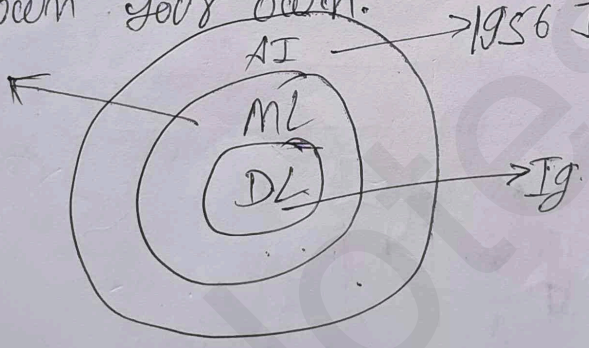
IML

Artificial Intelligence AI is man made thinking power. It is a branch of computer science by which we can create intelligent machine which can behav, think like human and are able to make decision. AI exists when a machine can have human based skills, like learning, reasoning, problem solving. It is a process of building intelligence machine from vast volume of data. AI uses complex Algo and methods to build machine that can make decision own your own.



1956 John Mc Carthy

1959 Arthur Samuel

Ig

vast volume of data →

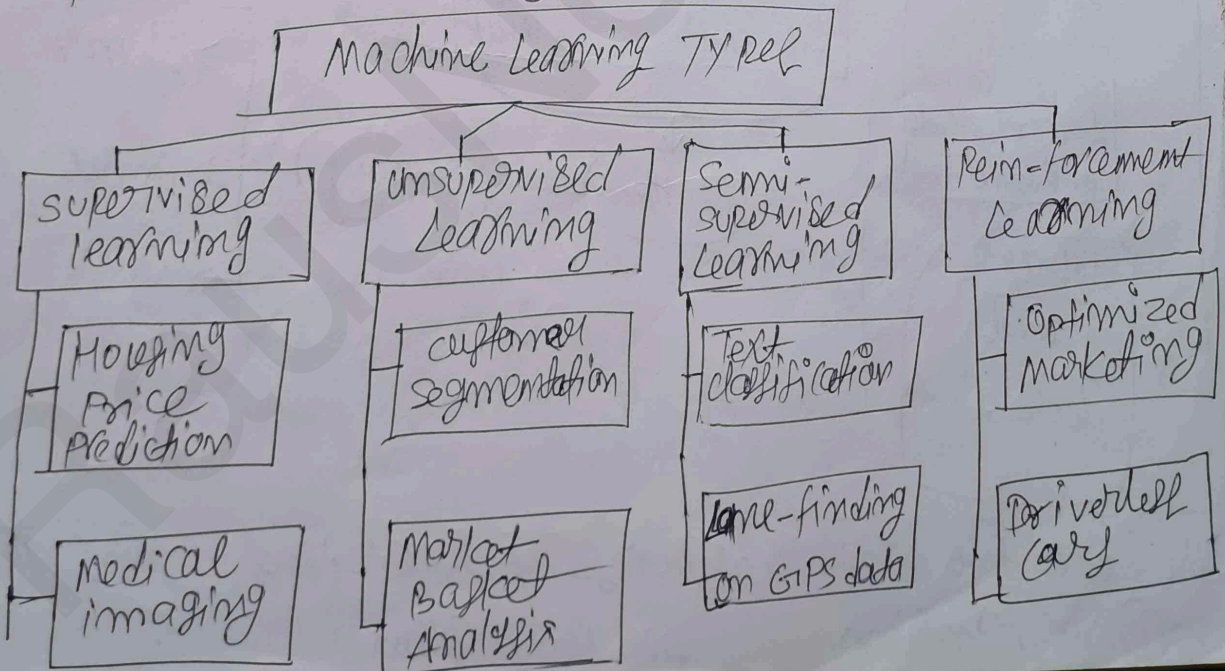
during world war second, world war the first computer was developed to breaks the government communi-- in 1950 Turnel published a paper in

(P25) Types of ML

Machine learning: Machine learning is a subset of Artificial intelligence (AI) that enables systems to learn from data, identify patterns and make decisions with minimaal human intervention. Machine learning is used today for a wide range of commercial purposes, including suggesting products to consumers based on their past purchases, predicting stock market and translating text from one language to another. Machine learning contains a set of algorithms that work on a huge amount of data. Data is fed to these algorithms to train them, and on the basis of training, they build the model & perform a specific task.

Machine learning is divided into mainly four types

(i) Supervised Machine Learning
(ii) Unsupervised Machine learning
(iii) Semi-supervised Machine learning
(iv) Reinforcement Learning



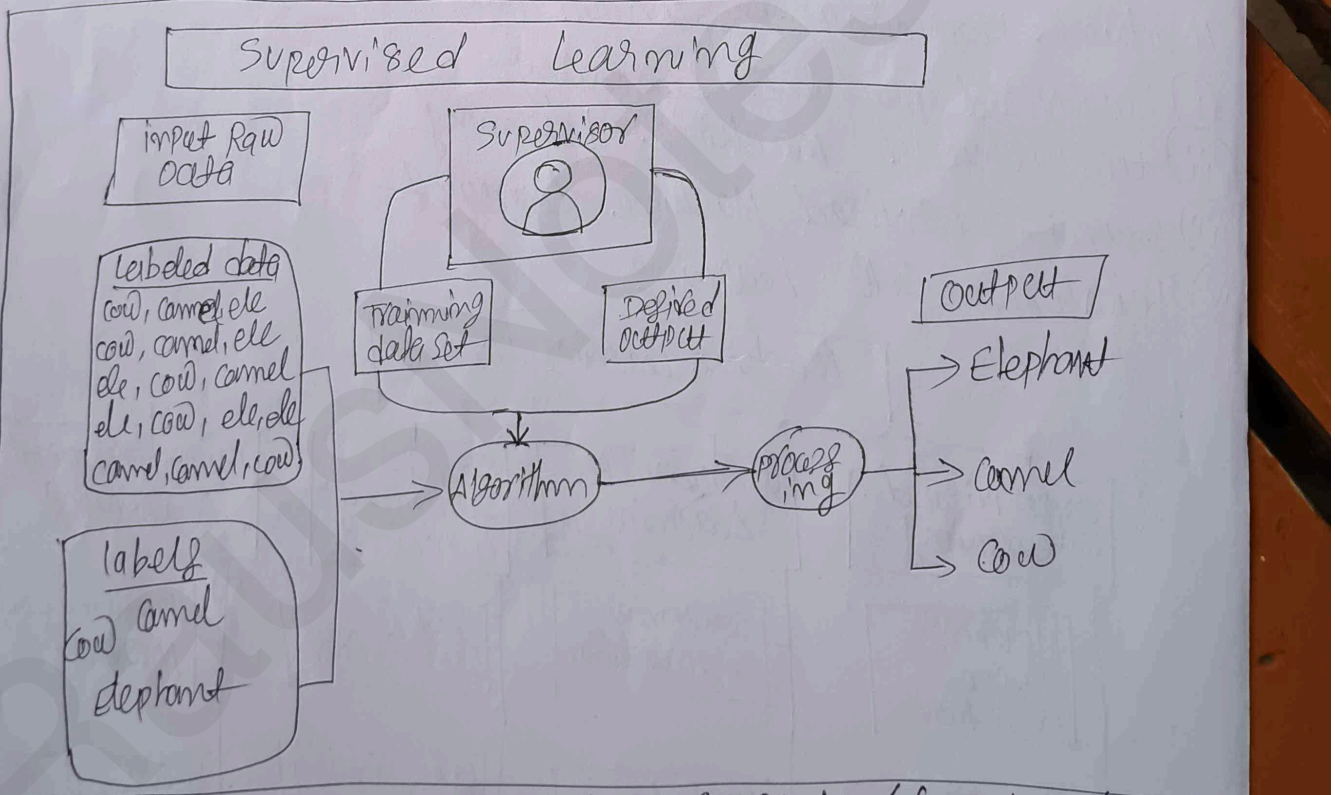
① Supervised Machine Learning: Supervised machine learning is based on supervision. It means in the supervised learning technique, we train the machines using the "labelled" dataset, and based on the training, the machine predicts the output.

labeled data: Data that includes both input features and their output labels

Ex ⇒ input: features of a house (size, location, no of rooms)
output: price of the house

Supervised learning maps the input variable (x) with the output variable (y).

Application: (i) Spam detection (ii) Recommendation system (iii) Fraud detection (iv) Stock market prediction (v) weather forecasting (vi) sports analytics.



There are two main categories of supervised learning that are
(i) classification (ii) Regression

(i) classification: classification algorithm are used to slove the classification problems in which the output variable is categorical, such as "yes" or "No", male or female, Red or Blue, etc. The classification algorithm predict the categories present in the dataset. Ex → spam detection

Here are some classification algorithms.

- Logistic Regression
- Random forest
- Decision Tree

(ii) Regression: Regression predicting continuous target variables, which represent numerical values. Regression algorithms are used to slove regression problems in which there is a linear relationship b/w input and output variables. These are used to predict continuous output variables, such as market trends, weather prediction. Here are some regression algorithms.

- Linear Regression
- polynomial Regression
- Random forest
- Decision Tree

Advantages of Supervised Machine learning

(i) It have high accuracy as they are trained on labeled data.

(ii) It can often be used in pre-trained models which save time and resources when developing new models from scratch.

(iii) In supervised learning we can have exact idea about the classes of objects.

(iv) The process of decision making in supervised learning models is often interpretable.

Disadvantages: (i) These algorithms are not able to slove complex tasks.

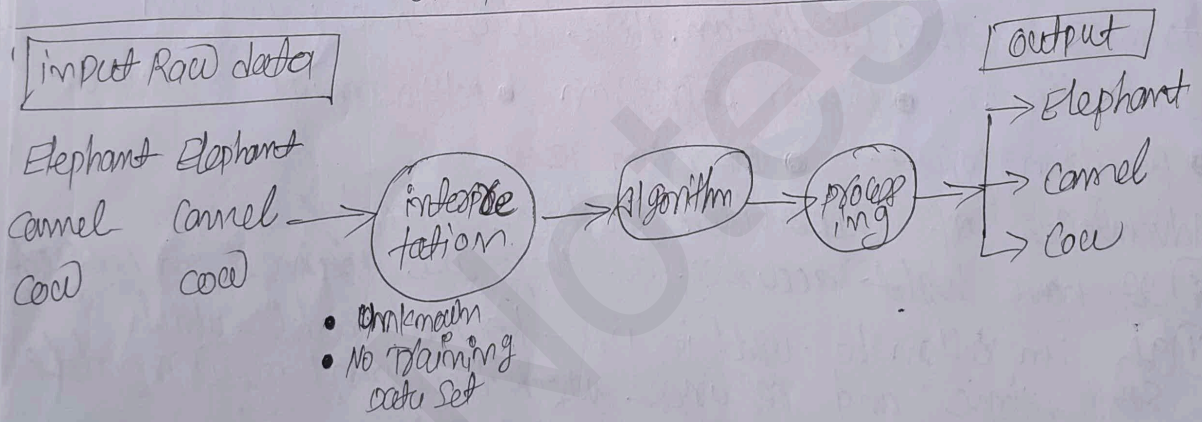
(ii) It may predict the wrong output if the test data is different from the training data.

(iii) It can be time-consuming and costly as it relies on labeled data only.

(iv) It required lots of computational time to train them.

② Unsupervised Machine Learning: Unsupervised learning is a type of machine learning technique in which an algorithm discovers patterns and relationships using unlabled data. Unlike supervised learning. It is different from the supervised learning technique; as its name suggests, there is no need for supervision. It means in unsupervised machine learning, the machined is trained using the unlabeled dataset and the machine predicts the output without any supervision. The primary goal of unsupervised learning is often to discover hidden patterns, similarities, or clusters within the data, which can them be used for various purpose.

Unsupervised Learning



Application of unsupervised Learning

(i) clustering (ii) Anomaly detection (iii) Recommendation system
(iv) Image and video compression (v) Data preprocessing

* There are two main categories of unsupervised learning that are mentioned below.

(i) clustering (ii) Association

(i) clustering: clustering is the process of grouping data points into clusters based on their similarity. This technique is useful for identifying patterns and relationship

in data without the need for labeled examples. The clustering technique is used when we want to find the inherent groups from the data. It is a way to group the objects into a cluster such that the objects with the most similarities remain in one group and have fewer or no similarities with the objects of other groups. Ex ⇒ Grouping the customers by their purchasing behaviour.

Here some clustering algorithm.

- k-means clustering algorithm
- DBSCAN Algorithm
- Mean-shift algorithm

- Association: Association rule learning is an unsupervised learning technique, which finds interesting relations among variables within a large dataset. The main aim of this learning algorithm is to find the dependency of one data item on another data item and map those variables accordingly so that it can generate maximum profit. This algorithm is mainly applied in market Basket analysis, web usage mining, continuous production. Some popular algorithms of Association rule learning are
  - Apriori Algorithm
  - Eclat, FP-growth algorithm

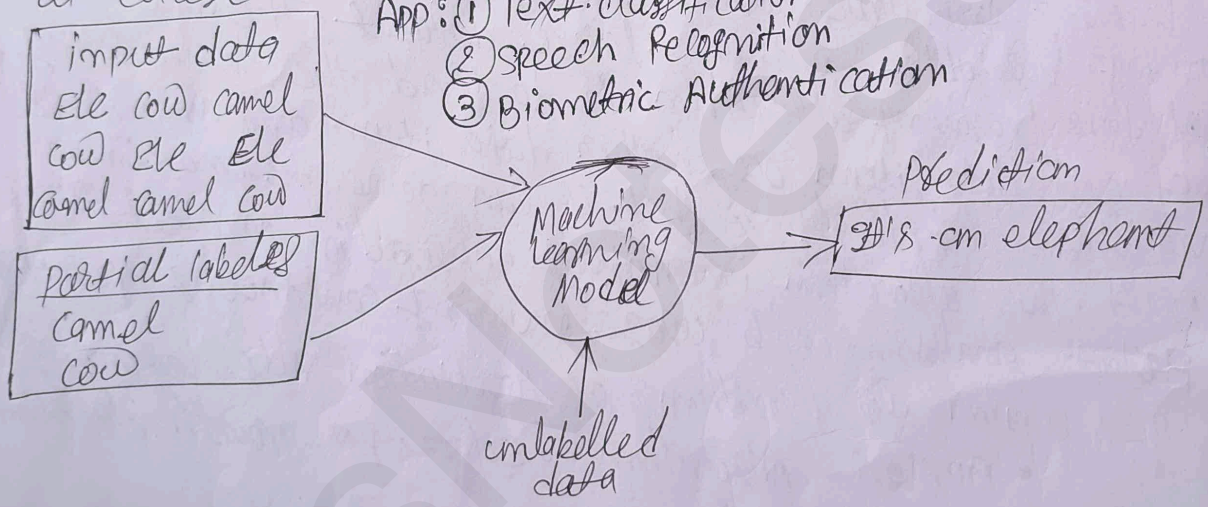
* Advantage of unsupervised Machine learning
  (i) helps to discover hidden patterns and various relationships b/w the data.
  (ii) used for tasks such as customer segmentation, anomaly detection, and data exploration.
  (iii) it does not require labeled data and reduces the effort of data labeling.

Disadvantages of unsupervised machine learning
- without using labels, it may be difficult to predict the

quality of the model's output.
- buffer interpretability may not be clear and may not have meaningful interpretations.

(3) Semi-supervised learning: Semi-supervised learning is a type of machine learning algorithm that lies b/w supervised and unsupervised machine learning. It represents the intermediate ground b/w supervised (with labelled data) and unsupervised learning (without labelled data) algorithms and uses the combination of labelled and unlabeled datasets during the training period.

Example: Student has to revise himself after analyzing the same concept under the guidance of an instructor at college.



App: (1) Text classification
(2) Speech Recognition
(3) Biometric Authentication

Advance: (i) It is simple and easy to understand the algorithm
(ii) It is highly efficient.
(iii) It is used to solve drawbacks of supervised and unsupervised algorithms.

Disadvantages:
(i) Iterations results may not be stable.
(ii) We cannot apply these algorithms to network-level data
(iii) Accuracy is low.

④ Reinforcement learning: Reinforcement learning works on a feedback-based process, in which an AI agent automatically explore its surrounding by hitting & trail, acting, learning from experience and improving its performance. Agent gets rewarded for each good action and get punished for each bad action. In this technique, the model keeps on increasing its performance using reward feedback to learn the behaviors or patterns.

Ex ⇒ Google self Driving car,

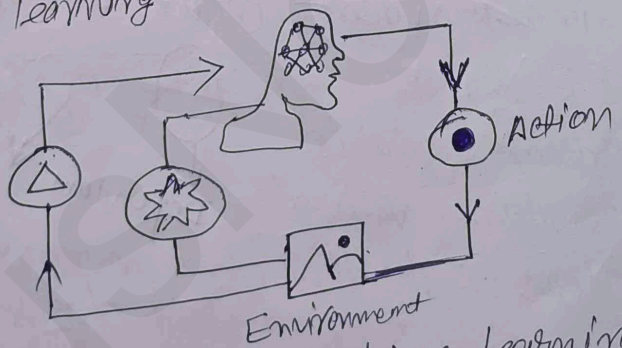
Applications of reinforcement Machine learning

(i) Game playing (ii) Robotics (iii) ~~Automation vehicle~~ Autonomous vehicle (iv) Recommendation system (v) Healthcare (vi) Natural language processing (vii) Agriculture

*Here are some of most common reinforcement learning algorithm-

(i) Q-learning (ii) State-Action-Reward-State-Action

(iii) Deep-Q-Learning



Advantages of Reinforcement Machine learning:

(i) It has autonomous decision-making that is well-suited for tasks and that can learn to make a sequence of decisions, like robotics and game-playing

(ii) It is used to solve complex problems that cannot

be sloved by conventional technique.

Disadvantages of Reinforcement Machine learning

(i) Training Reinforcement learning agents can be computationally expensive and time consuming.

(ii) It is not preferable to sloving simple problems

(iv) It needs a lot of data. a lot of computations that makes it costly.

Unit - 2

* Simple linear regression: Simple Linear Regression is a statistical technique used to model the relationship b/w a dependent variable y and a single independent variable x. It assumes a linar relationship b/w these two variables, represented by the equation of a straight line. It is also a type of supervised machine learning algorithms that learns from the labelled datasets. It predicts the continuous output variables based on the independent input variable.

Equation of linear regression $\Rightarrow y = ax + b$

x → independent variable

y → dependent variable (y: Dependant, x: Independant)

a → is the slope of the line (how much y changes for a unit change in x).

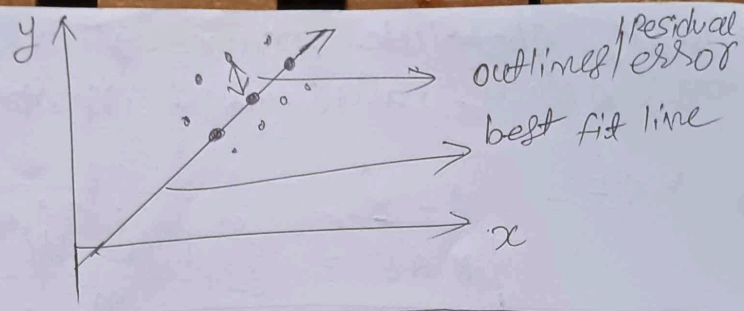
b → is the intercept

$$y = ax + b$$

where

$$a = \frac{n(\Sigma xy) - \Sigma x \cdot \Sigma y}{n(\Sigma x^2) - (\Sigma x)^2}$$

$$b = \frac{\Sigma y - a(\Sigma x)}{n}$$



Residual error = Actual − Predicted

Q Expenditure of business (in thousands) for every year is shown in table below.

x (year)	1	2	3	4	5
y (Expenditure)	12	19	24	37	45

Soln:

Sr. N	x	y	xy	x^2
1	1	12	12	1
2	2	19	38	4
3	3	29	87	9
4	4	37	148	16
5	5	45	225	25

$\Sigma x = 15, \Sigma y = 142, \Sigma xy = 510, \Sigma x^2 = 55$

$y = ax + b$

$b = \frac{\Sigma y - a\Sigma x}{n}$

$a = \frac{n\Sigma xy - \Sigma x \Sigma y}{n\Sigma x^2 - (\Sigma x)^2} = \frac{5 \times 510 - 15 \times 142}{5 \times 55 - (15)^2} = 8.4$

$b = \frac{142 - 8.4 \times 15}{5}$ $\quad b = 3.2$

Equation of line $\Rightarrow y = 8.4x + 3.2$

for $x = 6 \Rightarrow y = 8.4 \times 6 + 3.2$

$y = 53.6$ thousands

(Q) Write perform comparison b/w simple linear reg and polynomial regression with example.

$\Sigma(y-\bar{y})$

Evaluate Metrics for linear regression

(i) Mean Absolute Error(MAE): Mean absolute Error is the Average difference between actual value and predicted value.

$$MAE = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n}$$

Residual error = $y_i - \hat{y}_i$

X (items)	Y (Sales)
11	80
12	90
13	100
14	110
15	120

x	y_i	$\hat{y}_i$
16	80	75
17	75	85

$$MAE = \frac{|80-75| + |75-85|}{2} = \frac{15}{2} = 7.5$$

(ii) Mean square Error(MSE): Mean ~~absolute~~ square Error is the square of average difference b/w actual value and predicted value.

$$MSE = \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n}$$

$$MSE = \frac{(80-75)^2 + (75-85)^2}{2} \Rightarrow \frac{25+100}{2} = \frac{125}{2} = 62.5$$

(CRMSE)

(iii) Root Mean Square Error = $\sqrt{MSE} \Rightarrow \sqrt{62.5} \Rightarrow 7.90$

Root Mean square Error is the root of ~~square~~ Mean Square Error.

(iv) Relative MSE : (RelMSE)

$$\text{RelMSE} = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}$$

$\bar{y}$ → mean

x_i	sale (y_i)
11	80
12	90
13	100
14	110
15	120

The avg of y is: $\frac{80+90+110+110+120}{5}$

$\Rightarrow \frac{500}{5} = 100$ $\boxed{\bar{y} = 100}$

Item	AV	PV
16	80	75
17	75	85

$$\text{Rel MSE} = \frac{(80-75)^2 + (75-85)^2}{(80-100)^2 + (75-100)^2}$$

$$\text{RelMSE} = \frac{125}{1025} = 0.1219$$

* Coefficient of variation $CV = \frac{\text{RelMSE}}{\bar{y}}$

$$\boxed{CV = \frac{RMSE}{\bar{y}}} \Rightarrow CV = \frac{\sqrt{62.5}}{100} \quad \boxed{CV = 0.08}$$

(v) Coefficient of Determination (R^2) :-

X	y	xy	x^2
1	3	3	1
2	4	8	4
3	2	6	9
4	4	16	16
5	5	25	25
$\Sigma x = 15$	$\Sigma y = 18$	$\Sigma xy = 58$	$\Sigma x^2 = 55$

$$a = \frac{n\Sigma xy - \Sigma x \Sigma y}{n\Sigma x^2 - (\Sigma x)^2} \Rightarrow \frac{5\times58 - 15\times18}{5\times55 - 15\times15}$$

$$a = \frac{290 - 270}{275 - 225} \qquad a = \frac{20}{50} \quad \boxed{a = 0.4}$$

$$\boxed{b = \frac{\Sigma y - a\Sigma x}{n}} \qquad b = \frac{18 - (0.4)(15)}{5}$$

$$b = \frac{18-6}{5} \Rightarrow b \Rightarrow \frac{12}{5}$$

$$\boxed{b = 2.4}$$

$\boxed{y = ax + b}$

$\boxed{y = 0.4x + 2.4}$ (OD)

Let's predict value of y for $x = 1, 2, 3, 4, 5$ & calculate coefficient of determination.

X	y	$\hat{y}$	$(y_p - \bar{y})^2$	$(y_p - \bar{y})^2$	$(\hat{y} - \bar{y})$	$(y - \bar{y})^2$
1	3	2.8	0.04			
2	4	3.2	0.04			
3	2	3.6				
4	4	4.0				
5	5	4.4				

for $x_1 = 0.4(x) + 2.4 \Rightarrow x_1 = 0.4(1) + 2.4 \Rightarrow x_1 = 0.4 + 2.4 = 2.8$

for $x_2 = 0.4(2) + 2.4 \Rightarrow 0.8 + 2.4 \Rightarrow 3.2$

and so on till x5.

$$R^2 = \frac{\Sigma (y_p - \bar{y})^2}{\Sigma (y - \bar{y})^2}$$

$$\bar{y} = \frac{3 + 4 + 2 + 4 + 5}{5}$$

$$\bar{y} = \frac{18}{5}$$

$$\bar{y} = 3.6$$

* Multiple linear regression: Linear reg is a statistical approach for modeling relationship b/w dependent y and one independent variable x but multiple linar regression accepts two model relationship b/w two or more features to fit a linear equation to predict one dependent variable, regression model learns a function from the dataset and use it to predict y values for unknown x_1, x_2, x_3 by adding more predictors to simple linear regression the technique help to understand better how the predictors effects the outcome variable as a whole. for example prediction of CO2 emmition based on no of size and no of cinders in a car. multiple linear reg. tries to fit a reg line to multi dimensiony of data point.

$$y = a_1x_1 + a_2x_2 + a_3x_3 + \cdots - a_nx_n + b$$

$a_1, a_2 \ldots$ → slope
b → intersept
x → y →

Agent Salary

X	experiment	Salary y
35	1	20
25	1	10
40	2	35
50	4	?

$y =$

$$y = a_1x_1 + a_2x_2 + b$$

$$a_1 = \frac{(\Sigma x_2^2)(\Sigma x_1 y) - (\Sigma x_1 x_2)(\Sigma x_2 y)}{(\Sigma x_1^2)(\Sigma x_2^2) - (\Sigma x_1 x_2)^2}$$

$$a_2 = \frac{(\Sigma x_1^2)(\Sigma x_2 y) - \Sigma(x_1 x_2)(\Sigma(x_1 y))}{(\Sigma x_1^2)(\Sigma x_2^2) - (\Sigma x_1 x_2)^2}$$

$b = \bar{y} - a_1\bar{x}_1 - a_2\bar{x}_2$

x_1	x_2	y	x_1^2	x_2^2	x_1x_2	x_1y	x_2y
35	1	20	1225	1	35	700	20
25	1	10	625	1	25	250	10
40	2	35	1600	4	80	1400	70
50	4	?					

$\Sigma x_1 = 100$
$\Sigma x_2 = 4$
$\Sigma y = 65$

$\Sigma x_1^2 =$
$\Sigma x_2^2 = 6$
$\Sigma x_1 x_2 = 140$

$\Sigma x_1 y =$
$\Sigma x_2 y = 100$

$b = \bar{y} - a_1\bar{x}_1 - a_2\bar{x}_2$

$a_1 =$

$y = a_1x_1 + a_2x_2 + b$

$y = 0.09x_1 + 14.5x_2 + 4.5$

$y = 0.09x1(50) + 14.5(4) + 4.5$

$y = 67\ K$

* polynomial regression: polynomial is a regression algorithm that models a the relationship b/w a dependent & independent variable as nth degree. It is known as of a special case of MLR.

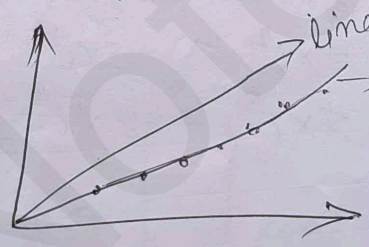
$$y = a_1 x_1^1 + a_2 x_1^2 + a_3 x_1^3 + \dots + a_n x^n + FP$$

polynomial function

we add some polynomial terms to the multiple linear regression equation to convert it into polynomial regression (i) we required it to increase accuracy (ii) Data set used of non-linear nature.

Q Why we need polynomial regression.

If we apply a linear model on the linear dataset it provides good result, but if we apply the same model without any modification on a non-linear dataset, it will produce drastic output.



in this figure data-points are arranged in non linear fashion. we need a polynomial regression.

in above we taken a data set which is arranged in non linear. if we convert it with linear model it hardly covers any data point. A curve is suitable to cover most of the datapoints i.e. polynomial model. If dataset is polynomial

Regression instead of simple LR. Simple & MLR are also polynomial regression with a single degree (x^1). But PR is linear equation with n^{th} degree (x^n). If we add a degree to our LR then it will converted to PR.

Use: When there is a non-linear co-relation b/w 2 variables - it can easily fit a vast range of curvatures.

$$\boxed{Y = b + \sum_{i=1}^{n} a_i x^i + fp}$$

$$a = X^{-1} B$$

$$X = \begin{bmatrix} n & \Sigma x_i & \Sigma x_i^2 \\ \Sigma x_i & \Sigma x_i^2 & \Sigma x_i^3 \\ \Sigma x_i^2 & \Sigma x_i^3 & \Sigma x_i^4 \end{bmatrix}^{-1} \begin{bmatrix} \Sigma y_i \\ \Sigma (x_i y_i) \\ \Sigma (x_i^2 \cdot y_i) \end{bmatrix} \begin{matrix} \rightarrow b \\ \rightarrow a_1 \\ \rightarrow a_2 \end{matrix}$$

Ex:

x	y
1	1
2	4
3	9
4	15

$$y = a_1 x_1 + a_2 x_1^2 + b$$

$$a = X^{-1} B \qquad a \approx \begin{bmatrix} b \\ a_1 \\ a_2 \end{bmatrix}$$

$$a = (X) \begin{bmatrix} n & \Sigma x_i & \Sigma x_i^2 \\ \Sigma x_i & \Sigma x_i^2 & \Sigma x_i^3 \\ \Sigma x_i^2 & \Sigma x_i^3 & \Sigma x_i^4 \end{bmatrix}^{-1} \begin{bmatrix} \Sigma y_i \\ \Sigma (x_i y_i) \\ \Sigma (x_i^2 y_i) \end{bmatrix}$$

x	y	$x_i y_i$	x_i^2	x_i^3	x_i^4	$x_i^2 y_i$
1	1	1	1	1	1	1
2	4	8	4	8	16	16
3	9	27	9	27	81	81
4	15	60	16	64	256	240

Any =
$\begin{bmatrix} -0.75 \\ 0.95 \\ 2.75 \end{bmatrix}$

$$\begin{bmatrix} b \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 4 & 10 & 30 \\ 10 & 30 & 100 \\ 30 & 100 & 354 \end{bmatrix}^{-1} \begin{bmatrix} 2 \\ 96 \\ 338 \end{bmatrix}$$

(10, 30, 100)

$$A^{-1} = \frac{1}{|A|} (\text{adj } A)$$

$$(\text{adj } A) = \begin{matrix} 30 & 100 & 10 & 30 \\ 100 & 354 & 30 & 100 \\ 10 & 30 & 4 & 10 \\ 30 & 100 & 10 & 30 \end{matrix}$$

$$\text{adj} = \begin{bmatrix} 620 & -540 & 100 \\ -540 & 516 & -100 \\ 100 & -100 & 20 \end{bmatrix}$$

$$\begin{bmatrix} b \\ a_1 \\ a_2 \end{bmatrix} \quad A = \begin{bmatrix} 4 & 10 & 30 \\ 10 & 30 & 100 \\ 30 & 100 & 354 \end{bmatrix}$$

$$\text{adj } A = \begin{bmatrix} 620 & -540 & 100 \\ -540 & 516 & -100 \\ 100 & -100 & 20 \end{bmatrix}$$

$$|A| = 248.0 + (-5400) + 3000$$

$$|A| = 80$$

$$A^{-1} = \frac{1}{80} \begin{bmatrix} 620 & -540 & 100 \\ -540 & 516 & -100 \\ 100 & -100 & 20 \end{bmatrix}$$

$$A^{-1} = \begin{bmatrix} \frac{620}{80} & \frac{-540}{80} & \frac{100}{80} \\ \frac{-540}{80} & \frac{516}{80} & \frac{-100}{80} \\ \frac{100}{80} & \frac{-100}{80} & \frac{20}{80} \end{bmatrix}$$

$$A^{-1} = \begin{bmatrix} \frac{31}{4} & \frac{-27}{4} & \frac{5}{4} \\ \frac{-27}{4} & \frac{129}{20} & \frac{-5}{4} \\ \frac{5}{4} & \frac{-5}{4} & \frac{1}{4} \end{bmatrix}$$

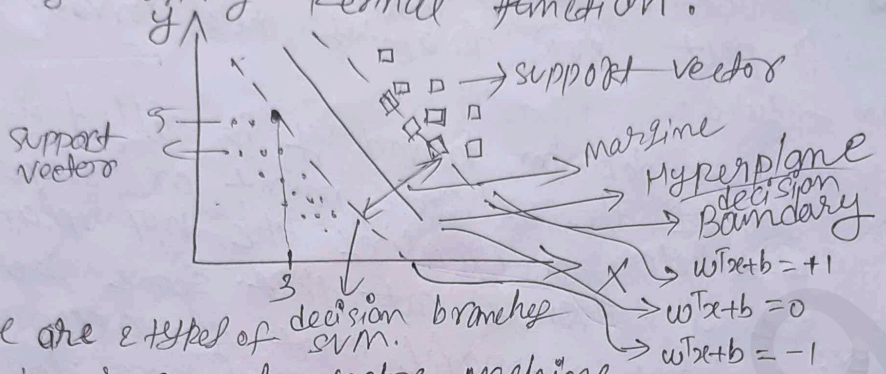
$$A^{-1} = \begin{bmatrix} 7.75 & -6.075 & 1.25 \\ -6.75 & 6.45 & -1.25 \\ 1.25 & -1.25 & 0.25 \end{bmatrix}_{3\times3} \times \begin{bmatrix} 2 \\ 96 \\ 338 \end{bmatrix}_{1\times1}$$

$$\begin{bmatrix} b \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} -210 \\ 183.2 \\ -33 \end{bmatrix}$$

$$y = 183.2x_1 + (-33)x_1^2 + (-210)$$

$$y = 183.2x_1 - 33x_1^2 - 210$$

* Support vector machine (SVM): SVM is a supervised machine algorithm used for both classification and regression. It can solve both linear and nonlinear problems using kernal function.



There are 2 types of SVM.
(i) Linear Support vector Machine
(ii) Non-Linear Support vector Machine

(i) Linear SVM : Linear SVM is used for linearly seprable data which means if a dataset can be classified into two classes by using a single straight line then such data is linearly seprable data.

(ii) Non Linear SVM : Non-SVM used for non linear seprably data, which means if a data set cannot be classified by using a straight line such data is termed as non-linear data, SVM works by finding a hyper plane in a high dimentional space that best seperates data into diff classes. It aims to find a function that approximate the relationship b/w input variables and continuous

target variable by minimizing prediction error.

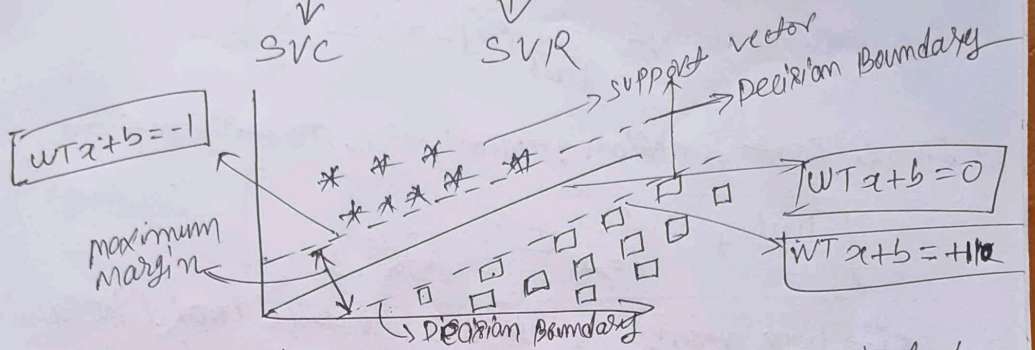
Goal is to · Seprate best suitable line or decision boundary that can segerate n dimensional space into classes so that we can easily put new data points into correct categorys in future.

This Best decision boundary is called Hyperplane. SVM choose the extreme point or support vector. that helps in creating the hyperplane we can see a strange cat that some features of dog; Ex: we want a model that can accurately identify whether it is a cat or dog. Train model with a lot of images of cat or dogs so that it can learn about the different features of dogs, SVM creates a decision boundary b/w these two data and choose extreme case of dog and cat. Ex ⇒ face detection

(*) Hyperplane: There can be multiple lines or the decision boundary to seprate the classes in n dimentional space but we need to find out the best decision boundary that help to classifies the data points, this best boundary is known of a hyper plane. The dimension of the hyper plane dependon upon the feteres present in the datalets which means if there are two features then hyper plane will be a straight line. If there are three feature then hyper plane will a 2D plane.

SVM

SVC SVR



Decision Boundaries are also known as marginal planes

vectors helpful to make decision boundaries are known as support vectors.

Hard Margin / Soft Margin

$y = wx+c$

$y = w^T x+c$

$ax+by+c$ equation of line

$by = -ax-c \Rightarrow \boxed{y = \frac{-ax}{b} - \frac{c}{b}}$

Let it a line $3x+2y+4=0$

$3(4)+2(4)+4 \Rightarrow 12+8+4 = 24$ (4,4)

$w^T x_1+b=+1$

$w^T x_2+b=-1$

how SVM used for classification of linearly separable data,

$w^T(x_1-x_2) = +2$ dividing w by magnitude, we get a vector

$\boxed{\frac{w^T(x_1-x_2)}{|w|} = \frac{2}{|w|}}$ → we need to maximize it {maximum margin} given

Marginal plane distance

must see

$y = \begin{cases} +1 & w^T x+b \geq +1 \\ -1 & w^T x+b \leq -1 \end{cases}$

$y w^T x + b \geq 1$

for minimising $\frac{w^T}{|w|}(x_1 - x_2) = \frac{|w|}{2}$

final cost function: Minimise cost function

$$\frac{|w|}{2} + c_i \sum_{i=1}^{n} \varepsilon \quad \{eta\}$$

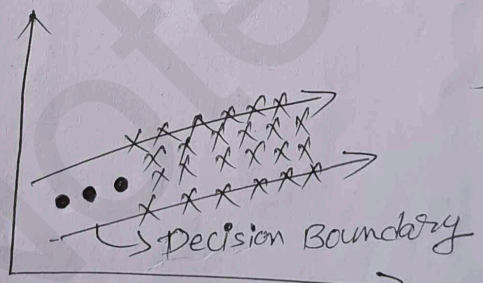
$c_i \rightarrow$ how many points we want to avoid because some errors are acceptable.

$\varepsilon \rightarrow$ eta is the submission of distance of misclassification of points from marginal plane.

Kernel → sigmoid kernel
→ Linear kernel
→ RBF kernel

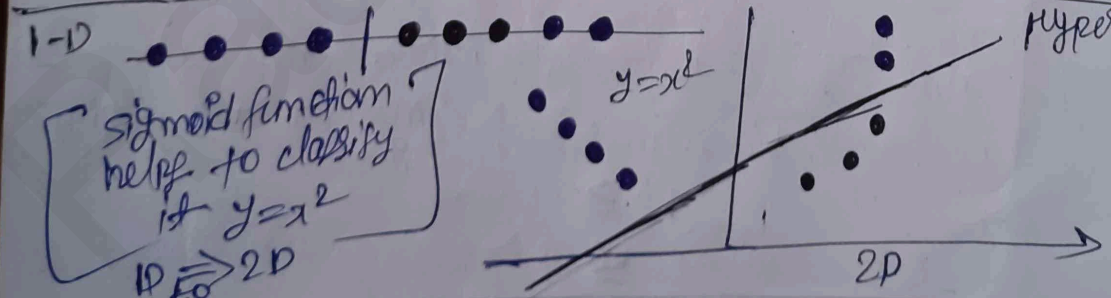
$y = (x^2)$ kernel Trick is used in SVM

* SVM Regression:



Regression to find a hyperplane such that most of the data points are inside marginal line:

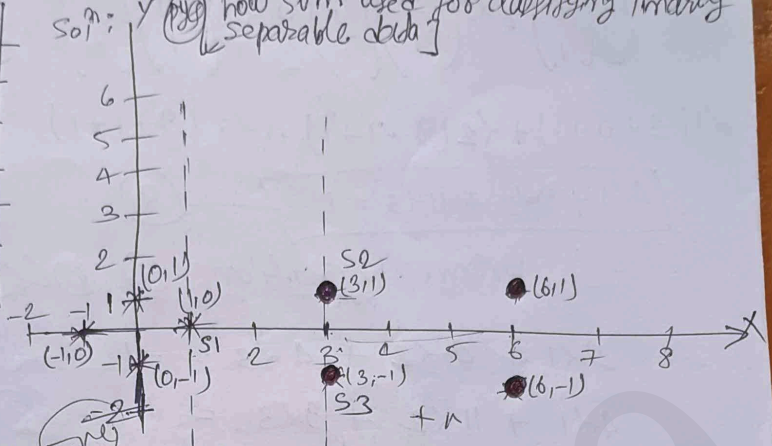
Same for $\frac{|w|}{2} + c_i \cdot \sum_{i=1}^{n} \varepsilon \rightarrow (eta)$



Q Generate a hyper plane with the following points (Using SVM used for classifying linearly separable data)

(3,1)	+1
(3,-1)	+1
(6,1)	+1
(6,-1)	+1
(1,0)	-1
(0,1)	-1
(0,-1)	-1
(-1,0)	-1

Soln:



Soln:

$S_1 = \begin{pmatrix}1\\0\end{pmatrix}$, $S_2 = \begin{pmatrix}3\\1\end{pmatrix}$ $S_3 = \begin{pmatrix}3\\-1\end{pmatrix}$

Adding Bias vectors to convert these vector to augmented vectors.

$\bar{S_1} = \begin{pmatrix}1\\0\\1\end{pmatrix}$, $\bar{S_2} = \begin{pmatrix}3\\1\\1\end{pmatrix}$, $\bar{S_3} = \begin{pmatrix}3\\-1\\1\end{pmatrix}$

$\alpha_1, \alpha_2, \alpha_3$ are slopes

$\alpha_1 \bar{S_1}\bar{S_1} + \alpha_2 \bar{S_1}\cdot\bar{S_2} + \alpha_3 \bar{S_1}\bar{S_3} = -1$

$\alpha_1 \bar{S_1}\bar{S_2} + \alpha_2 \bar{S_2}\bar{S_2} + \alpha_3 \bar{S_3}\bar{S_2} = +1$

$\alpha_1 \bar{S_1}\bar{S_3} + \alpha_2 \bar{S_2}\bar{S_3} + \alpha_3 \bar{S_3}\bar{S_3} = +1$

$$\alpha_1 \begin{pmatrix}1\\0\\1\end{pmatrix}\begin{pmatrix}1\\0\\1\end{pmatrix} + \alpha_2 \begin{pmatrix}1\\0\\1\end{pmatrix}\begin{pmatrix}3\\1\\1\end{pmatrix} + \alpha_3 \begin{pmatrix}1\\0\\1\end{pmatrix}\begin{pmatrix}3\\-1\\1\end{pmatrix} = -1$$

$\alpha_1(1+0+1) + \alpha_2(3+0+1) + \alpha_3(3+0-1) = -1$

$2\alpha_1 + 4\alpha_2 + 4\alpha_3 = -1$ — (1)

$$\alpha_1 \begin{pmatrix}1\\0\\1\end{pmatrix}\begin{pmatrix}3\\1\\1\end{pmatrix} + \alpha_2 \begin{pmatrix}3\\1\\1\end{pmatrix}\begin{pmatrix}3\\1\\1\end{pmatrix} + \alpha_3 \begin{pmatrix}3\\1\\1\end{pmatrix}\begin{pmatrix}3\\-1\\1\end{pmatrix} = 1$$

$\alpha_1(3+0+1) + \alpha_2(9+1+1) + \alpha_3(9-1+1) = 1$

$4\alpha_1 + 11\alpha_2 + 9\alpha_3 = 1$ — (2)

$$\alpha_1\begin{pmatrix}3\\-1\\1\end{pmatrix}\begin{pmatrix}1\\0\\1\end{pmatrix}+\alpha_2\begin{pmatrix}3\\-1\\1\end{pmatrix}\begin{pmatrix}3\\1\\1\end{pmatrix}+\alpha_3\begin{pmatrix}3\\-1\\1\end{pmatrix}\begin{pmatrix}3\\-1\\1\end{pmatrix}$$

$$\alpha_1(3+0+1)+\alpha_2(9-1+1)+\alpha_3(9+1+1)$$

$$\boxed{4\alpha_1+9\alpha_2+11\alpha_3=1} \quad \text{—— (3)}$$

From equation (1), (2) & (3)

$$2\alpha_1+4\alpha_2+4\alpha_3=-1 \quad \text{— (1)}$$
$$4\alpha_1+11\alpha_2+9\alpha_3=1 \quad \text{— (2)}$$
$$4\alpha_1+9\alpha_2+11\alpha_3=1 \quad \text{— (3)}$$

From eqn (2) & (3)

$$4\alpha_1+11\alpha_2+9\alpha_3=1$$
$$4\alpha_1+9\alpha_2+11\alpha_3=1$$

$$2\alpha_2+(-2\alpha_3)=0$$
$$2\alpha_2-2\alpha_3=0$$
$$\boxed{\alpha_2=\alpha_3}$$

Multiply (2) in eq (1)

$$4\alpha_1+8\alpha_2+8\alpha_3=-2$$ taking eqn (3)
$$4\alpha_1+9\alpha_2+11\alpha_3=1$$

$$-\alpha_2-3\alpha_3=-3$$
$$\alpha_2+3\alpha_3=3 \quad \text{& put } \alpha_3=\alpha_2$$
$$\alpha_2+3\alpha_2=3$$
$$4\alpha_2=3$$
$$\boxed{\alpha_2=\frac{3}{4}} \quad \because \alpha_2=\alpha_3 \quad \boxed{\alpha_3=\frac{3}{4}}$$

put α_2 and α_3 in eq (1)

$2\alpha_1 + 4\alpha_2 + 4\alpha_3 = -1$

i. $2\alpha_1 + 4\times\frac{3}{4} + 4\times\frac{3}{4} = -1$

$2\alpha_1 + 3 + 3 = -1$

$2\alpha_1 + 6 = -1 \Rightarrow 2\alpha_1 = -1 - 6$

$2\alpha_1 = -7$

$\alpha_1 = -\frac{7}{2}$ $\boxed{\alpha_1 = -3.5}$

$\because \boxed{\vec{w} = \sum_{i=1}^{n} \alpha_i \vec{s_i}}$

$\vec{w} = -3.5\begin{pmatrix}1\\0\\1\end{pmatrix} + 0.75\begin{pmatrix}3\\1\\1\end{pmatrix} + 0.75\begin{pmatrix}3\\-1\\1\end{pmatrix}$

$\vec{w} = \begin{pmatrix}-3.5\\0\\-3.5\end{pmatrix} + \begin{pmatrix}2.25\\0.75\\0.75\end{pmatrix} + \begin{pmatrix}2.25\\-0.75\\0.75\end{pmatrix}$

$\vec{w} = \begin{pmatrix}1\\0\\-2\end{pmatrix}$] w slop &
-> bias & intercept)

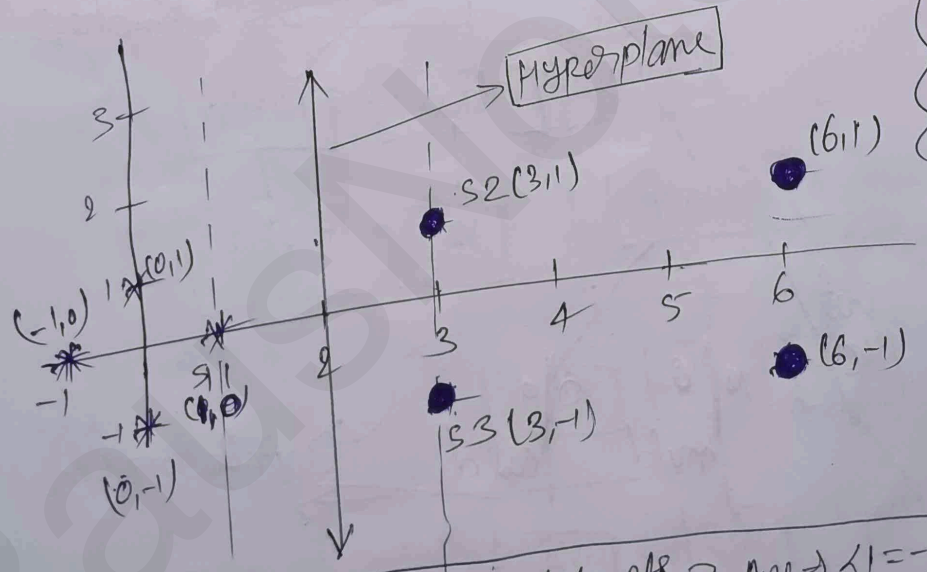
$w = \begin{pmatrix}1\\0\end{pmatrix}$ x, y → hyperplane will be parallel to y

$b = -2$

$\boxed{b + 2 = 0}$

best if $\begin{pmatrix}1\\1\end{pmatrix}$ then it is to x & y

$\begin{pmatrix}0\\1\end{pmatrix}$ line parallel to x axis

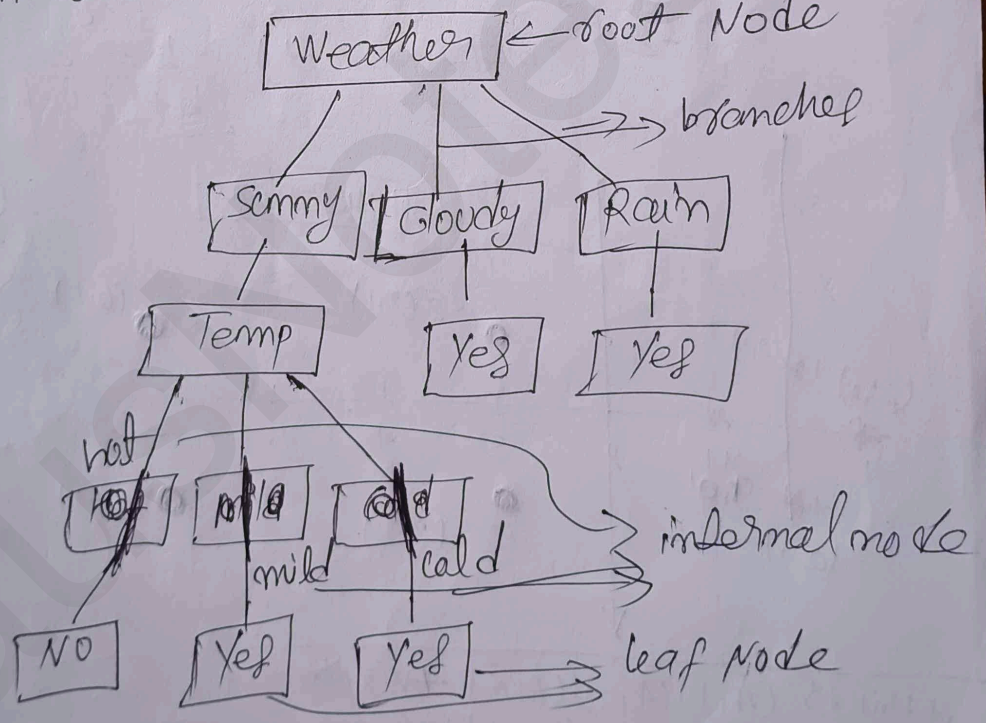


*How => (4,1) (4,-1) (6,0) +ve pts
(1,0) (0,1) (0,-1) -ve pts

Any → $\alpha_1 = -3$
$\alpha_2 = 1$
$\alpha_3 = 0$

Decision Tree (classification)

Decision Tree is a supervised Algorithm used for both classification and regression, it models decision tree like structure where internal node represent attribute test, branches represent attribute value and leaf node represent final decision. It's a graphical represent of different options for solving a problem. It has a hierarchical tree structure having root node. It is starting points that represent entire dataset. Branches are the lines that connect nodes. It shows the flow from one decision to another.



* Root Node: Root Node is the starting point that represents the entire dataset.

* Branches: These are the lines that connect nodes. It shows the flow from one decision to another.

* Internal Nodes: Internal Nodes are points where decisions are made based on the input features.

* Leaf Node: These are the terminal nodes at the end of branches that represent final outcomes or predictions.

* Gini index:

* Entropy (H): Entropy measures the impurity or uncertainty in the dataset. The formula for entropy is:

$$H(S) = -\sum_{i=1}^{c} p_i \log_2 p_i$$

where
$H(S)$ → entropy of set S.
c → c is the no of classes in dataset.
p_i → p_i is the proportion of samples belonging to class i.
$\log_2 p_i$ → is the logarithm (base 2) of p_i.

$$H(S) = -[P_+ \log_2 P_+] - [P_- \log_2 P_-]$$

- P_- → Probability of NO
- P_+ → Probability of Yes

* Information Gain (IG): Information Gain measures the reduction in entropy after a dataset is split on an attribute. It is given by:

$$IG(S, A) = H(S) - \sum_{v \in values(A)} \frac{|S_v|}{|S|} H(S_v)$$

- $H(S_v)$: is the entropy of subset S_v.

- $IG(S, A)$ is the information Gain for splitting on attribute A
- $H(S)$ is the entropy of the original dataset.
- S_v is the subset of S where attribute A takes value v.
- $\frac{|S_v|}{|S|}$ is the proportion of S that falls into subset

Q: Decision Tree:

Day	Weather	Temperature	Humidity	wind	play
Day 1	Sunny	Hot	High	weak	NO
Day 2	Sunny	Hot	High	strong	NO
Day 3	cloudy	Hot	High	weak	Yes
Day 4	Rain	Mild	High	weak	Yes
Day 5	Rain	Cool	Normal	weak	Yes
Day 6	Rain	cool	Normal	strong	NO
Day 7	cloudy	cool	Normal	strong	Yes
Day 8	sunny	Mild	High	weak	NO
Day 9	sunny	Cool	Normal	weak	Yes
Day 10	Rain	Mild	normal	weak	Yes
Day 11	sunny	Mild	Normal	strong	Yes
Day 12	cloudy	Mild	High	strong	Yes
Day 13	cloudy	Hot	Normal	weak	Yes
Day 14	Rain	Mild	High	strong	NO

Now Calculate IG (information Gain) of Weather

Step1: Entropy of Entire dataset

$H(W) = -\sum_{i=1}^{c} p_i \log_2 p_i$ (whole) $\Big\{ H(w) = -P_+ \log_2 P_+ - P_- \log_2 P_- \Big\}$ or

$H(9,-5) = -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14} = 0.94$

Step2: Entropy of all attributes {weather}

- Entropy of Sunny $\{+2,-3\} = -\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5} = 0.97$
- Entropy of cloudy $\{+4,-0\} = -\frac{4}{4}\log_2\frac{4}{4} - \frac{0}{4}\log_2\frac{0}{4} = 0$
- Entropy of Rain $\{+3,-2\} = -\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5} = 0.97$

Information Gain = Entropy(whole data) $- \frac{5}{14}$Ent(S) $- \frac{4}{14}$Ent(C) $- \frac{5}{14}$Ent(R)

$= 0.246$

* Calculate IG of Temperature

Step1: Entropy of entire dataset

$H(+9,-5) = 0.94$

Step2: Entropy of Hot $\{+2,-2\} = -\frac{2}{4}\log_2\frac{2}{4} - \frac{2}{4}\log_2\frac{2}{4} = 1.0$

Entropy of mild $\{+4,-2\} = -\frac{4}{6}\log_2\frac{4}{6} - \frac{2}{6}\log_2\frac{2}{6} = 0.91$

" " cool $\{+3,-1\} = -\frac{3}{4}\log_2\frac{3}{4} - \frac{1}{3}\log_2\frac{1}{3} = 0.81$

Information Gain = Entropy(whole dataset) $- \frac{4}{14}$ Ent(H) $- \frac{6}{14}$ Ent(m)

$- \frac{4}{14}$ Ent(c) $\Rightarrow 0.029$

* Calculate IG of wind

Step1: Entropy of entire dataset

$H(+9,-5) = 0.94$

Step2: Entropy of all attributes

Entropy of strong $\{+3,-3\} = -\frac{3}{6}\log_2\frac{3}{6} - \frac{3}{6}\log_2\frac{3}{6} = 1.0$

Entropy of weak $\{+6,-2\} = -\frac{6}{8}\log_2\frac{6}{8} - \frac{2}{8}\log_2\frac{2}{8} = 0.81$

IG = Entropy(whole data) $- \frac{6}{14}$ Ent(S) $- \frac{8}{14}$ Ent(W) $= 0.0478$

* Calculate IG of Humidity

Step1: Entropy of entire dataset $H\{+9,-5\} = -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14}$

$= 0.94$

Step2: Entropy of all attributes:

Entropy of High $\{+3,-4\} = -\frac{3}{7}\log_2\frac{3}{7} - \frac{4}{7}\log_2\frac{4}{7} = 0.98$

" of Normal $\{+6,-1\} = -\frac{6}{7}\log_2\frac{6}{7} - \frac{1}{7}\log_2\frac{1}{7} = 0.59$

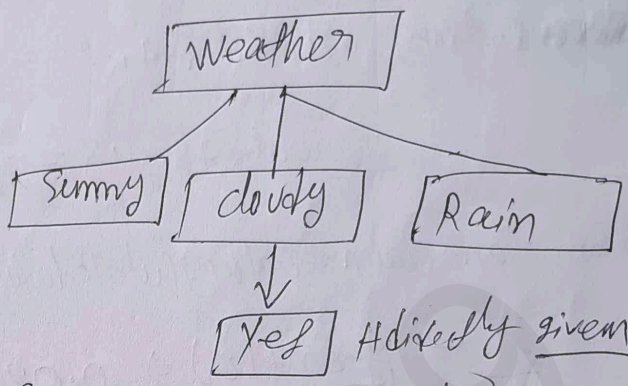
$IG = Entropy(whole\ data) - \frac{7}{14} Ent(H) - \frac{7}{14} Ent(N)$

$= 0.15$

$IG(Weather) = 0.246$.
$IG(Temp) = 0.029$
$IG(Humidity) = 0.15$
$IG(wind) = 0.0478$

Root Node = Max of all

Root Node = Weathe (0.246)



Hdirectly given

Now we will calculate IG from sunny point of view.

~~* Calculate IG of~~

Day	weather	Temp	Humidity	wind	play
Day 1	Sunny	Hot	High	weak	NO
Day 2	Sunny	Hot	High	strong	NO
Day 3	Sunny	Mild	High	weak	NO
Day 4	Sunny	cool	normal	weak	Yes
Day 5	Sunny	Mild	Normal	strong	Yes

* Calculate IG of Temperature

Step 1: Entropy of Sunny $[+2, -3] = -\frac{2}{5}\log\frac{2}{5} - \frac{3}{5}\log\frac{3}{5} = 0.97$

Step 2: Entropy of all attributes: (Temp)

Entropy of Hot $[+0, -2] = -\frac{0}{2}\log_2\frac{0}{2} - \frac{2}{2}\log_2\frac{2}{2} = 0$

" " Mild $[+1, -1] = -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2} = 1.0$

" " cool $[+1, -0] = -\frac{1}{1}\log_2\frac{1}{1} - \frac{0}{1}\log_2\frac{0}{1} = 0$

$IG = Entropy(sunny) - \frac{2}{5} Ent(H) - \frac{2}{5} Ent(M) - \frac{1}{5} Ent(C)$

$= 0.57$

✱ Calculate IG of ~~Wind~~ Humidity

Step1: Entropy of Sunny ⇒ H{+2,-3} = 0.97

Step2: Entropy of all attributes:

Entropy of High {+0,-3} = $-\frac{0}{3}\log_2\frac{0}{3} - \frac{3}{3}\log_2\frac{3}{3} = 0$

" " Normal {+2,-0} = $-\frac{2}{2}\log_2\frac{2}{2} - \frac{0}{2}\log_2\frac{0}{2} = 0$

IG = Entropy(Sunny) − $\frac{3}{5}$Ent(H) − $\frac{2}{5}$Ent(N) = 0.97

✱ Calculate IG of Wind

Step1: Entropy of Sunny {+2,-3} = 0.97

Step2: Entropy of all attribute

- Entropy of Strong {+1,-1} = $-\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2} = 1$
- " " Weak {+1,-2} = $-\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3} = 0.918$

~~Mi~~

IG = Entropy(Sunny) − $\frac{2}{5}$Ent(S) − $\frac{3}{5}$Ent(W)

= 0.019

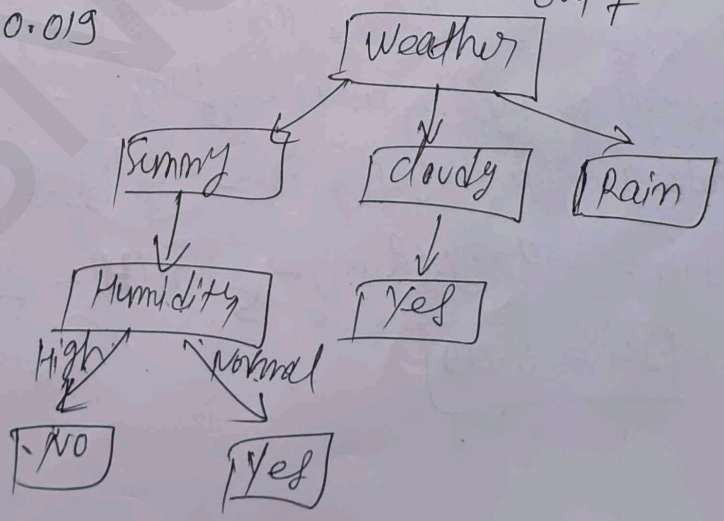
IG (Temp) = 0.57

IG (Humidity) = 0.97

IG (Wind) = 0.019

Node (Humidity)

0.97



Day	weather	Temperature	Humidity	~~play~~ wind	play
Day 4	Rain	Mild	High	Weak	Yes
Day 5	Rain	cool	Normal	weak	Yes
Day 6	Rain	cool	Normal	Strong	No
Day 10	Rain	Mild	Normal	~~Strong~~ Weak	Yes
Day 14	Rain	Mild	High	Strong	No

* Calculate IG1 of Temperature

Step1 : Entropy of ~~Rain~~ Tem [+3, -2] $= -\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5} = 0.97$

Step2 : Entropy of all attributes

- Entropy of Hot [+0, -0] $= -\frac{0}{2}\log_2\frac{0}{2} - \frac{0}{2}\log_2\frac{0}{2} = 0$
- Entropy of mild [+2, -1] $= -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3} = 0.918$
- Entropy of cool [+1, -1] $= -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2} = 1.0$

IG = Entropy(Rain) $- \frac{0}{5}$ Ent(H) $- \frac{3}{5}$ Ent(M) $- \frac{2}{5}$ Ent(C)

IG = 0.019

* Calculate IG of Humidity

Step1 : Entropy of ~~Rain~~ Hum [+3, -2] $= -\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5} = 0.97$

Step2 : Entropy of all attributes :

Entropy of High [+1, -1] $= -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2} = 1$

Entropy of Normal [+2, -1] $= -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3} = 0.918$

IG1 = Entropy(Rain) $- \frac{2}{5}$ Ent(H) $- \frac{3}{5}$ Ent(N)

IG = 0.019

* Calculate IG of wind

Entropy of ~~Rain~~ wind $\{+3, -2\} = -\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5} = 0.97$

Entropy of all attributes:

Entropy of strong $\{+0, -2\} = -\frac{0}{2}\log_2\frac{0}{2} - \frac{2}{2}\log_2\frac{2}{2} = 0$

Entropy of weak $\{+3, -0\} = -\frac{3}{3}\log_2\frac{3}{3} - \frac{0}{3}\log_2\frac{0}{3} = 0$

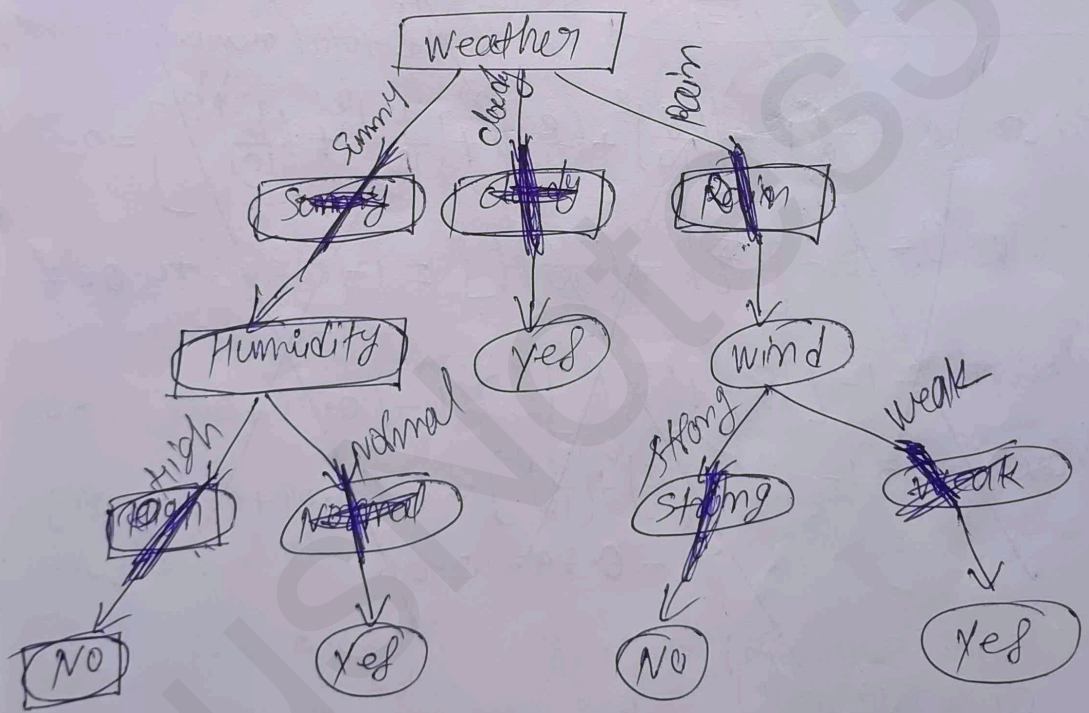
$IG = Ent(Rain) - \frac{2}{5}Ent(S) - \frac{3}{5}Ent(W)$

$IG = 0.97$

Now, $IG(\text{~~Rain~~ temp}) = 0.019$

$IG(Hum) = 0.019$

$IG(wind) = 0.97$ ✓ (~~Root~~ Leaf Node)



Weekend	Weather	parents	Money	Decision
W1	Sunny	Yes	Rich	cinema
W2	Sunny	No	Rich	Tennis
W3	windy	Yes	Rich	Cinema
W4	Rainy	Yes	Poor	cinema
W5	Rainy	No	Rich	Stay in
W6	Rainy	Yes	Poor	cinema
W7	windy	No	Poor	Cinema
W8	windy	No	Rich	Shopping
W9	windy	Yes	Rich	cinema
W10	Sunny	No	Rich	Tennis

There are 4 possible outcome variables ~~cinema~~, ~~Tennis~~, ~~stay in~~, ~~shopping~~ ~~weather~~, ~~parents~~, Money, Decision

$$\text{Gini Idx} = 1 - \sum_{i=1}^{n} Pi^2$$

Pi = is the probability of class i.
n is the total number of class.

$$\text{Gini (Decision)} = 1 - \left[\left(\frac{6}{10}\right)^2 + \left(\frac{2}{10}\right)^2 + \left(\frac{1}{10}\right)^2 + \left(\frac{1}{10}\right)^2\right] = 0.58$$

$$\text{Gini (Money)} = 1 - \left[\left(\frac{7}{10}\right)^2 + \left(\frac{3}{10}\right)^2\right] = 1 - [0.343 + 0.027]$$
$$= 1 - 0.37 = 0.63$$

$$\text{Gini (parents)} = 1 - \left[\left(\frac{5}{10}\right)^2 + \left(\frac{5}{10}\right)^2\right] \Rightarrow 1 - [0.25 + 0.25] = 0.5$$

$$\text{Gini (weather)} = 1 - \left[\left(\frac{3}{10}\right)^2 + \left(\frac{4}{10}\right)^2 + \left(\frac{3}{10}\right)^2\right] = 1 - [0.09 + 0.16 + 0.09]$$
$$\Rightarrow 1 - 0.34 = 0.66$$

Gini (Decision) = 0.58
Gini (Money) = 0.63
Gini (parents) = 0.5
Gini (weather) = 0.66

Decision Tree (classification)

In decision there are 4 possible out-comes variables cinema, Tennis, stay-in, shopping

$$Gini(Decision) = \left[1 - \sum_{i=1}^{n} pi^2\right]$$

pi = is the probability of class i.
n is the total number of classes

Gini for entire dataset

$$Gini(Decision) = \left[1 - \left(\frac{6}{10}\right)^2 + \left(\frac{2}{10}\right)^2 + \left(\frac{1}{10}\right)^2 + \left(\frac{1}{10}\right)^2\right] = 0.58$$

there are three elements that affects Decision so we will Calculate Gini for Money, parents, weather them we will get out root Node.

① Gini (Money) → 2 variables Rich → 7 instances
poor → 3 instances

Gini (Rich) ⇒ 7 instances → 3 cinema
→ 2 Tennis
→ 1 stay-in
→ 1 shopping

$$Gini(Rich) = \left[1 - \left(\frac{3}{7}\right)^2 + \left(\frac{2}{7}\right)^2 + \left(\frac{1}{7}\right)^2 + \left(\frac{1}{7}\right)^2\right] = 0.694$$

Gini (Poor) ⇒ 3 instances → 3 cinemas

$$Gini(Poor) = \left[1 - \left(\frac{3}{3}\right)\right] = 0$$

weighted Avg Gini for Money = (Gini Idx) × (proportion)

$$Avg = \left(0.694 \times \frac{7}{10}\right) + \left(0 \times \frac{3}{10}\right) = 0.486$$

* Gini(parents) ⇒ 2 variables → Yes → 5 Inf
→ No → 5 Inf

- Gini(Yes) ⇒ 5 instances → 5 Cinemas

Gini(Yes) = 0

- Gini(No) ⇒ 5 Inst → 2 Tennis
→ 1 shopping
→ 1 stay-in
→ 1 cinema

$$Gini(No) = \left[1 - \left(\frac{2}{5}\right)^2 + \left(\frac{1}{5}\right)^2 + \left(\frac{1}{5}\right)^2 + \left(\frac{1}{5}\right)^2\right] =$$

Gini(No) = 0.72

Weight Avg Gini for parents $= \left(0 \times \frac{5}{10}\right) + \left(0.72 \times \frac{5}{10}\right)$

Avg = 0.36

* Gini(Weather) ⇒ 3 variables → Sunny → 3 inf
→ windy → 4 inf
→ rainy → 3 inf

Gini(Sunny) ⇒ 3 inf → 1 cinema
2 Tennis

$$Gini(Sunny) = \left[1 - \left(\frac{1}{3}\right)^2 + \left(\frac{2}{3}\right)^2\right] = 0.444$$

Gini(windy) ⇒ 4 inf → 3 cinema
1 shopping

$$Gini(windy) = \left[1 - \left(\frac{3}{4}\right)^2 + \left(\frac{1}{4}\right)^2\right] = 0.375$$

Gini(rainy) ⇒ 3 inf → 2 cinema
1 stay in

$$Gini(rainy) = \left[1 - \left(\frac{2}{3}\right)^2 + \left(\frac{1}{3}\right)^2\right] = 0.444$$

Weighted Avg gini for weather $= \left(0.444 \times \frac{3}{10}\right) + \left(0.375 \times \frac{4}{10}\right) + \left(0.444 \times \frac{3}{10}\right)$

Avg = 0.416

Gini for Money = 0.486
" " parent = 0.36
weather = 0.416
choose → minimum

Parent → Yes / No

Gini for parent

week	weath	parent	money	Decision
W1	Sunn	Yes	Rich	Cin
W3	Win	Yes	Rich	Cin
W4	Rain	Yes	poor	Cin
W6	Rain	Yes	poor	Cin
W9	Win	Yes	Rich	Cin

week	weath	parent	money	Decision
W2	Sunn	No	Rich	Tennis
W5	Rain	No	Rich	Stay-in
W7	Win	No	Poor	Cin
W8	Win	No	Rich	Shop
W10	Sunn	No	Rich	Tennis

parent → Yes / No

Yes → Cinema

* Gini (parent = No & weather)

Gini(weather) → 3 variable → Sunny → 2 inf
→ Rainy → 1 Inf
→ windy → 2 inf

Gini(sunny) ⇒ 2 inf. → 2 Tennis

$$Gini(Sunny) = 1 - \left[\frac{2}{2}\right]^2 = 0$$

Gini(Rainy) → 1 inf → 1 Stay-in

$$Gini(Rainy) = \left[1 - \left(\frac{1}{1}\right)^2\right] = 0$$

Gini(windy) = 2 inf → 1 shopp
→ 1 chem

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

Weighted Avg Gin for weather $= (0 \times \frac{2}{5}) + (0 \times \frac{1}{5}) + (0.5 \times \frac{2}{5})$

Avg = 0.2

Decision

Gini(Parent = No of money)

Gini(Money) ⇒ vary → 4 Inst → Rich
→ 1 Inst → Poor

Gini(Rich) ⇒ 4 inst → 2 Term
→ 1 shop
→ 1 stay-in

$$Gini(Rich) = \left[1 - \left(\frac{2}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2\right] = 0.625$$

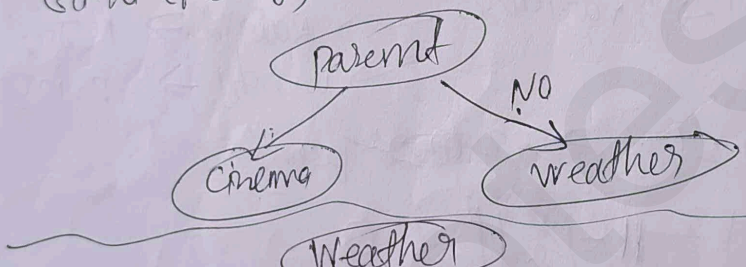
Gini(Poor) ⇒ 1 inst → cinema

Gini(poor) = 0

weighted Avg Gini for Money $= (0.625 \times \frac{4}{5}) + (0 \times \frac{1}{5})$

Avg = 0.5

Gini(weather) = 0.2 ✓ Selected
Gini(Money) = 0.5



Weather

Sunny | Windy | Rainy

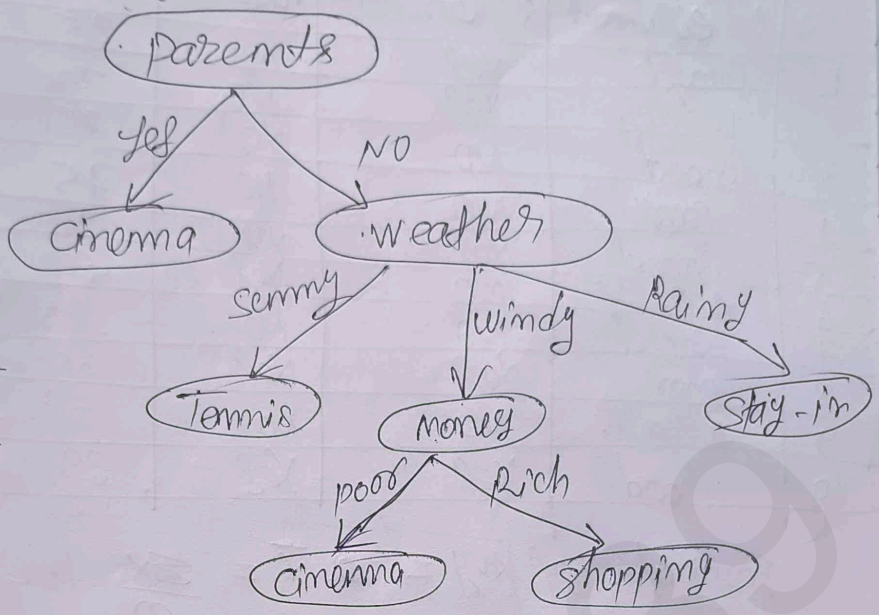
weather	parents	money	decision
W2	No	Rich	Term
W8	No	Rich	Shop
W10	No	Rich	Term

weather	parents	money	decision
W1	No	Poor	Cinema
W4	No	Poor	Cinema
W8	No	Rich	Shop

weather	parents	money	decision
W5	No	Rich	Stay-in

Now decision tree is ready

Decision Tree:



(Regression Tree)

Decision Tree (Regression)

SNO	Assessment	Assignment	Project	Result (%)
1	Good	Yes	Yes	95
2	Average	Yes	No	70
3	Good	No	Yes	75
4	Poor	No	No	45
5	Good	Yes	Yes	98
6	Average	No	Yes	80
7	Good	No	No	75
8	Poor	Yes	Yes	65
9	Average	No	No	58
10	Good	Yes	Yes	89

Formula of $\boxed{Avg = \bar{x} = \frac{\Sigma x}{n}}$

Standard deviation $S = \sqrt{\frac{\Sigma (x - \bar{x})^2}{n}}$

Computing Standard deviation for each attribute with respect to the target attribute which is result.

$Avg = (95+70+75+45+98+80+75+65+58+89)/10 = 75$

Standard deviation for entire dataset = 16.55

$$S = \sqrt{\frac{(95-75)^2+(70-75)^2+(75-75)^2+(45-75)^2+(98-75)^2+(80-75)^2+(75-75)^2+(65-75)^2+(58-75)^2+(89-75)^2}{10}}$$

$\boxed{S = 16.55}$ total Std deviation

Let Attribute

Assessment → Good
Assessment → Avg
Assessment → Poor

Assessment = Good

Avg = (95+75+98+75+89)/5 = 86.4

$$S = \sqrt{\frac{(95-86.4)^2+(75-86.4)^2+(98-86.4)^2+(75-86.4)^2+(89-86.4)^2}{5}}$$

$S = 10.9$

* Assessment = Average

Avg = (70+80+58)/3 = 69.3

$$S = \sqrt{\frac{(70-69.3)^2+(80-69.3)^2+(58-69.3)^2}{3}}$$

$S = 11.01$

Assessment = poor

Avg = (45+65)/2 = 55

$$S = \sqrt{\frac{(45-55)^2+(65-55)^2}{2}}$$

$S = 14.14$

	S	instances
Assessment (Good)	10.9	5
(Avg)	11.01	3
(poor)	14.14	2

weighted Standard deviation for Assessment :

$$\left(\frac{5}{10}\right) \times 10.9 + \left(\frac{3}{10}\right) \times 11.01 + \left(\frac{2}{10}\right) \times 14.14 = 11.58$$

Standard deviation reduction for Assessment

$\Rightarrow 16.55 - 11.58 = 4.97$

(*) ~~Standard~~ Assignment = Yes

Avg = (95+70+98+65+89)/5 = 83.4

Standard dev = $\sqrt{\frac{(95-83.4)^2+(70-83.4)^2+(98-83.4)^2+(65-83.4)^2+(89-83.4)^2}{5}}$

$S = 14.98$

Assignment = NO

Avg = $(75+45+80+75+58)/5 = 66.6$

$S = \sqrt{\frac{(75-66.6)^2+(45-66.6)^2+(80-66.6)^2+(75-66.6)^2+(58-66.6)^2}{5}}$

$S = 14.7$

Standard deviation for Assignment

Assignment	S	instances
Yes	14.98	5
NO	14.7	5

Weighted standard deviation for Ass $\Rightarrow \left(\frac{5}{10}\right) \times 14.98 + \left(\frac{5}{10}\right) \times 14.7$

$= 14.84$

Standard deviation reduction for Assignment

$\Rightarrow 16.55 - 14.84 = 1.71$

(A) Project = Yes

Avg = $95+75+98+80+65+89/6 = 83.7$

Std dev = $\sqrt{\frac{(95-83.7)^2+(75-83.7)^2+(98-83.7)^2+(80-83.7)^2+(65-83.7)^2+(89-83.7)^2}{6}}$

$S = 12.6$

(B) Project = NO Avg = $70+45+75+58/4 = 62$

Std dev = $\sqrt{\frac{(70-62)^2+(48-62)^2+(75-62)^2+(58-62)^2}{4}}$

Std dev = 13.39

Std dev for project

project	std deviation	data instances
Yes	12.6	6
No	13.39	4

Weighted standard deviation for project

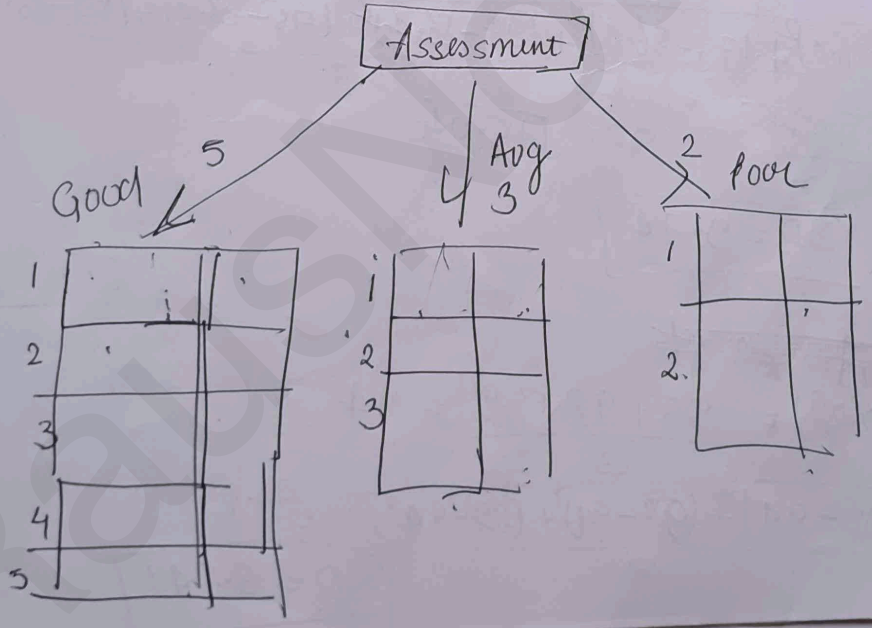
$\Rightarrow \left(\frac{6}{10}\right) \times 12.6 + \left(\frac{4}{10}\right) \times 13.39 = 12.92$

Std dev reduction for project = 16.55 − 12.92 = 3.63

Standard deviation Reduction for each Att

Attributes	std dev Reduction
Assessment	4.97
Assignment	1.71
Project	3.63

Max = Assessment



Assignment (yes) = 70 | SD = 0

Assignment (No) = $\frac{80+58}{2} = 69$

$SD = \sqrt{\frac{(80-69)^2+(58-69)^2}{2}}$ | SD = 11

weighted SD for Assignment $\Rightarrow \frac{1}{3} \times 0 + \left(\frac{2}{3}\right) \times 11 =$

Reduction SD for Assignment $\Rightarrow 8.99 - 7.33 =$ 1.65

project (yes) $\Rightarrow$ Avg = 80 | SD = 0

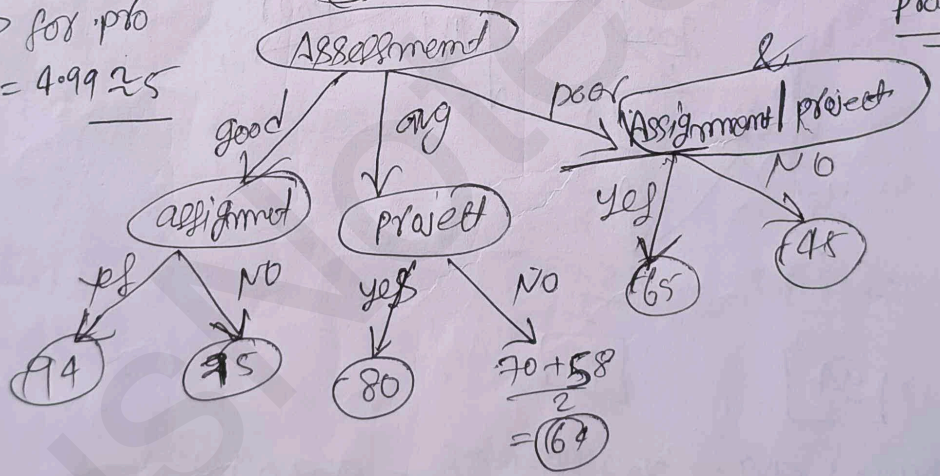
project (No) = $\frac{70+58}{2} = 64$

$SD = \sqrt{\frac{(70-64)^2+(58-64)^2}{2}} = \sqrt{\frac{72}{2}} = 6$

weighted SD for pro = $\left(\frac{1}{3}\right) \times 0 + \left(\frac{2}{3}\right) \times 6 =$ 4

Reduction SD for pro
$= 8.99 - 4 = 4.9925$

Maximum project



Assignment (No) $\Rightarrow \frac{75+75}{2} = 75$

$SD = \sqrt{\frac{(75-75)^2+(75-75)^2}{2}} = 0$

Weighted SD for assi $\Rightarrow \left(\frac{3}{5}\right) \times 3.74 + 0 \times \left(\frac{2}{5}\right) = 2.244$

Reduced SD for Assign $= 9.74 - 2.244 = 7.496$

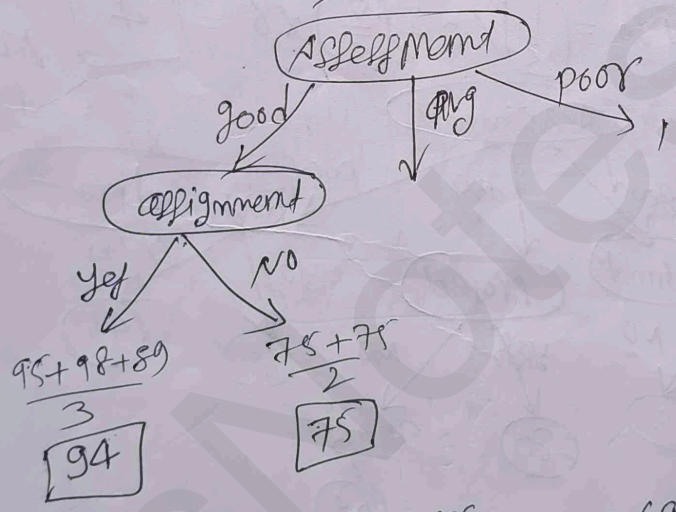
Project (Yes) $\Rightarrow$ Avg $= \frac{95+75+98+89}{4} = 89$

$SD = \sqrt{\frac{(95-89)^2+(75-89)^2+(98-89)^2+(89-89)^2}{4}} = \sqrt{\frac{313}{4}} = 8.84$

Project (No) $= 75$ $\quad SD = 0$

Weighted SD for Pro $\Rightarrow \left(\frac{4}{5}\right) \times 8.84 + \left(\frac{1}{5}\right) \times 0 = 7.072$

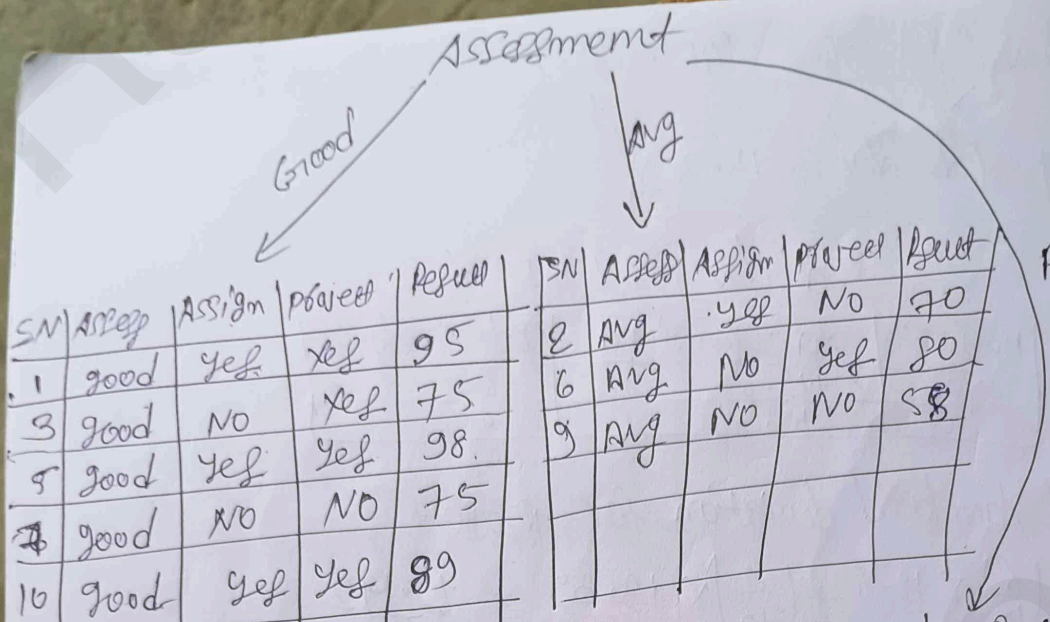
Reduction SD for Pro $\Rightarrow 9.74 - 7.072 = 2.668$



(2) Average: avg $= \frac{70+80+58}{3} = 69.33$

$SD = \sqrt{\frac{(70-69.3)^2+(80-69.3)^2+(58-69.3)^2}{3}}$

$SD = \sqrt{\frac{242.67}{3}}$ $\quad SD = 8.99$



SN	Assess	Assign	Proceed	Result
1	good	Yes	Yes	95
3	good	No	Yes	75
5	good	Yes	Yes	98
7	good	No	No	75
10	good	Yes	Yes	89

SN	Assess	Assign	Proceed	Result
2	Avg	Yes	No	70
6	Avg	No	Yes	80
9	Avg	No	No	58

SN	Assess	Assign	Proceed	Result
4	Poor	No	No	45
8	Poor	~~Yes~~	Yes	65
			Need to Calculate	Need to Calculate

(1) Good

$$Avg = \frac{95+75+98+75+89}{5} = \frac{432}{5} = 86.4$$

$$S.D = \sqrt{\frac{(95-86.4)^2+(75-86.4)^2+(98-86.4)^2+(75-86.4)^2+(89-86.4)^2}{5}}$$

$$SD = \sqrt{\frac{475.2}{5}} \qquad SD = \sqrt{95.04}$$

$$\boxed{SD = 9.74}$$

~~Assessment~~

$$\text{Assignment (Yes)} = \frac{95+98+89}{3} = 94$$

$$SD = \sqrt{\frac{(95-94)^2+(98-94)^2+(89-94)^2}{3}} = \cdots$$

$$\boxed{SD = 3.741}$$

(P26)

Random forest

Describe RF algo to improve accuracy.

Randomforest: Random Forest is a supervised machine learning algorithm that is used for both classification and regression. It is an ensemble learning technique that creates multiple decision trees and merges them together to produce a more accurate and stable prediction.

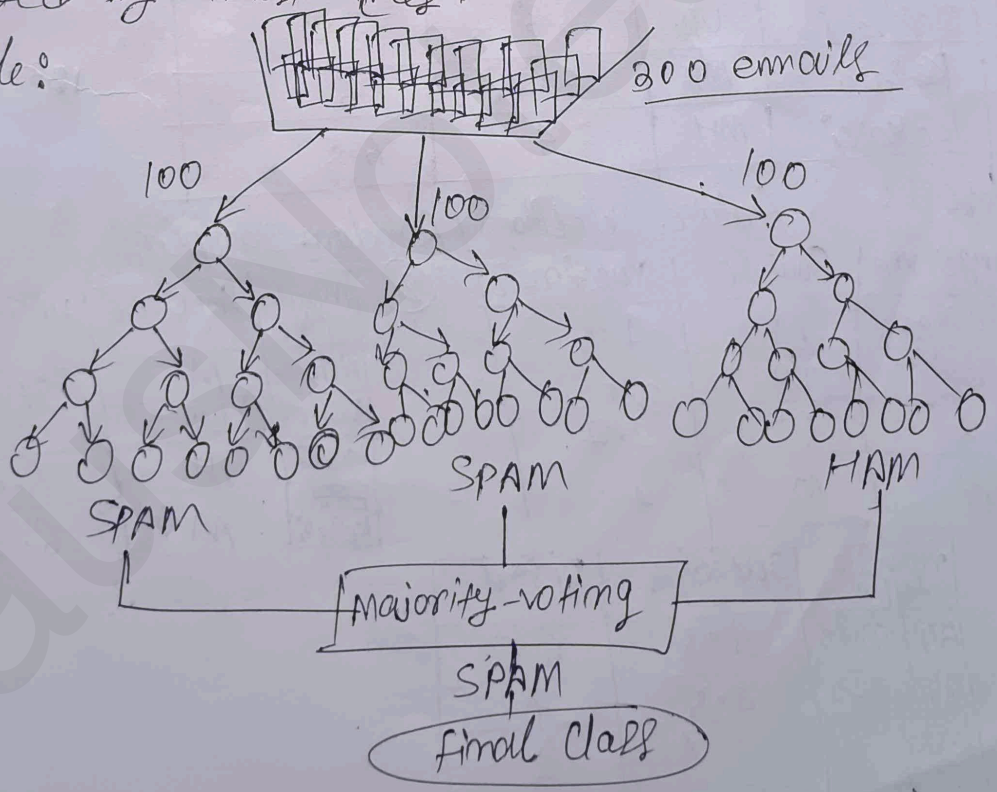
steps of Algorithm

step1: Create Bootstrap Dataset from original data by Randomly choosing data (repetition is allowed)

step2: Create Randomized Decision Tree from Bootstrap Dataset.

step3: Finally output of the random forest is the class selected by most trees.

Example:



Features of Random forest

1. It takes less training time as compared to other algorithms
2. It predicts output with high accuracy, even for the large dataset it runs efficiently.
3. It can also maintain accuracy when a large proportion of data is missing.
4. It can handle both classification and Regression.
5. It reduces overfitting.
6. It combines multiple decision trees to improve accuracy.
7. Low variance
8. High accuracy, No need of Normalization.

Ex =>

MangoType	Source	sweetness	Diameter
Alphanso	Maharashtra (MH)	4.5	7
Alphanso	MH	4.5	6.5
Desheri	UP	3.8	9
Desheri	UP	3.6	7.9
Kesari	MH	3.7	6
Kesari	MH	4	6.5

123, 124, 134

table 1: Feature Selection / Random Sampling

1.

MangoType	Source	sweetness
Alphanso	M	4.5
Desheri	UP	3.8
Kesri	M	3.7
Kesri	M	4

2.

Mango Type	Source	Diameter
Alphanso	M	7
Desheri	UP	7.9
Kesri	M	6.5
Kesari	M	6.5

3.

mango type	sweetness	Diameter
Alphanso	4.5	6.5
Alphanso	4.5	7
Kesri	3.7	6
Desheri	3.8	9

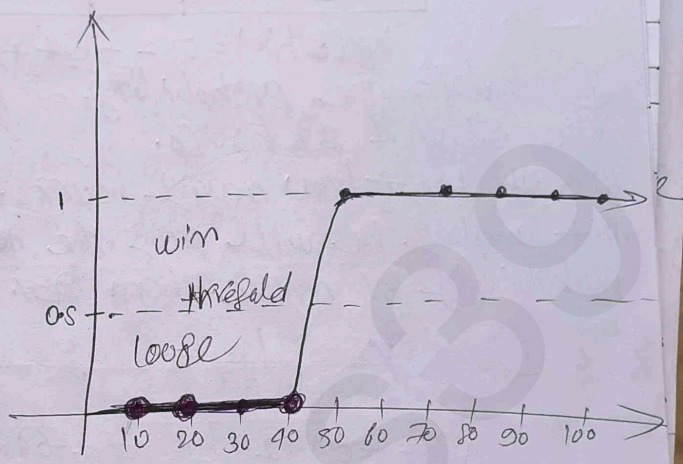
To Predict India will win or loose by watching runs of virat kholi.

$$y = \frac{1}{1+e^{-x}} \quad = \quad y = \frac{1}{1+e^{-(\beta_0 + \beta_1 x)}}$$

β_0 = intercept
β_1 = slope

Q.

Runs(x)	Win or loose(y)
10	0
20	0
40	0
50	1
70-80	1
80	1
90	1
100	1



In our example the runs above threshold tends to win the runs below threshold tends india will loose.

Advantages of Logistic Regression.

(i) performance better when data is linearly separable.
(ii) It does not required computational cost
(iii) does not require fine tuning.
(iv) Easy to implement and train the model.
(v) Gives the size of how relevant.

same of
the previous.

Logistic Regression (classification Alg)

~~Linear Regression~~

Logistic Regression: Logistic Regression is a supervised learn algorithm. It is a classification algorithm, used for predicting the output of a categorical dependent variable. The o/p can be either yes(1) or no(0). It gives the probability b/w 0 and 1.

Ex ⇒ disease, fraud detection, spam detection.

Types of Logistic Regression

(i) Binary (0 or 1)

(ii) Multinomial (three or more ordered type) [A, B, C]

Sigmoid function $y = \frac{1}{1+e^{-x}}$ — S-curved

Y is the dependent variable

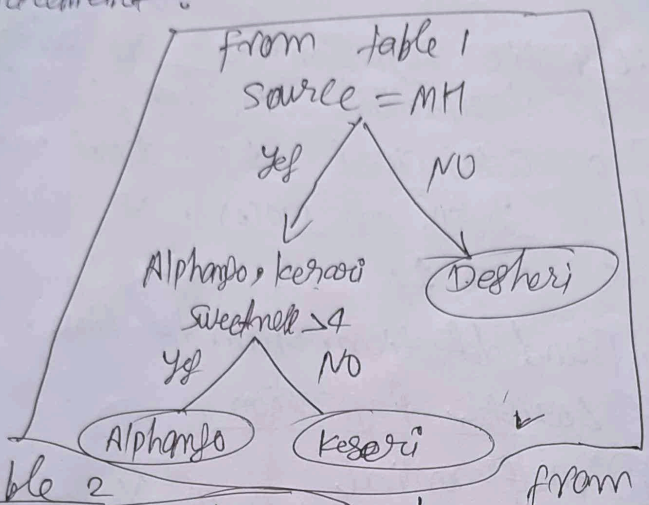
x → independent variable

e^x → exponential of independent variable

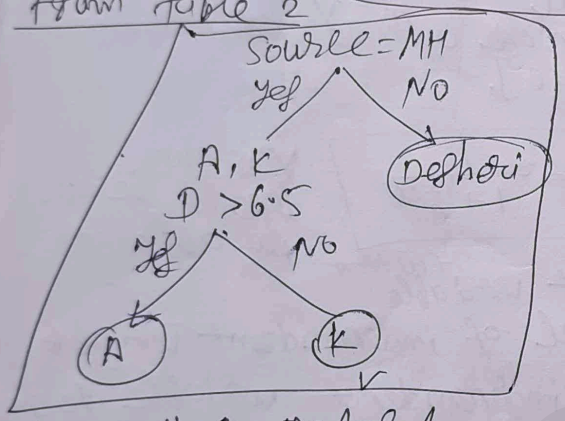
Sigmoid function converts the independent variable to an expression of probability in range b/w 0 and 1. It maps real value b/w 0 and 1. Target variable is must be binary and desired outcome is represented by factor level 1. No multi-collinearity in the model which means independent variable ~~of each~~ must be independent of each-other.

LG transform its output using logistic sigmoid function where y is a independent with the help of sigmoid function we are able to reduce a loss during time of training because it eliminates the gradient problem in the ML model during the training.

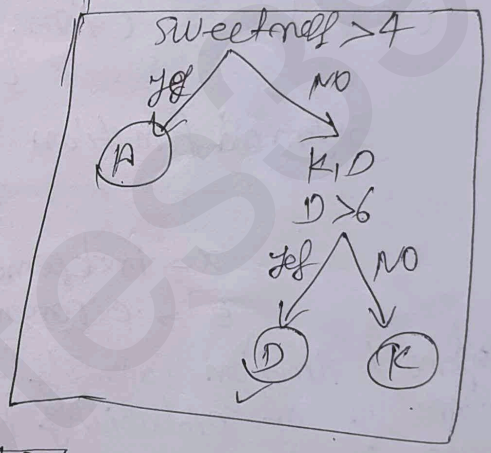
This particular process of creating Bootstrapping dataset is known as random sampling with replacement.



From table 2



From table 3



Testing Dataset

Source	Sweetness	Diameter
MH	3.9	6.4

According to first tree type = Kesari
" " second " " = Kesari
" " third " " = Deshari

Ans = Kesari

The new point is classified as category 2 because most of its closet neighbours are blue squares. KNN assigns the category based on the majority of nearby points.

- The black diamonds represent category 1 and blue squares represent category 2.
- The new data point checks its closest neighbours.
- Since the majority of its closest neighbours are blue squares (Category 2) KNN predicts the new data point belongs to category 2.

Working of KNN algorithm

Step 1: Selecting the optimal value of K (nearest neighbours)

Step 2: calculating distance: Euclidean distance

$$\sqrt{|x_1 - y_1|^2 + |x_2 - y_2|^2}$$

Step 3: Finding Nearest Neighbours.

Step 4: voting for classification or Taking Average for Regression.

Example:

IMDb Rating	Duration	Genre
8.0 (Mission impo)	160	Action
8.2 (Gadar 2)	170	Action
7.2 (Rocky & Rani	168	Comedy
8.2 (OMG 2)	155	Comedy

Predict the genre of Barbie movie with IMDb rating 7.4 and Duration 114 minutes.

Distance between cluster centers and data-points

Datapoints	Distance from K1 (2, 1.75	Distance from K2 (4.5, 4)	Assigned center
a1 (1,1)	1.25	4.61	K1
a2 (2,1)	0.75	3.9	K1
a3 (2,3)	1.25	2.69	K1
a4 (3,2)	1.03	2.5	K1
a5 (4,3)	2.36	1.12	K2
a6 (5,5)	4.42	1.12	K2

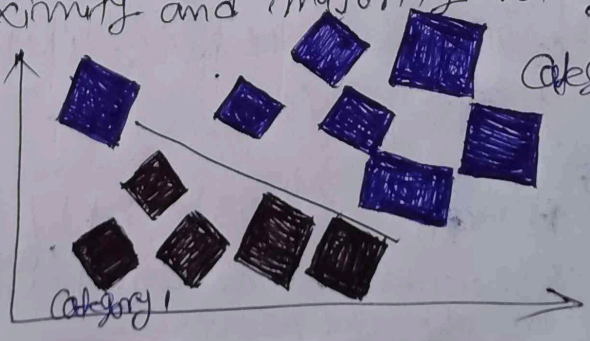
Euclidean distance $\sqrt{|x1-y1|^2 + |x2-y2|^2}$

cluster 1 of K1 = {a1, a2, a3, a4}
cluster 2 of K2 = {a5, a6}
cluster elements are same as in the previous iteration

final cluster is:
cluster 1: {(1,1)(2,1)(2,3)(3,2)}
cluster 2: {(4,3), (5,5)}

(12) K-Nearest Neighbour (K-NN): K-Nearest Neighbour (K-NN) is a supervised machine learning algorithm used for classification and regression. It is a lazy algorithm, meaning it doesn't explicitly learn a model during training. Instead, it memorizes the training dataset and makes decisions only during prediction. K-NN works by using proximity and majority voting to make predictions.

Ex:



Step 5: Repeat from step 2 untill we get same duster center or same duster elements as in the previous iteration.

Distance table :-

Data point	distance from K1(2, 1.33)	Distance from K2 (3.67, 3.67)	Assigned center
a1 (1,1)	1.05	3.78	K1
a2 (2,1)	0.33	3.15	K1
a3 (2,3)	1.67	1.8	K1
a4 (3,2)	1.204	1.8	K1
a5 (4,3)	2.605	0.75	K2
a6 (5,5)	4.74	1.88	K2

duster 1 of K1=(a1, a2, a3, a4)
duster 2 of K2 = {a5, a6}

calculate Euclidean distance for duster 1
$\sqrt{(x_1-y_1)^2+(x_2-y_2)^2}$
$\sqrt{(1-2)^2+(1-1.33)^2} = 1.05$

duster elements are not same as in the previous iteration.

duster 1: {(1,1)(2,1)(2,3);(3,2)}
duster 2: {(4,3)(5,5)}

for duster 2
$Ed = \sqrt{(1-3.67)^2+(1-3.67)^2}$
$Ed = 3.78$

duster 1: {a1, a2, a3, a4}
duster 2: {a5, a6}

Recalculating the duster centers

$K1 = \frac{1}{4}[a1+a2+a3+a4] \Rightarrow \frac{1}{4}[(1,1)+(2,1)+(2,3)+(3,2)]$

$= \frac{1}{4}[8,7] = (2, 1.75)$

$K2 = \frac{1}{2}[a5+a6] = \frac{1}{2}[(4,3)+(5,5)] \Rightarrow \frac{1}{2}(9,8) = (4.5, 4)$ centers

So duster elements and centers are not same as in the previous.

3. We repeat the process for a given number of iterations and at the end, we have our clusters.

Q. Use k-means clustering algorithm to divide the following data into two clusters.

x1	1	2	2	3	4	5
x2	1	1	3	2	3	5

Ans ⇒

Step 1: choosing randomly 2 cluster centers.

say $K_1 = (2,1)$ $K_2 = (2,3)$

Step 2: finding the distance b/w the cluster centers and each data points. Euclidean dist $= \sqrt{(x_1-y_1)^2+(x_2-y_2)^2}$

Data point	Distance from K1 (2,1)	Distance from K2 (2,3)	Assigned cluster
a1 (1,1)	1	2.24	K1
a2 (2,1)	0	2	K1
a3 (2,3)	2	0	K2
a4 (3,2)	1.41	1.41	K1
a5 (4,3)	2.83	2	K2
a6 (5,5)	5	3.61	K2

we have to find distance b/w each center and given data point $(x_1, x_2) = (1,1)$ $(y_1, y_2) = (2,1)$ for cluster 1

Euclidean distance $= \sqrt{(x_1-y_1)^2+(x_2-y_2)^2}$
$= \sqrt{(1-2)^2+(1-1)^2}$
$= \sqrt{1} = 1$ and so on

Step 3: cluster 1 of $K_1 = \{a1, a2, a4\}$
cluster 2 of $K_2 = \{a3, a5, a6\}$

Step 4: Recalculate the cluster centers.

$K_1 = \frac{1}{3}[(1,1)+(2,1)+(3,2)] = \frac{1}{3}(6,4) = (2, 1.33)$

$K_2 = \frac{1}{3}[a3+a5+a6] \Rightarrow \frac{1}{3}[(2,3)+(4,3)+(5,5)]$

$K_2 = \frac{1}{3}(11,11) = (3.67, 3.67)$

K-Means clustering

* k - no of cluster/groups

Clustering: dividing the dataset into diff groups.

Clustering: Clustering is the task of dividing a dataset into groups (clusters) such that data points in the same group are more similar to each other than to those in other groups. It is used when we don't have labeled data.

What is K-Means clustering?

ans ⇒ K-Means clustering is an unsupervised machine learning algorithm which groups the unlabeled dataset into different clusters. K-means clustering is a technique used to organize data into groups based on their similarity.

Example: Online store uses k-means to group customers based on purchase frequency and spending creating segments like Budget shoppers, frequent buyers and Big spenders for personalized marketing.

The algorithm works by first randomly picking some central points called 'centroids' and each data point is then assigned to the closest centroid forming a cluster.

The algorithm will categorize the items into k-group or clusters of similarity. To calculate that similarity, we will use the Euclidean distance as a measurement.

. Working of K-means clustering

1. First, we randomly initialize k points, called means or cluster centroids.
2. We categorize each item to its closet closest mean, and we update the mean's coordinates, which are the averages of the items categorized in that cluster so far.

k=2
3

Q. calculate the probability of pass for the student who studied 33 hr.

Hours study	pass(1) / Fail(0)
29	0
15	0
33	1
28	1
39	1

$\log(odds) = z = -64 + 2 * hours.$

(i) calculate the probability of pass for the student who studied 33 hours.

(ii) At least how many hours student should study that makes he will pass the course with the probability of more than 95%.

Soln: (i) $y = \frac{1}{1+e^{-z}}$ $\Rightarrow$ $y = \frac{1}{1+e^{-z}}$

$z = -64 + 2 * hr$ $\quad z = -64 + 2 * 33$

$\boxed{z = 2}$

$p = \frac{1}{1+e^{-2}}$ $\quad \boxed{p = 0.88}$

Conclusion: If student studied 33 hours; then there is 88% chance that the student will pass the exam.

(ii) $p = \frac{1}{1+e^{-z}} = 0.95$

$0.95 * (1+e^{-z}) = 1$

$0.95 * e^{-z} = 1 - 0.95$

$e^{-z} = \frac{0.05}{0.95}$

$e^{-z} = 0.0526$

$\ln(e^{-z}) = \ln(0.0526)$

$\ln(e^{x}) = x$

$-z = \ln(0.0526) = -2.94$

$\boxed{z = 2.94}$

$\log(odds) =$

$z = -64 + 2 * hr$

$2.94 = -64 + 2 * hr$

$\boxed{hr = 33.47 hr}$

Naive Bayes ML-2

def ⇒ Naive Bayes classifiers are supervised machine learning algorithms used for classification tasks, based on Bayes theorem to find probabilities. It is named as 'naive' because it assumes the presence of one feature does not affect other features. The 'Bayes' part of the name refers to the basis in Bayes theorem.

$$P(A|B) = \frac{P(A) + P(B|A)}{P(B)}$$

Features of Naive Bayes

(i) Based on Bayes Theorem: Naive Bayes Theorem uses Bayes Theorem to compute the probability of each class given a set of input feature.

(ii) Naive Assumption of feature independence: It assume all features are independent of each other, given the class label.

(iii) Simple and fast: Naive Bayes is a simple and powerful algorithm.

(iv) Lower number of parameters: It requires fewer parameters which makes it efficient and easy to train.

(v) Application: Spam filtering, Recommendation system.

Probability

(1) Independent Event

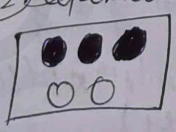
Rolling a Dice
{1, 2, 3, 4, 5, 6}

$P(1) = \frac{1}{6}$ $P(2) = \frac{1}{6}$

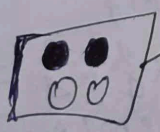
$P(3) = \frac{1}{6}$

Tossing a coin $= \frac{1}{2}$

(2) Dependent Event



$\rightarrow P(B) = \frac{3}{5}$



$\rightarrow P(B) = \frac{2}{4}$

$Pr(B) = \frac{3}{5} \times \frac{2}{4}$

Height(CM)	Weight(KG)	class	Distance
169	58	Normal	1.4
170	55	Normal	2
173	57	Normal	3
174	56	underweight	4.1
167	51	underweight	6.7
173	64	normal	7.6
172	65	normal	8.2
182	62	Normal	13
176	69	Normal	13.4
170	57	?	

if k=1 Normal
k=2 Normal
k=3 = Normal

k=4 underweight but majority is = Normal

k=5 = Normal

? = Normal

Step1: calculate distances

distance to (8.0, 160) = $\sqrt{(7.4-8)^2+(114-160)^2}$ = 46

distance to (6.2, 160) = $\sqrt{(7.4-6.2)^2+(114-170)^2}$ = 56.01

" to (7.2, 168) = $\sqrt{(7.4-7.2)^2+(114-168)^2}$ = 54.00

" to (8.2, 155) = $\sqrt{(7.4-8.2)^2+(114-155)^2}$ = 41.00

Step2: Select k Nearest Neighbours

k=1

least distance (41)

41 = dist (8.2, 155)

OMG2 = Comedy

k=3

41, 46, 54

(OMG2) (MI) (R&R)

(Comedy) (Action) (Comedy)

Step3: Majority voting → Comedy

Hence, Barbie movie with IMDb rating 7.4 and duration 114 minutes, its genre is "Comedy"

Q.

Height (cm)	weight (kg)	class
167	51	underweight
182	62	Normal
176	69	Normal
173	64	Normal
172	65	Normal
174	86	underweight
169	58	Normal
173	57	Normal
170	55	Normal
170	57	?

Height (cm)

169

170

173

Q. Estimate conditional probabilities of each attribute (color, legs, height, smelly) for the species classes {M, H} using the data given in the table.
use these probabilities estimate the probability value for the new instance (color = Green, legs = 2, Height = Tall and Smelly = No)

No	color	Legs	Height	Smelly	Species
1	white	3	short	yes	M
2	Green	2	Tall	NO	M
3	Green	3	short	yes	M
4	white	3	short	yes	M
5	Green	2	short	NO	H
6	white	2	Tall	NO	H
7	white	2	Tall	NO	H
8	white	2	short	yes	H

$P(M) = \frac{4}{8} = 0.5$, $P(H) = \frac{4}{8} = 0.5$

New Instance
(color = Green, legs = 2, Height = Tall, smelly = No)

color	M	H
white	2/4	3/4
Green	2/4	1/4

legs	M	H
2	1/4	4/4
3	3/4	0/4

Height	M	H
Tall	3/4	2/4
short	3/4	2/4

Smelly	M	H
yes	3/4	1/4
NO	1/4	3/4

$P(M|\text{New Instance}) = P(M) * P(\text{color} = \text{Green}|M) * P(\text{Legs} = 2|M) * P(\text{Height} = \text{tall}|M) * P(\text{Smelly} = \text{no}|M)$

$P(M|\text{New Instance}) = 0.5 * \frac{2}{4} * \frac{1}{4} * \frac{2}{4} * \frac{1}{4} = 0.00390$

$P(H|\text{New Instance}) = P(H) * P(\text{color} = \text{Green}|H) * P(\text{Legs} = 2|H) * P(\text{Height} = \text{tall}|H) * P(\text{smelly} = \text{no}|H)$

$P(H|\text{New Instance}) = 0.5 \times \frac{1}{4} \times \frac{4}{4} \times \frac{2}{4} \times \frac{3}{4} = 0.0468$

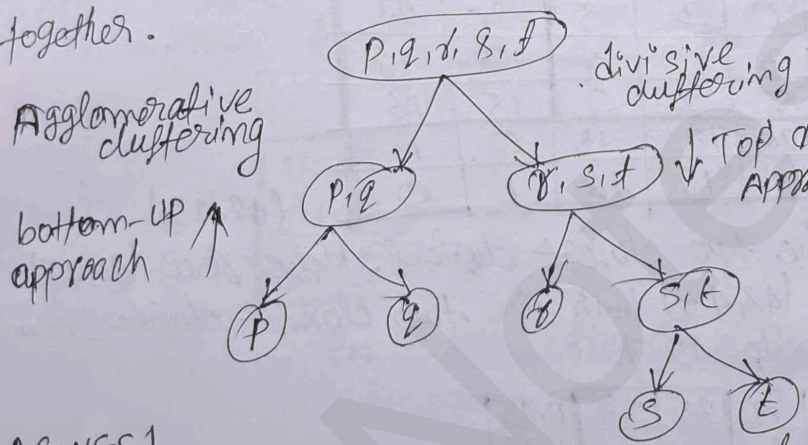
$\because P(H|\text{New Instance}) > P(M|\text{New Instance})$

Hence the new Instance belongs to species H

Clustering

- Hierarchical clustering
  - → Agglomerative clustering
  - → Divisive clustering
- partitioning clustering
  - → K-Means clustering
  - → K-Mode clustering

Hierarchical clustering is a technique used to group similar data points together based on their similarity ~~their s~~ creating a hierarchy or tree-like structure. A dendrogram is like a family tree for clusters. It shows how individual data points or groups of data merge together.



(AGNES)
Agglomerative clustering: Agglomerative clustering is a type of hierarchical clustering that builds clusters by merging them successively. It is a bottom-up approach, meaning

1. Each data points starts in its own cluster.
2. The algorithm repeatedly merges the two closest clusters until all points are in a single cluster.

(DIANA)
Divisive cluster: Divisive clustering is the opposite of Agglomerative clustering - it is a top-down hirarchical clustering approach.

Iteratively compare the cluster (leaders) data points to each of the observations. similar data points given 0; dissimilar data points give 1

Comparing leader p1 to the observation p2

blone	amber	fair	
brunette	gray	brown	
+1	+1	+1	= 1+1+1 = 3

like wise calculate all the dissimilarities and put them in a matrix..

1.1

	cluster 1 (P1)	cluster 2 (P4)	cluster 3 (P8)	cluster
P1	0 ✓ (minimum)	2	2	cluster 1
P2	3 ✓	3	3	cluster 1
P3	3	1 ✓	3	cluster 2
P4	3	3	1 ✓	cluster 3
P5	1 ✓	2	2	cluster 1
~~P6~~	3	3	2 ✓	cluster 3
~~P7~~	2	0 ✓	2	cluster 2
P8	2	2	0 ✓	cluster 3

After step 2; the observations P1, P2, P5 are assigned to cluster 1, P3, P7 to cluster 2 and P4, P6, P8 are assigned to cluster 3

✓ → minimum in row

Same → any 1st

Step 3: Define new modes for the clusters

	hair color	eye color	skin color
cluster 1	brunette	amber	fair
cluster 2	red	green	fair
cluster 3	black	hazel	brown

Repeat ~~step~~ steps 2 - 4

Q. Imagine we have a dataset that has the information about hair color, eye color and skin color of persons, we aim to group them based on the available information.

Person	hair color	eye color	skin color
P1	blonde	amber	fair
P2	brunette	gray	brown
P3	red	green	brown
P4	black	hazel	brown
P5	brunette	amber	fair
P6	black	gray	brown
P7	red	green	fair
P8	black	hazel	fair

Step 1: Pick k observations at random and use them as leaders / clusters. choosing k=3 P1, P7, P8 as leaders

Leaders

P1	blonde	amber	fair
P7	red	green	fair
P8	black	hazel	fair

person	hair color	eye color	skin color
P1	blonde	amber	fair
P2	brunette	gray	brown
P3	red	green	brown
P4	Black	hazel	brown
P5	brunette	amber	fair
P6	black	gray	brown
P7	red	green	fair
P8	black	hazel	fair

Step 2: Calculate the dissimilarities (no of mismatches) and assign each observation to its closest cluster.

	18	22	25,27	42,43
18	0	4	7	24
22	4	0	3	20
25,27	7	3	0	15
42,43	24	20	15	0

(42,43), ((25,27),

Step 4: Repeate until!
The distance b/w 22 and (25,27) is minimum so merge (25,27) into 22.

	18	22,25,27	42,43
18	0	4	24
22,25,27	4	0	20
42,43	24	20	0

((42,43),
((25,27),22))

Step 5: The distance b/w (22,25,27) and 18 is minimum so merge (22,25,27) into 18.

	18,22,25,27	42,43
18,22,25,27	0	20
42,43	20	0

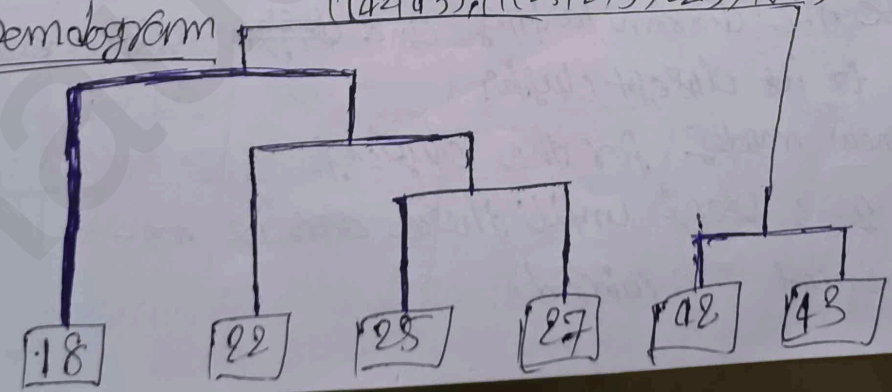
((42,43),
(((25,27),22),18))

Step 6: The distance b/w (42,43) and (18,22,25,27) is minimum so merge (42,43) into (18,22,25,27)

	18,22,25,27,42,43
18,22,25,27,42,43	0

((42,43),(((25,27),22),18))

- Dendogram



Q. Consider the following set of 6 one dimensional data points: 18, 22, 25, 27, 42, 43. Apply the

(i) agglomorative clustering algorithm to build the hierarchical clustering dendogram.

(ii) merge the clusters using Min distance and update the proximity matrix accordingly

(iii) clearly show the proximity matrix corresponding to each iteration of the algorithm.

Soln:

Step 1: Compute distance between all pairs of clusters

	18	22	25	27	42	43
18	0	4	7	9	24	25
22	4	0	3	5	20	21
25	7	3	0	2	17	18
27	9	5	2	0	15	16
42	24	20	17	15	0	1
43	25	21	18	16	1	0

(42,43)

Step 2: Merge the two closest clusters: Here row (42,43) and column (42,43) both are two closest clusters hence merge 43 into 42 here.

	18	22	25	27	42,43
18	0	4	7	9	24
22	4	0	3	5	20
25	7	3	0	2	17
27	9	5	2	0	15
42,43	24	20	17	15	0

(42,43)

Step 3: update the distance matrix: The distance b/w 27 and 25 is minimum so merge 25 and 27 making cluster. Merge 27 into 25.

Self-organizing Maps (SOMs)

Self-organizing Maps (SOMs) also known as Kohonen maps, are a type of unsupervised neural network used for dimensionality reduction, clustering and data visualization. They were invented by Finnish professor Teuvo Kohonen in the 1980s. It trained its network through a competitive learning algorithm. SOM is used for clustering and mapping techniques to map multidimensional data onto lower-dimensional which allows people to reduce complex problems for easy interpretation. SOM has two layers, one is the input layer and the other one is the output layer.

pyq. Discuss the various steps to create a self organized map

Architecture of SOM

The SOM consists of two layers

(i) Input layer: Each neuron corresponds to a feature in the input vector.

(ii) Output layer: Usually a 2D grid (eg. 10x10) of neurons. (Map/Grid).

Each node (neuron) in the output layer has a weight vector of the same dimension as the input vector.

Q Consider the network shown in figure which considers four training samples each vector of length 4 and two output units. Train the SOFM network by determining the class membership of the input data

Training samples;

$x1 = (1,0,1,0)$ $x2 = (1,0,0,0)$

$x3 = (1,1,1,1)$ $x4 = (0,1,1,0)$

New leaders

	hair color	eye color	skin color
cluster 1	brunette	amber	fair
cluster 2	red	green	fair
cluster 3	black	hazel	brown

person	hair color	eye color	skin color
P1	blonde	amber	fair
P2	brunette	gray	brown
P3	red	green	brown
P4	black	hazel	brown
P5	brunette	amber	fair
P6	black	gray	brown
P7	red	green	fair
P8	black	hazel	fair

Compare cluster 1 to the observation P1 gives 1 dissimilarity.
Likewise, calculate all the dissimilarities and put them in a matrix. Assign each observations to its closest cluster.

@ 1.2

	cluster 1	cluster 2	cluster 3	cluster
P1	1 ✓	2	3	cluster 1
P2	2 ✓	3	2	cluster 1
P3	3	1 ✓	2	cluster 2
P4	3	3	0 ✓	cluster 3
P5	0 ✓	2	3	cluster 1
P6	3	3	1 ✓	cluster 3
P7	2	0 ✓	3	cluster 2
P8	2	2	1 ✓	cluster 3

Stop because 1.1 == 1.2 diagram

The observations P1, P2, P5 are assigned to cluster 1, P3, P7 are assigned to cluster 2, and P4, P6, P8 are assigned to cluster 3 we stop here as we see here there is no change in the assignment of observations.

K-Mode clustering

K-Mode is an one of the unsupervised algorithm used to cluster categorical variables unlike traditional clustering algorithm that uses distance metrics. K Modes works by identifying the modes or most frequent values within each cluster to determine its centroid. KModes is ideal for clustering categorical data such as customer demographics, market segments or survey responses. It is a powerful tool for data analysts and scientists to gain insights into their data and make informed decisions.

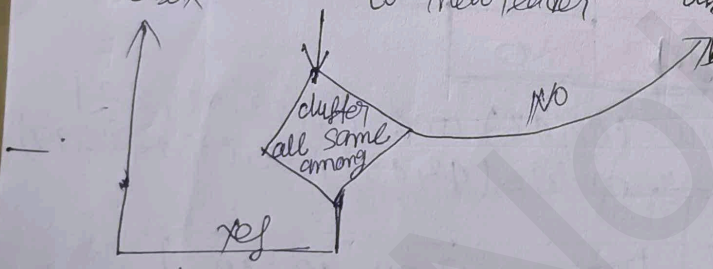
Process Set 'K' → Assign 'K' leaders → Iterate among 'p' →

Calculate mode of each cluster → Assign new mode to new leader

pickup random instance or leaders

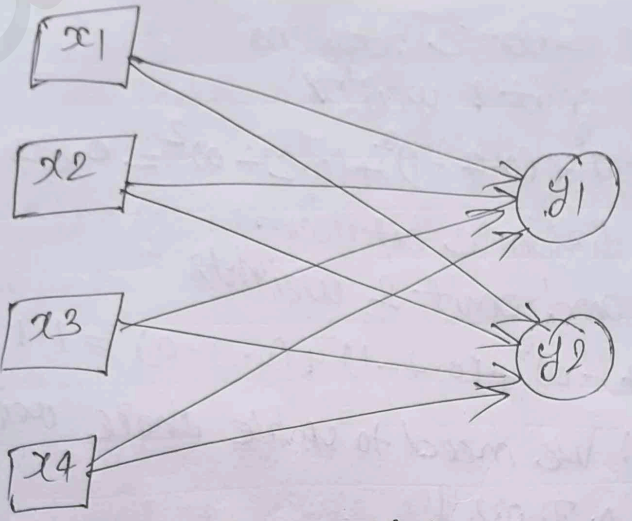
Identify leaders who is closer to the instance and assign (Identity dissimilarity new leaders)

Consecutive ebox



Working of K Modes clustering

1. Pick K observations at random and use them as cluster leaders.
2. Calculate the dissimilarities and assign each observation to its closest cluster
3. Define new modes for the clusters
4. Repeate 2-3 steps until there are is no-reassignement required.



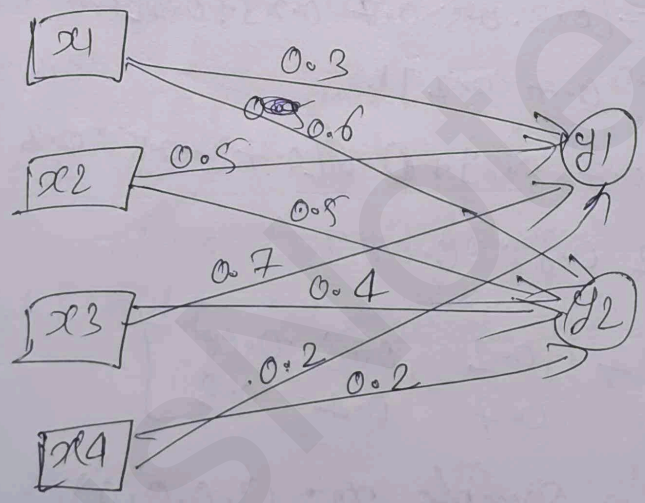
output units: unit 1, unit 2

Learning rate $\eta(t) = 0.6$

initial weight matrix

$$\begin{bmatrix} \text{unit 1} \\ \text{unit 2} \end{bmatrix} = \begin{bmatrix} 0.3 & 0.5 & 0.7 & 0.2 \\ 0.6 & 0.5 & 0.4 & 0.2 \end{bmatrix}$$

Ans ⇒



Iteration 1: Training Sample $x_1 : (1,0,1,0)$

weight matrix:

$$\begin{bmatrix} \text{unit 1} \\ \text{unit 2} \end{bmatrix} : \begin{bmatrix} 0.3 & 0.5 & 0.7 & 0.2 \\ 0.6 & 0.7 & 0.4 & 0.3 \end{bmatrix}$$

Compute Euclidean distance between
X1: [1,0,1,0] and unit 1 weights

$d^2 = (0.3-1)^2 + (0.5-0)^2 + (0.7-1)^2 + (0.2-0)^2 = 0.87$

Compute Euclidean distance between
x1: [1,0,1,0] and unit 2 weights

$d^2 = (0.6-1)^2 + (0.7-0)^2 + (0.4-1)^2 + (0.3-0)^2 = 1.1$

unit 1 wins now we need to update those weights
Here [0.3, 0.5, 0.7, 0.2]

$$w_j(t+1) = w_j(t) + \eta(t)(x_s - w_j(t))$$

$w_j(t+1)$: New weights; $w_j(t)$: old weights; $\eta(t)$: learning rate; x_s: input; $w_j(t)$: old weights

update the weights of the winning unit

New unit 1 weight = [0.3, 0.5 0.7 0.2] + 0.6([1 0 1 0] - [0.3 0.5 0.7 0.2]).

= [0.3 0.5 0.7 0.2] + 0.6([0.7 −0.5 0.3 −0.2])

= [0.72 0.2 0.88 0.08]

$$\begin{bmatrix} \text{unit 1} \\ \text{unit 2} \end{bmatrix} : \begin{bmatrix} 0.72 & 0.2 & 0.88 & 0.08 \\ 0.6 & 0.7 & 0.4 & 0.3 \end{bmatrix}$$

Iteration 2: Training Sample x_2: [1,0,0,0)

weights Matrix

$$\begin{bmatrix} \text{unit 1} \\ \text{unit 2} \end{bmatrix} : \begin{bmatrix} 0.72 & 0.2 & 0.88 & 0.08 \\ 0.6 & 0.7 & 0.4 & 0.3 \end{bmatrix}$$

Compute Euclidean distance between x_2: (1,0,0,0) and unit 1 weights $d^2 = (0.72-1)^2 + (0.2-0)^2 + (0.88-0)^2 +$

$x_1 : (1,0,1,0) \rightarrow$ unit 1
$x_2 : (1,0,0,0) \rightarrow$ unit 1
$x_3 : (1,1,1,1) \rightarrow$ unit 2
$x_4 : (0,1,1,0) \rightarrow$ unit 2

} Epoch

This process is continued for many epochs until the feature map does not change.

DIANA - Divisive Analysis clustering

Q. Given the dataset (a, b, c, d) and the distance matrix in table, Apply divisive analysis clustering algorithm (DIANA) to form clusters.

	a	b	c	d	e
a	0	9	3	6	11
b	9	0	7	5	10
c	3	7	0	9	2
d	6	5	9	0	8
e	11	10	2	8	0

Soln:

→ Step 1 : initially $C_1 = \{a, b, c, d, e\}$

Step 2 : $C_i = C_1$ and $C_j = \phi$

Step 3 : Initial Iteration

Let us calculate the average dissimilarities of the objects in C_i with the other objects in C_i.

Average dissimilarity of a

$$a = \frac{1}{4} * (d(a,b) + d(a,c) + d(a,d) + d(a,e))$$

$$a = \frac{1}{4}(9+3+6+11) = 7.25$$

$= [0.6 \cdot 0.7 \quad 0.4 \quad 0.3] + 0.6[0.4 \; 0.3 \quad 0.6 \quad 0.7]$

$= [0.84 \; 0.88 \quad 0.76 \quad 0.72]$

$\begin{bmatrix} \text{unit 1} \\ \text{unit 2} \end{bmatrix} = \begin{bmatrix} 0.89 & 0.08 & 0.35 & 0.03 \\ 0.84 & 0.88 & 0.76 & 0.72 \end{bmatrix}$

Iteration 4:

Training sample $x4 : (0, 1, 1, 0)$

weight matrix

$\begin{bmatrix} \text{unit 1} \\ \text{unit 2} \end{bmatrix} = \begin{bmatrix} 0.89 & 0.08 & 0.35 & 0.03 \\ 0.84 & 0.88 & 0.76 & 0.72 \end{bmatrix}$

Compute Euclidean distance b/w $x4 : (0, 1, 1, 0)$ and unit 1 weights

$d^2 = (0.89 - 0)^2 + (0.08 - 1)^2 + (0.35 - 1)^2 + (0.03 - 0)^2 = 2.01$

Compute Euclidean distance b/w $x1 : (0, 1, 1, 0)$ and unit 2 weights $d^2 = (0.84 - 0)^2 + (0.88 - 1)^2 + (0.76 - 1)^2 + (0.72 - 0)^2$

$= 1.3$

$w_j(t+1) = w_j(t) + \eta(t)(x_s - w_j(t))$

unit 2 wins

update the weights of the winning unit:

New unit 2 weights:

$[0.84 \; 0.88 \; 0.76 \; 0.72] + 0.6[0 \quad 1 \quad 1 \quad 0] - [0.84 \; 0.88 \; 0.76 \; 0.72]$

$= [0.34 \; 0.95 \quad 0.9 \quad 0.29]$

$\begin{bmatrix} \text{unit 1} \\ \text{unit 2} \end{bmatrix} = \begin{bmatrix} 0.89 & 0.08 & 0.35 & 0.03 \\ 0.34 & 0.95 & 0.9 & 0.29 \end{bmatrix}$

Best mapping units for each of the sample taken are

$+(0.08-0)^2 = 0.74$

Compute Euclidean distance between $x_2 : (1,0,0,0)$ and unit 2 weights

$d^2 = (0.6-1)^2 + (0.7-0)^2 + (0.4-0)^2 + (0.3-0)^2 = 0.9$

unit 1 wins

$$w_j(t+1) = w_j(t) + \eta(t)(x_s - w_j(t))$$

update the weights of the winning unit:

New unit 1 weights $= [0.72 \;\; 0.2 \;\; 0.88 \;\; 0.08] +$

$0.6[[1, 0 \;\; 0 \;\; 0] - [0.72 \;\; 0.2 \;\; 0.88 \;\; 0.08]$

$= [0.89 \;\; 0.08 \;\; 0.35 \;\; 0.03]$

$$\begin{bmatrix}\text{unit 1}\\ \text{unit 2}\end{bmatrix} : \begin{bmatrix}0.89 & 0.08 & 0.35 & 0.03\\ 0.6 & 0.7 & 0.4 & 0.3\end{bmatrix}$$

Iteration 3:

Training sample $x_3 : (1,1,1,1)$

weight matrix:

$$\begin{bmatrix}\text{unit 1}\\ \text{unit 2}\end{bmatrix} : \begin{bmatrix}0.89 & 0.08 & 0.35 & 0.03\\ 0.6 & 0.7 & 0.4 & 0.3\end{bmatrix}$$

Compute Euclidean distance between $x_3 : (1,1,1,1)$ and unit 1 weights

$d^2 = (0.89-1)^2 + (0.08-1)^2 + (0.35-1)^2 + (0.03-1)^2$

$= 2.2$

Compute Euclidean distance b/w $x_3 : (1,1,1,1)$ and unit 2 weights $d^2 = (0.6-1)^2 + (0.7-1)^2 + (0.4-1)^2 + (0.3-1)^2$

unit 2 wins. $w_j(t+1) = w_j(t) + \eta(t)(x_s - w_j(t))$

update the weights of the winning unit:

New unit 2 weights $= [0.6 \;\; 0.7 \;\; 0.4 \;\; 0.3] + 0.6([1 \;\; 1 \;\; 1 \;\; 1] -$

$[0.6 \;\; 0.7 \;\; 0.4 \;\; 0.3])$.

Similarly we have

Average dissimilarity of a = 7.25

" " b = 7.75

" " c = 5.25

" " d = 7.00

" " e = 7.75

The highest average distance is 7.75 and there are two corresponding objects.

We choose one of them, b, arbitrarily.

We move b to Cj.

We now have $C_i = \{a, c, d, e\}$ and $C_j = \phi \cup \{b\} = b$

Step 3: Remaining iteration $C_i = \{a, c, d, e\}$ and $C_j = \{b\}$

(i) 2nd iteration

$Da = \frac{1}{3}(d(a,c) + d(a,d) + d(a,e)) - \frac{1}{1}(d(a,b)) = \frac{20}{3} - 9 = -2.33$

$Dc = \frac{1}{3}(d(c,a) + d(c,d) + d(c,e)) - \frac{1}{1}(d(c,b)) = \frac{14}{3} - 7 = -2.33$

$Dd = \frac{1}{3}(d(d,a) + d(d,c) + d(d,e)) - \frac{1}{1}(d(d,b)) = \frac{23}{3} - 7 = 0.67$

$De = \frac{1}{3}(d(e,a) + d(e,c) + d(e,d)) - \frac{1}{1}(d(e,b)) = \frac{21}{3} - 7 = 0$

Dd is the largest and $Dd > 0$

So we move d to Cj

We now have $C_i = \{a, c, e\}$ and $C_j = \{b\} \cup \{d\} = \{b, d\}$

(ii) 3rd iteration

$Da = \frac{1}{2}(d(a,c) + d(a,e)) - \frac{1}{2}(d(a,b) + d(a,d))$

$Da = \frac{14}{2} - \frac{15}{2} = -0.5$

$Dc = \frac{1}{2}[d(c,a) + d(c,e)) - \frac{1}{2}(d(c,b) + d(c,d))$

$Dc = \frac{5}{2} - \frac{16}{2} = -13.5$

of a weighted sum of the inputs, which then passed through an activation function.

The weighted sum =

$$z = w_1 x_1 + w_2 x_2 + \cdots + w_n x_n = x^T w$$

The step function compares this weighted sum to a threshold. if the input is larger than the threshold value. the output is 1: otherwise, it is 0.

$$h(z) = \begin{cases} 0 & \text{if } z < \text{Threshold} \\ 1 & \text{if } z \geq \text{Threshold} \end{cases}$$

A perceptron consists of a single layer of Threshold Logic units (TLU) with each (TLU) fully connected to all input nodes. The output of the fully connected layer is computed as:

$$f_{w,b}(x) = h(xW + b)$$

where x is the input w is the weight for each inputs neurons and b is the bias and h is the step function.

The weight update formula is

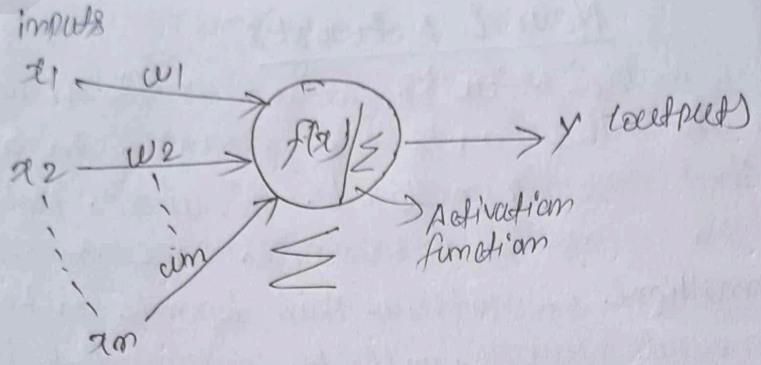
$$w_{i,j} = w_{i,j} + \eta (y_j - \hat{y}_j) x_i$$

where $w_{i,j}$ is the weight b/w i^{th} inp and j^{th} o/p neuron.

x_i is the i^{th} inp value

y_j is the actual value, and $\hat{y}_j$ is the predicted value.

η is the learning rate



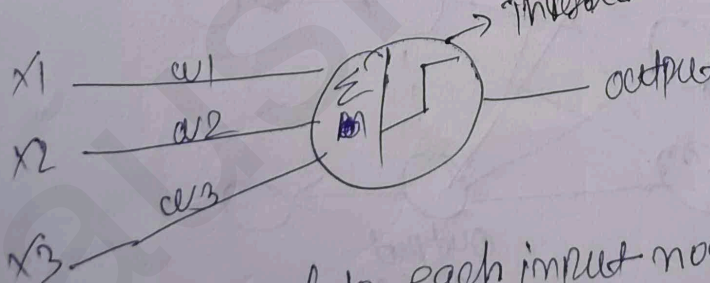
(Q5) what is perceptron? How does a perceptron work in Neural networks?

ans → perceptron is a type of neural network that performs binary classification that maps input features to an output decision, usually classifying data into one of two categories such as 0 or 1.

perceptron consists of a single layer of input nodes that are fully connected to a layer of output nodes. It is particularly good at learning linearly seprable patterns. It utilizes a variation of artificial neurons called Threshold Logic units (TLU).

Types of perceptron (i) single layer perceptron
(ii) multi-layer perceptron

How does perceptron work!



A weight is assigned to each input node of a perceptron, indicating the importance of that input in determining the output. The perceptron output is calculated

$De = \frac{1}{2}(d(e,a)+d(e,c)) - \frac{1}{2}(d(e,b)+d(e,d))$

$De = \frac{13}{2} - \frac{18}{2} = -2.5$

None of them are +ve here

All are negative, so we stop and form the clusters c_i and c_j.

$c_i = \{a, c, e\}$ and $c_j = \{b, d\}$

Step 4:

To divide c_i and c_j, we compute their diameters.

diameter $(c_i) = \max\{d(a,c), d(a,e), d(c,e)\}$

diameter $(c_i) = \max(3, 11, 2) = 11$

diameter $(c_j) = \max(d(b,d)) = 5$

The cluster with the large diameter is c_i.

So we now split c_i.

We repeat the process by taking $c_i = \{a, c, e\}$

$\therefore c_i = \{a, c, e\}$, $c_j = \phi$ — iteration-1

$Da = \frac{1}{2}(d(a,c)+d(a,e)) = \frac{1}{2}(14) = 7$

$Dc = \frac{1}{2}(d(c,a)+d(c,e)) = \frac{1}{2} \times 5 = 2.5$

$De = \frac{1}{2}(d(e,a)+d(e,c)) = \frac{1}{2} \times 13 = 6.5$

Da is the maximum so we move a to c_j.

Now $c_i = \{c, e\}$, $c_j = \{a\}$

iteration 2

$Dc = \frac{1}{1}(d(c,e)) - \frac{1}{1}(d(c,a)) = 2 - 3 = -1$

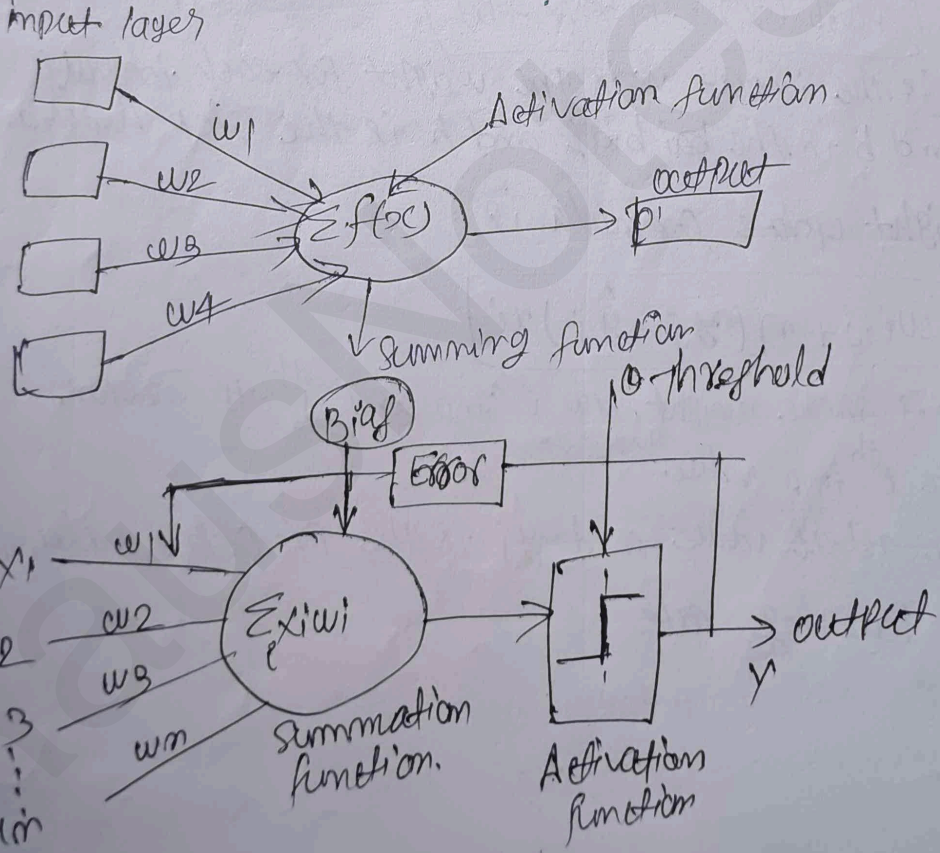
$De = \frac{1}{1}(d(e,c)) - \frac{1}{1}(d(e,a)) = 2 - 11 = -9$

Since None of them are +ve here

All are negative, so we stop and from the clusters c_i and c_j

* Single Layer Perceptron : Single layer perceptron is inspired by biological neurons and their ability to process information. To understand the SLP we first need to break down the workings of a single artificial neuron which is the fundamental building block of neural networks. An Artificial neuron is a simplified computational model that mimics the behavior of a biological neuron. It takes inputs, processes them and produces an output.

- Receive signal from outside.
- Process the signal and decide whether we need to send information or not.
- Communicate the signal to the target cell, which can be another neuron.



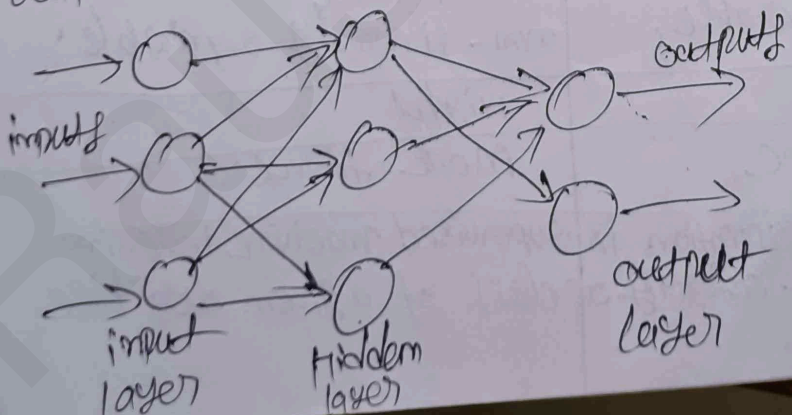
* Multi-layer perceptron: Multi-layer Perceptron is an artificial neural network widely used for solving classification and regression tasks. MLP transforms input data from one dimension to another. It is called "multi-layer" because it contains an input layer, one or more hidden layers, and an output layer. The purpose of an MLP is to model complex relationships b/w inputs and outputs, making it a powerful tool for various machine learning tasks.

Key Components of Multi-layer Perceptron

(i) Input layer: Each neuron in this layer corresponds to an input feature. ~~the input~~ for instance, if you have three input features, the input layer will have three neurons.

(ii) Hidden layers: An MLP can have any no of hidden layers, with each layer containing any number of nodes. These layers process the information received from the input layer

(iii) Output layer: The output layer generates the final prediction or result. If there are multiple outputs, the output layer will have a corresponding no of neurons



* Difference b/w Multiclass and Multi-label

features	Multi-class classification	Multi-label classification
Definition	In multi-class classification, each instance is classified into one class out of more than two possible classes.	In multi-label classification each instance can be assigned multiple classes.
Example	classifying an image as cat, dog	classifying a movie into action, comedy, thrilles.
Exclusion	The classes are mutually exclusive.	The classes are not mutually exclusive
Label representation	one-hot-encoded vector	multi-hot encoded binary vector
loss Function	Categorical Cross-Entropy	Binary Cross-Entropy

* Diff b/w Linear and Non-linear classification

features	Linear classification	Non-Linear classification
Definition	A classification that uses a straight line to seprate data points into classes.	A classification that uses curve to seprate data points into classes
Decision Boundary	Straight line	curved plane
Example	Logistic Regression	Decision Trees
Work well	works well when data is linearly separable.	works well when data is non-linearly separable.
Complexity	Low	High
Training Time	Less Time	More time
Define	classification: classification is algorithm used to predict category or class of a given data base on past labeled data.	supervised machine learning