

n = total no of attributes

mm = # of mismatches

Text Classification

	0	1	2	3	4	5	class	cosine Distance
0	2	1	0	0	0	0	I	0.81
1	2	0	1	0	0	0	I	0.81
2	1	0	0	1	0	0	I	0.64
3	1	0	0	0	1	1	C	0.87
4	3	0	0	0	1	1	?	

words

S.NO	Name	Give Birth	can Fly?	lives in water	Have legs	class	Distance
1	Human	y	n	n	n	m	2/4
2	salmon	n	n	y	n	nm	1/4
3	python	n	n	n	n	nm	2/4
4	whale	y	n	y	n	m	0
5	Frog	n	n	y	y	nm	2/4
6	komodo	n	n	n	y	nm	3/4
7	Bat	y	y	n	y	m	3/4
8	pigeon	n	y	n	y	nm	1
9	Cat	y	n	n	y	m	2/4
10	Dolphin	y	n	y	n	m	0
11	—	y	n	y	n	? ↑ m	

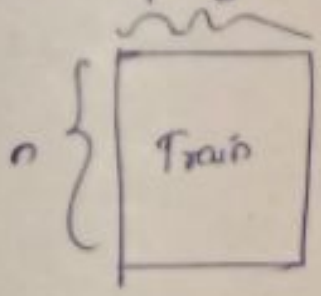
$k = \sqrt{10} = 3$ 2m 1nm

2-mammals are there (majority)
so mammal

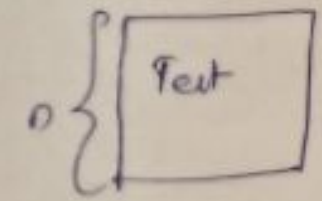
Performance of knn

① Time Complexity - Expensive

To determine the Nearest Neighbour on test data we must compute distances to all training data. It is expensive.



$O(nxd + n\log n)$



$O(n(nxd + n\log n))$

$O(n^2d)$

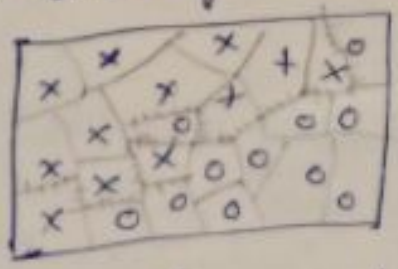
- Remove redundant data - condensation
- pre sort data into fast data structures
  - → kd tree
- Compute only approximate distance (LSH)

② Space complexity - must store all training data

Training - Condensation

Condensation

Decision bounding in kNN -



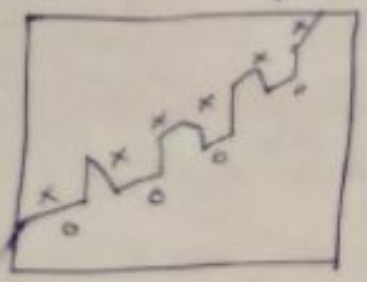
A voronoi diagram devides the space into cells, such that each cell contain one sample and every location within cell is closer to that sample than any other sample

The Decision Boundary seperates the class Region
Varnol NN Decision Rule

knowledge of the Decision boundary is sufficient
classify new Data. Retain only those points necessary
generate an identical boundary. This is called Condensa

Two types:-

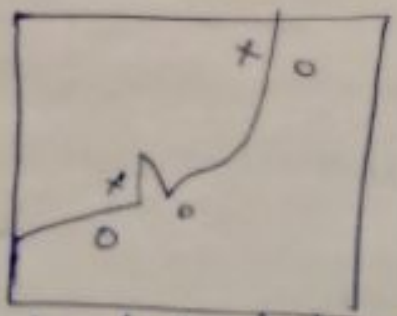
① Decision Boundary Condensation



This condensation is subset onto to Nearest Neighbour
Decision boundary

- A subset whose nearest neighbour decision boundary
is identical to the boundary of the entire training set

Minimum consistent set condensation



The smallest subset of training data that correctly
classifies all the original data