Savitribai Phule Pune University
Semester VII (Computer Engineering)

Machine Learning (Code : 410242)

Chapter 3 : Supervised Learning : Regression

Q. 1 Define regression analysis. **(2 Marks)**

OR Define dependent and independent variables with a suitable example. **(4 Marks)**

Ans. :

- **Definition :** A functional relationship between two or more correlated variables that is often empirically determined from data and is used specially to predict values of one variable when given values of the others.
- Regression analysis aims to establish a relationship (or influence) of a set of variables on another variable (outcome).
- **Definition :** The variables that have influence on the outcome are called input, independent, explanatory or predictor variable.
- **Definition :** The variable whose outcome (or value) depends on other variables is called response, outcome, or dependent variable.

Q.2 What is linear regression? **(4 Marks)**

Ans. :

- **Definition :** The process of finding a straight line that best approximates a set of points on a graph is called linear regression.

There are two types of linear regression.

1. **Simple Linear Regression (SLR)**

   This has only one independent (or input) variable.

2. **Multiple Linear Regression (MLR)**

   This has more than one independent (or input) variables.

The general formula (or model) for linear regression analysis is

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \ldots \beta_n X_n + \varepsilon$$

Where,

- Y is the outcome (dependent) variable
- X_i are the values of independent variables
- β_0 is the value of Y when each X_i is equal to 0. It is also called as y-intercept
- β_i is the change in Y based on the unit change in X_i. It is also called as regression coefficient or slope of the regression line.
- ε is the random error or noise that represents the difference between the predicted value (based on the regression model) and actual value.

The goal is to find the regression line that best approximates (or fits) the relation between the input variable and output variable. The objective is to find a linear model where the sum of squares of the distances is minimal.

Q. 3 For the following data set, find the linear regression line. Predict the value of Y if X = 10. **(6 Marks)**

X	Y
0	1
1	3
2	2
3	5
4	7
5	8
6	8
7	9
8	10
9	12

Ans. :

X	Y	$X_i - \bar{X}$	$Y_i - \bar{Y}$	$(X_i - \bar{X}) \times (Y_i - \bar{Y})$	$(X_i - \bar{X})^2$
0	1	-4.5	-5.5	24.75	20.25
1	3	-3.5	-3.5	12.25	12.25
2	2	-2.5	-4.5	11.25	6.25
3	5	-1.5	-1.5	2.25	2.25
4	7	-0.5	0.5	-0.25	0.25
5	8	0.5	1.5	0.75	0.25
6	8	1.5	1.5	2.25	2.25
7	9	2.5	2.5	6.25	6.25
8	10	3.5	3.5	12.25	12.25
9	12	4.5	5.5	24.75	20.25
$\bar{X}$(Mean of X) = 4.5	$\bar{Y}$(Mean of Y) = 6.5			Total = 96.5	Total = 82.5

Hence,
$$\beta_1 = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2}$$

$$\beta_1 = \frac{96.5}{82.5} = 1.1696$$

$$\beta_0 = \bar{y} - \beta_1 \bar{x}$$

$$\beta_0 = 6.5 - 1.1696 \times 4.5 = 1.2368$$

Hence, the regression line is

$$Y = 1.2368 + 1.1696X$$

A sample plot of the regression line is as shown in Fig. 3.1.

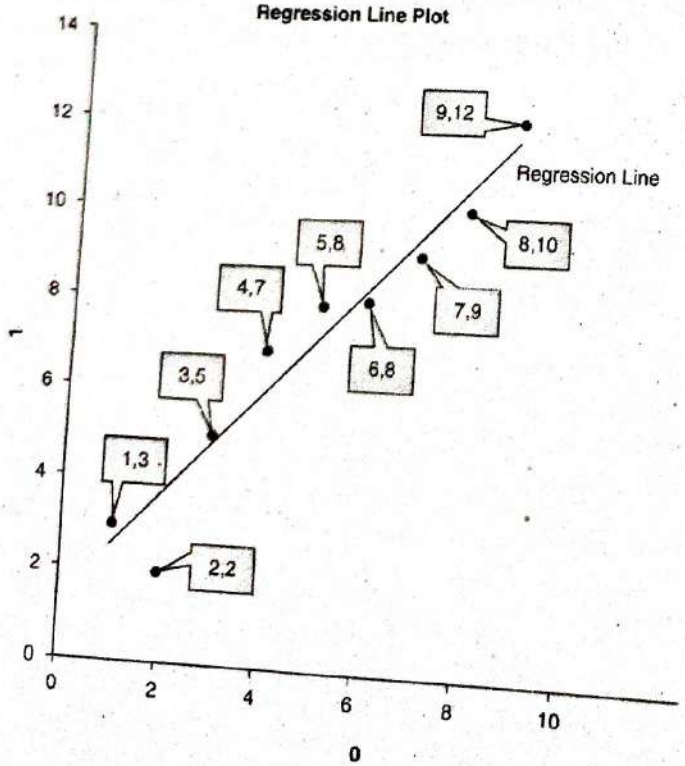


Fig. 3.1

To predict the value of Y when X = 10, put the value of X in the regression line. You get

$Y = 1.2368 + 1.1696 \times 10 = 12.932$ or 13 when rounded.

Q.4 A clinical trial gave the following data for BMI and Cholesterol level for 10 patients. Predict the likely value of cholesterol level for someone who has a BMI of 27. **(6 Marks)**

BMI	Cholesterol
17	140
21	189
24	210
28	240
14	130
16	100
19	135
22	166
15	130
18	170

Ans. :

BMI	Cholesterol	$X_i - \bar{X}$	$Y_i - \bar{Y}$	$(X_i - \bar{X}) \times (Y_i - \bar{Y})$	$(X_i - \bar{X})^2$
17	140	-2.4	-21	50.4	5.76
21	189	1.6	28	44.8	2.56
24	210	4.6	49	225.4	21.16
28	240	8.6	79	679.4	73.96
14	130	-5.4	-31	167.4	29.16
16	100	-3.4	-61	207.4	11.56
19	135	-0.4	-26	10.4	0.16
22	166	2.6	5	13	6.76
15	130	-4.4	-31	136.4	19.36
18	170	-1.4	9	-12.6	1.96
$\bar{X}$(Mean of X) = 19.4	$\bar{Y}$(Mean of Y) = 161			Total = 1522	Total = 172.4

Hence,
$$\beta_1 = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2}$$

$$\beta_1 = \frac{1522}{172.4} = 8.8283$$

$$\beta_0 = \bar{y} - \beta_1 \bar{x}$$

$$\beta_0 = 161 - 8.8283 \times 19.4$$

$$= -10.2690$$

Hence, the regression line is

$$Y = -10.2690 + 8.8283X$$

Hence, someone having a BMI of 27 would likely have cholesterol level as

$$\text{Cholesterol} = -10.2690 + 8.8283 \times 27$$

$$\text{Cholesterol} = 228$$

Q.5 Describe some of the use cases for Linear Regression. **(4 Marks)**

Ans. :

Some of the common use cases (or applications) of linear regression are as following.

1. Healthcare

Healthcare industry is evolving and there are several researches that are going on. Linear regression could be used to establish the relationship between treatment and its effects or to understand complex operations of the human body to derive certain relationships.

2. Demand forecasting

Businesses are always looking to maximise sales and reduce inventory. Sales might depend on several factors and it could be really helpful to determine the relationship of sales with those factors. Businesses can then try to modify those factors and appropriately forecast sales.

3. Other predictions

There are several other areas where predictions can be made using the established linear relationship between the variables. It could be sports outcomes, crop output, machinery performance, fitness, and other similar areas.

Q.6 Write a short note on Ridge regression. **(4 Marks)**

Ans. :

- Multicollinearity is a statistical concept where several independent variables in a model are correlated. Two variables are considered to be perfectly collinear if their correlation coefficient is +/- 1.0. Multicollinearity among independent variables results in less reliable statistical inferences.
- It is better to use independent variables that are not correlated or repetitive when building multiple regression models that use two or more variables. The existence of multicollinearity in a data set can lead to less reliable results due to larger standard errors.
- The linear regression fits a linear model with coefficients $w = (w_1, w_2, \ldots, w_p)$ to minimise the residual sum of squares between the observed targets in the dataset, and the targets predicted by the linear approximation. Mathematically it solves a problem of the following form.

$$\min_w ||Xw - y||_2^2$$

- The coefficient estimates for Ordinary Least Squares rely on the independence of the features. When features are correlated and the columns of the design matrix have an approximately linear dependence, the design matrix becomes close to singular and as a result, the least-squares estimate becomes highly sensitive to random errors in the observed target, producing a large variance. This situation of multicollinearity can arise, for example, when data are collected without an experimental design.
- Ridge regression is used in the case of multicollinearity. It places restrictions on the magnitudes of the estimated coefficients to avoid distortion. A penalty term proportional to the sum of the squares of the coefficients is added to reduce the standard errors. It is also called as L2 regularisation.
- The cost function of Ridge regression is given as following.

$$\min_w ||Xw - y||_2^2 + \alpha ||w||_2^2$$

- The complexity parameter $\alpha \geq 0$ controls the amount of shrinkage: the larger the value of α, the greater the amount of shrinkage and thus the coefficients become more robust to collinearity. The Fig. 3.2 illustrates Ridge coefficients as a function of the regularisation.
- L2 regularisation results in smaller overall weight values and stabilises the weights when there is high correlation between the input features. L2 regularisation tries to estimate the mean of the data

to avoid overfitting. L2 regularisation is more commonly used than L1 regularisation.

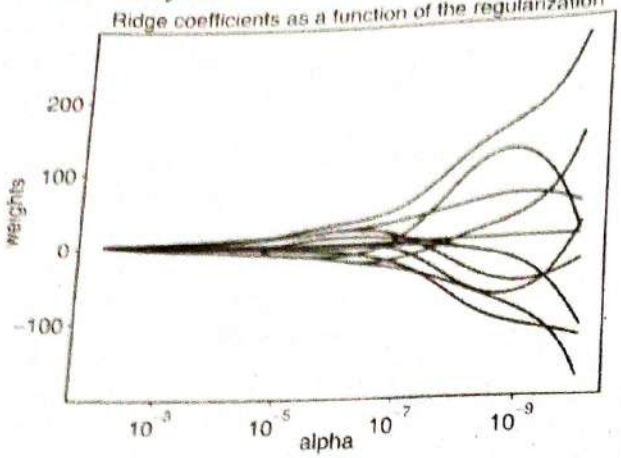


Fig. 3.2

Q.7 Write a short note on L1 regularisation. **(4 Marks)**

Ans. :

- Lasso regression adds a penalty term proportional to the sum of the absolute values of the coefficients. It is also called as L1 regularisation.
- L1 regularisation has the effect of reducing the number of features used in the model by pushing to zero the weights of features that would otherwise have small weights. As a result, L1 regularisation results in sparse models and reduces the amount of noise in the model. L1 regularisation tries to estimate the median of the data to avoid overfitting.
- Mathematically, it consists of a linear model with an added regularisation term. The objective function to minimise is as following.

$$\min_{w} \left(\frac{1}{2nsamples}\right) ||Xw - y||_2^2 + \alpha ||w||_1$$

- The lasso estimate thus solves the minimisation of the least-squares penalty with $\alpha ||w||_1$ added, where α is a constant and $||w||_1$ is the l_1 – norm of the coefficient vector.

Q. 8 With an example, illustrate Mean Error (ME) cost function. **(4 Marks)**

Ans. :

Mean Error (ME) is the simplest of all cost functions. It is simply a mean (average) of the difference between the actual value and the predicted values.

Example

Y (Actual)	Y' (Predicted)	Error = Y - Y'
10	9.2	0.8
8	8.3	-0.3
5	4.7	0.3
7	7.9	-0.9
4	3.1	0.9
	Total	0.8
	Mean Error	0.8 / 5 = **0.16**

- The errors can be both negative or positive and during summation, they tend to cancel each other out. In fact, it is theoretically possible that the errors are such that positive and negatives cancel each other to give zero error which could incorrectly mean a perfect model. So, Mean Error is not a recommended cost function.

Q.9 With an example, illustrate Mean Squared Error (MSE) cost function. **(4 Marks)**

Ans. :

Mean Squared Error (MSE), an improvement upon Mean Error (ME) cost function, squares the difference between the actual and predicted value to avoid negative values cancelling out positive values in error calculation.

Example

Y (Actual)	Y' (Predicted)	Error = Y - Y'	Error Squared
10	9.2	0.8	0.64
8	8.3	– 0.3	0.09
5	4.7	0.3	0.09
7	7.9	-0.9	0.81
4	3.1	0.9	0.81
		Total	2.44
		Mean Squared Error	$\frac{2.44}{5}$ = **0.488**

Q.10 With an example, illustrate Mean Absolute Error (MAE) cost function. **(4 Marks)**

Ans. :

Mean Absolute Error (MAE) also attempts to solve the cancelling out problem with Mean Error (ME). MAE takes the absolute value (without sign) of the difference between the actual value and the predicted value and then calculates the mean of the sum.

Example

Y (Actual)	Y' (Predicted)	Absolute Error = |Y - Y'|
10	9.2	0.8
8	8.3	0.3
5	4.7	0.3
7	7.9	0.9
4	3.1	0.9
	Total	3.2
	Mean Absolute Error	$\frac{3.2}{5} = 0.64$

- If you compare the MAE value (0.64) with the ME value (0.16) that you found for the same data, you find that MAE is a better way to calculate error when compared with ME.

Q.11 With an example, illustrate Root Mean Squared Error (RMSE) cost function. **(4 Marks)**

Ans. :

- Root Mean Squared Error (RMSE) also attempts to solve the cancelling out problem with Mean Error (ME). RMSE takes the root of Mean Squared Error (MSE).

Example

Y (Actual)	Y' (Predicted)	Error = Y - Y'	Error Squared
10	9.2	0.8	0.64
8	8.3	– 0.3	0.09
5	4.7	0.3	0.09
7	7.9	– 0.9	0.81
4	3.1	0.9	0.81
		Total	2.44
		Root Mean Squared Error	$\sqrt{\frac{2.44}{5}} = 0.698$

Q.12 With an example, explain R-Squared. **(6 Marks)**

Ans. :

- R-Squared (R^2 or the coefficient of determination) is a statistical measure in a regression model that determines the proportion of variance in the dependent variable that can be explained by the independent variable. In other words, R^2 shows how well the data fit the regression model.
- For example, R^2 of 60% reveals that 60% of the data fit the regression model. Generally, a higher R^2 indicates a better fit for the model.
- The Fig. 3.3 illustrates the model's performance for various values of R2.

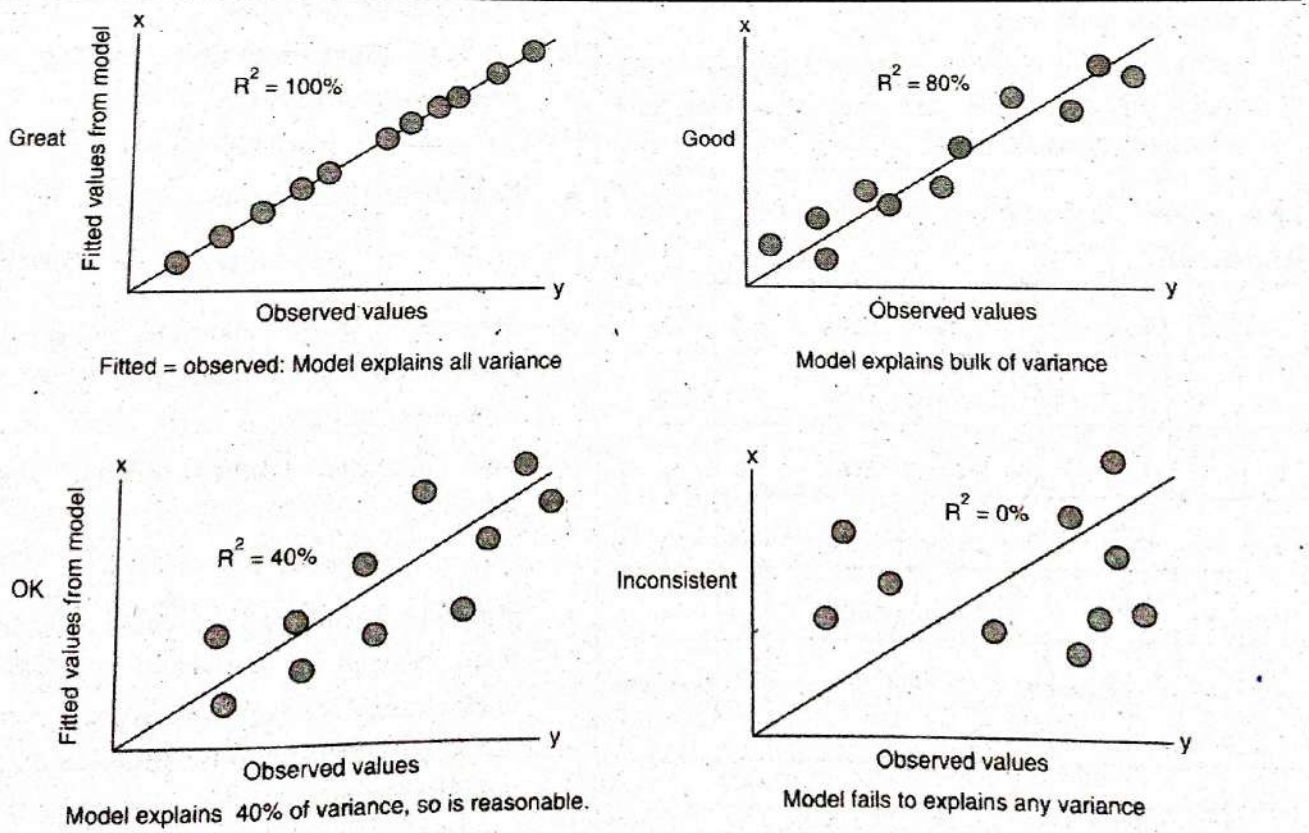


Fig. 3.3

- The quality of the statistical measure depends on many factors, such as the nature of the variables employed in the model, the units of measure of the variables, and the applied data transformation. Thus, sometimes, a high R^2 can indicate the problems with the regression model.
- A low R^2 figure is generally a bad sign for predictive models. However, in some cases, a good model may show a small value. There is no universal rule on how to incorporate the statistical measure in assessing a model. The context of the experiment or forecast is extremely important and, in different scenarios, the insights from the metric can vary. R2 can take any values between 0 to 1 (as it is denoted in percentage).
- The formula for calculating R-squared is as following.

$$R^2 = \frac{\text{Sum of squares due to regression}}{\text{total sum of squares}}$$

- The sum of squares due to regression measures how well the regression model represents the data that was used for modelling. The total sum of squares measures the variation in the observed data.

Example

- Assume that the regression line is given as the following function.
- $y = 2.2 + 0.6\ x$. For a given set of data points x, R2 could be calculated for the regression line $y = 2.2 + 0.6\ x$ as following.

Data Point x	Actual Value Y	$Y - \bar{Y}$	$(Y - \bar{Y})^2$	Estimated Value $\hat{Y} = 2.2 + 0.6\ x$	$\hat{Y} - \bar{Y}$	$(\hat{Y} - \bar{Y})^2$
1	2	−2	4	$2.2 + 0.6 \times 1 = 2.8$	−1.2	1.44
2	4	0	0	$2.2 + 0.6 \times 2 = 3.4$	−0.6	0.36
3	5	1	1	$2.2 + 0.6 \times 3 = 4.0$	0	0
4	4	0	0	$2.2 + 0.6 \times 4 = 4.6$	0.6	0.36
5	5	1	1	$2.2 + 0.6 \times 5 = 5.2$	1.2	1.44
	Mean $\bar{Y} = 4$		Total = 6			Total = 3.6

Hence,

$$R^2 = \frac{\text{Sum of squares due to regression}}{\text{total sum of squares}} = \frac{3.6}{6} = 0.6$$

Q.13 Explain the core concept behind derivative-based optimisation. **(4 Marks)**

Ans. :

- Derivative-based optimisation techniques use differentiation to find out derivatives of a continuous function. The derivative is then used to find optimum values for the particular function. The derivative information helps to determine the direction of search for an optimal value.
- At every step, you carefully evaluate two things :
  1. Which direction to go and
  2. How far to go in that direction
- A few extra steps in the wrong direction could lead to injury, lost energy, or backtracking. When you need to choose between a set of paths to go down, you usually pick-up the most promising path that is likely to get you down more quickly than other potential paths and provide maximum descent towards the downhill.
- That is the general concept behind derivative-based optimisation. You use derivative information to determine search direction and then descent in that direction, for a calculated distance, in the hope of finding an optimal value.
- Generally (mathematically) speaking, consider a simple function

$$y = f(x)$$

- Derivative of the function is denoted as

$$f'(x) \quad \text{or} \quad \text{as} \ \frac{dx}{dy}$$

- Derivative $f'(x)$ gives the slope of $f(x)$ at point x. It specifies how a small change in input x affects the change in y or $f(x)$.
- Now, consider another function such as the following :

$$f(x + \varepsilon) \approx f(x) + \varepsilon f'(x)$$

- You can reduce $f(x)$ by moving x in small steps with opposite sign of derivative. You will use this mathematical concept when you learn about the actual derivative-based optimisation techniques later in the section. For now, remember the core concepts (search direction and distance) behind derivative-based optimisation.

Q.14 Explain the characteristics of derivative-based optimisation techniques. **(6 Marks)**

Ans. :

- Derivative-based optimisation techniques have the following characteristics :

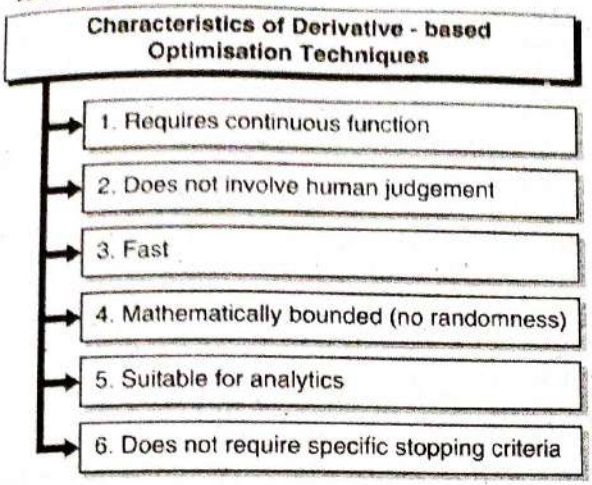


Fig. 3.4 : Characteristics of Derivative-based Optimisation Techniques

1. **Requires continuous function :** Derivative-based optimisation techniques require a continuous function to work with. Functions that cannot be differentiated are not suitable.
2. **Does not involve human judgement :** Derivative-based optimisation techniques do not involve human judgment to determine search direction for finding optima. Search direction is guided through derivatives of the continuous function.
3. **Fast :** Derivative-based optimisation techniques are comparatively faster than derivative-free optimisation techniques as they do not involve human judgement and random numbers. They are mathematically driven.
4. **Mathematically bounded (no randomness) :** Derivative-free optimisation techniques, derivative-based optimisation techniques do not use random numbers to determine search direction. Derivatiye-based optimisation techniques use function derivatives to determine search direction.
5. **Suitable for analytics :** Derivative-based optimisation techniques can be used for carrying out analytics on optimisation approaches and computation time. Irrespective of the function, you can analyse search performance parameters such as how much time did it take to find optima, where was optima found, and how many steps were required to search optima.
6. **Does not require specific stopping criteria :** Derivative-free optimisation techniques, derivative-based optimisation techniques do not require specific stopping criteria. Derivative-based optimisation techniques stop the optimisation search process after computing multiple rounds of derivatives as desired.

Q.15 Write the steps for Steepest Descent optimisation technique. **(4 Marks)**

Ans. :

- Steepest Descent is one of the oldest optimisation techniques that uses gradient-based approach for finding optimal value for a function in a multidimensional space. It is also known as the gradient descent method.
- The basic idea, is that you want to minimise a function f (x), where x is a vector (x1, x2, x3, ... xn) that has elements for each feature value, starting from some initial guess x (0). You try to find a sequence of new points x (i) that move downhill towards a minimal solution for f (x).
- The steepest descent method works for any number of dimensions. Therefore, you could take derivatives of the function in each of the different dimensions of x to find the optimal value for the function f (x). The whole set of functions can be written as $\nabla f(x)$, which is a vector with gradient elements $\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \frac{\partial f}{\partial x_3}, \ldots \frac{\partial f}{\partial x_n}$ for each dimension of x.
- In derivative-based optimisation technique that you need to find out the search direction as well step size (or distance) to move in that direction. At any given point, there could be thousands of search directions possible. Hence, it is crucial to find out which search direction to take and how far should you descent in that direction to find the minimal value.

So, if search direction is a line and you are at a point xk, then

$$x_{k+1} = x_k + a_k d_k$$

[where a_k is the step size and dk is the search direction]

- To find the search direction dk, you compute derivative ∇f (xk) and move in the negative direction.

Hence $d_k = -\nabla f(x_k)$ and the new point x_{k+1} is given as

$$x_{k+1} = x_k - a_k d_k$$

- If you assume $a_k = 1$, then it further simplifies the search by just repeatedly moving downhill in the search direction until the minimal value is found.
- The individual parts of the steepest descent method, write down the steps for it.

1. Take initial value of x say x_0 at point $k = 0$.
2. Determine search direction, $d_k = -\nabla f(x_k)$. If $d_k = 0$ (or less than a chosen low value), then stop.
3. Calculate $x_{k+1} = x_k - a_k d_k$
4. Calculate step size a_k by minimizing $f(x_{k+1})$ with respect to a_k
5. Set $x_{k+1} = x_k - a_k d_k$ and $k = k + 1$. Go to step 2.

Q.16 Use the Steepest Descent method to minimise $f(x_1, x_2) = x_1^2 - x_1x_2 + x_2^2$ up to a point where the difference between two successive iterations is less than 0.2. Take starting point as $\left(1, \frac{1}{2}\right)^T$. **(6 Marks)**

Ans. :

Iteration 1

Now, substitute the initial value (starting point) at $k = 1$ for $\nabla f(X)$ taking the given values for x_1, x_2 as 1 and $\frac{1}{2}$ respectively.

Hence,

$$\nabla f(X_1) = (2x_1 - x_2, -x_1 + 2x_2)^T$$

$$\nabla f(X_1) = \left(2 \times 1 - \frac{1}{2}, -1 + 2 \times \frac{1}{2}\right)^T$$

$$\nabla f(X_1) = \left(\frac{3}{2}, 0\right)^T$$

$$d_1 = -\nabla f(X_1)$$

$$d_1 = -\left(\frac{3}{2}, 0\right)^T$$

Now, according to steepest descent method, the next point X2 could be found as

$$X_2 = X_1 - a_1 d_1$$

X_1 the starting point is given as $\left(1, \frac{1}{2}\right)^T$

$$X_2 = \left(1, \frac{1}{2}\right)^T - a_1 \times \left(\frac{3}{2}, 0\right)^T$$

$$X_2 = \left(1 - \frac{3}{2}a_1, \frac{1}{2}\right)^T$$

- You now have values for point X_2. You can now calculate $f(X_2)$ by substituting the value of X_2 in the given function.

$$f(X_2) = x_1^2 - x_1x_2 + x_2^2$$

$$f(X_2) = \left(1 - \frac{3}{2}a_1\right)^2 - \left(1 - \frac{3}{2}a_1\right) \times \frac{1}{2} + \left(\frac{1}{2}\right)^2$$

$$f(X_2) = \left(1 - \frac{3}{2}a_1\right)^2 - \frac{1}{2} + \frac{3}{4}a_1 + \frac{1}{4}$$

$$f(X_2) = \left(1 - \frac{3}{2}a_1\right)^2 + \frac{3}{4}a_1 - \frac{1}{4}$$

$$f(X_2) = \left(1^2 + \left(\frac{3}{2}a_1\right)^2 - 2 \times 1 \times \frac{3}{2}a_1\right) + \frac{3}{4}a_1 - \frac{1}{4}$$ (using the formula for $(a - b)^2 = a^2 + b^2 - 2ab$)

$$f(X_2) = \left(1 + \frac{9}{4}a_1^2 - 3a_1\right) + \frac{3}{4}a_1 - \frac{1}{4}$$

$$f(X_2) = 1 + \frac{9}{4}a_1^2 - 3a_1 + \frac{3}{4}a_1 - \frac{1}{4}$$

- Now, to find the value of a_1 you differentiate $f(X_2)$ with respect to a_1 and minimise the derivative.

$$\frac{\partial f}{da_1} = \frac{18}{4}a_1 - 3 + \frac{3}{4}$$

$$\frac{\partial f}{da_1} = \frac{18}{4}a_1 - \frac{9}{4}$$

- To minimise the derivative, you can write

$$\frac{18}{4}a_1 - \frac{9}{4} = 0$$

$$a_1 = \frac{1}{2}$$

- Substituting the value of a_1 you get X_2 as

$$X_2 = \left(1 - \frac{3}{2}\alpha_1, \frac{1}{2}\right)^T$$

$$X_2 = \left(1 - \frac{3}{2} \times \frac{1}{2}, \frac{1}{2}\right)^T$$

$$X_2 = \left(\frac{1}{4}, \frac{1}{2}\right)^T$$

So, $f(X_2) = x_1^2 - x_1x_2 + x_2^2$

$$f(x_2) = \left(\frac{1}{4}\right)^2 - \frac{1}{4} \times \frac{1}{2} + \left(\frac{1}{2}\right)^2$$

$$f(x_2) = \frac{3}{16} = 0.1875$$

X_1 is given in the question as $\left(1, \frac{1}{2}\right)^T$.

- Hence, using these values for x_1 and x_2 respectively you get

$$f(X_1) = 1^2 - 1 \times \frac{1}{2} + \left(\frac{1}{2}\right)^2$$

$$f(X_1) = \frac{3}{4} = 0.75$$

Now calculate the difference between $f(X_2)$ and $f(X_1)$, you get

$$||f(X_2) - f(X_1)|| = 0.1875 - 0.75 = 0.5625$$

$0.5625 \not< 0.2$. Hence, you need to perform another iteration.

Iteration 2

Now, you need to find d_2 and X_3 using X_2.

$$X_2 = \left(\frac{1}{4}, \frac{1}{2}\right)^T$$

$$\nabla f(X_2) = (2x_1 - x_2, -x_1 + 2x_2)^T$$

$$\nabla f(X_2) = \left(2 \times \frac{1}{4} - \frac{1}{2}, -\frac{1}{4} + 2 \times \frac{1}{2}\right)^T$$

$$\nabla f(X_2) = \left(0, \frac{3}{4}\right)^T$$

$$d_2 = -\nabla f(X_2)$$

$$d_2 = -\left(0, \frac{3}{4}\right)^T$$

Now, according to steepest descent method, the next point X_3 could be found as

$$X_3 = X_2 - a_2 d_2$$

You calculated $X_2 = \left(\frac{1}{4}, \frac{1}{2}\right)^T$

$$\text{So, } X_3 = \left(\frac{1}{4}, \frac{1}{2}\right)^T - a_2 \times \left(0, \frac{3}{4}\right)^T$$

$$X_3 = \left(\frac{1}{4}, \frac{1}{2} - \frac{3}{4}a_2\right)^T$$

You now have values for point X_3. You can now calculate $f(X_3)$ by substituting the value of X_3 in the given function.

$$f(X_3) = x_1^2 - x_1x_2 + x_2^2$$

$$f(X_3) = \left(\frac{1}{4}\right)^2 - \left(\frac{1}{4}\right) \times \left(\frac{1}{2} - \frac{3}{4}a_2\right) + \left(\frac{1}{2} - \frac{3}{4}a_2\right)^2$$

$$f(X_3) = \frac{1}{16} - \frac{1}{8} + \frac{3}{16}a_2 + \frac{1}{4} + \frac{9}{16}a_2^2 - \frac{3}{4}a_2$$

$$f(X_3) = \frac{9}{16}a_2^2 + \frac{3 - 3 \times 4}{16}a_2 + \frac{1 - 2 + 4}{16}$$

$$f(X_3) = \frac{9}{16}a_2^2 - \frac{9}{16}a_2 + \frac{3}{16}$$

Now, to find the value of a_2 you differentiate f (3) with respect to a_2 and minimize the derivative.

$$\frac{\partial f}{da_2} = \frac{18}{16}a_2 - \frac{9}{16}$$

To minimize the derivative, you can write

$$\frac{18}{16}a_2 - \frac{9}{16} = 0$$

$$a_2 = \frac{1}{2}$$

Substituting the value of a_2 you get X_3 as

$$X_3 = \left(\frac{1}{4}, \frac{1}{2} - \frac{3}{4}a_2\right)^T$$

$$X_3 = \left(\frac{1}{4}, \frac{1}{2} - \frac{3}{4} * \frac{1}{2}\right)^T$$

$$X_3 = \left(\frac{1}{4}, \frac{1}{8}\right)^T$$

$$\text{So, } f(X_3) = x_1^2 - x_1x_2 + x_2^2$$

$$f(X_3) = \left(\frac{1}{4}\right)^2 - \frac{1}{4} \times \frac{1}{8} + \left(\frac{1}{8}\right)^2$$

$$f(X_3) = \frac{3}{64} = 0.047$$

Now calculate the difference between $f(X_3)$ and $f(X_2)$, you get

$$||f(X_3) - f(X_2)|| = 0.047 - 0.1875 = 0.14$$

$0.14 < 0.2$. Hence, you can now stop.

Hence, $X_3 = \left(\frac{1}{4}, \frac{1}{8}\right)^T$ is the optimal solution for $f(x_1, x_2) = x_1^2 - x_1x_2 + x_2^2$

Q.17 Use the Steepest Descent method to minimise $f(x_1, x_2) = 3x_1^2 + x_2^2$. Stop after two iterations. Use the initial values as $(4, 4)^T$. **(6 Marks)**

Ans. :

$$\nabla f(X) = (6x_1, 2x_2)^T$$

Iteration 1

Now, substitute the initial value (starting point) at $k = 1$ for $\nabla f(X)$ taking the given values for x_1, x_2 as and 4 respectively.

Hence, $\nabla f(X_1) = (6x_1, 2x_2)^T$

$$\nabla f(X_1) = (6 \times 4, 2 \times 4)^T$$

$$\nabla f(X_1) = (24, 8)^T$$

$$d_1 = -\nabla f(X_1)$$

$$d_1 = (24, 8)^T$$

Now, according to steepest descent method, the next point X_2 could be found as

$$X_2 = X_1 - a_1 d_1$$

X_1 the starting point is given as $(4, 4)^T$

$$X_2 = (4, 4)^T - a_1 \times (24, 8)^T$$

$$X_2 = (4 - 24a_1, 4 - 8a_1)^T$$

You now have values for point X_2. You can now calculate $f(X_2)$ by substituting the value of X_2 in the given function.

$$f(X_2) = 3x_1^2 + x_2^2$$

$$f(X_2) = 3(4 - 24a_1)^2 + (4 - 8a_1)^2$$

$$f(X_2) = 3(4^2 + (24a_1)^2 - 2 \times 4 \times 24a_1) + 4^2 + (8a_1)^2 - 2 \times 4 \times 8a_1$$

$$f(X_2) = 48 + 1728a_1^2 - 576a_1 + 16 + 64a_1^2 - 64a_1$$

$$f(X_2) = 1792a_1^2 - 640a_1 + 64$$

Now, to find the value of a_1 you differentiate $f(X_2)$ with respect to a_1 and minimise the derivative.

$$\frac{\partial f}{da_1} = 3584a_1 - 640$$

To minimise the derivative, you can write

$$3584a_1 - 640 = 0$$

$$a_1 = 0.179$$

Substituting the value of a_1 you get X_2 as

$$X_2 = (4 - 24a_1, 4 - 8a_1)^T$$

$$X_2 = (4 - 24 \times 0.179, 4 - 8 \times 0.179)^T$$

$$X_2 = (-0.296, 2.568)^T$$

So, $f(X_2) = 3x_1^2 + x_2^2$

$$f(X_2) = 3 \times (-0.296)^2 + (2.568)^2$$

$$f(X_2) = 0.26 + 6.59 = 6.85$$

$$f(X_1) = 3x_1^2 + x_2^2$$

$$f(X_1) = 3 \times (4)^2 + (4)^2$$

$$f(X_1) = 48 + 16 = 64$$

Indeed the function $f(X)$ is reducing as you moved from X_1 to X_2

$$f(X_1) = 64 \text{ whereas } f(X_2) = 6.85$$

Iteration 2

$$X_2 = (-0.296, 2.568)^T$$

$$\nabla f(X_2) = (6x_1, 2x_2)^T$$

$$\nabla f(X_2) = (6 \times -0.296, 2 \times 2.568)^T$$

$$\nabla f(X_2) = (1.78, 5.13)^T$$

$$d_1 = -\nabla f(X_2)$$

$$d_2 = -(1.78, 5.13)^T$$

Now, according to steepest descent method, the next point X_3 could be found as

$$X_3 = X_2 - a_2 d_2$$

From previous iteration,

$$X_2 = (-0.296, 2.568)^T$$

$$X_3 = (-0.296, 2.568)^T - a_2 \times (1.78, 5.13)^T$$

$$X_3 = (-0.296 - 1.78a_2, 2.568 - 5.13a_2)^T$$

You now have values for point X_3. You can now calculate $f(X_3)$ by substituting the value of X_3 in the given function.

$$f(X_3) = 3x_1^2 + x_2^2$$

$$f(X_3) = 3(-0.296 - 1.78a_2)^2 + (2.568 - 5.13a_2)^2$$

$$f(X_3) = 3((-0.296)^2 + (1.78a_2)^2 - 2 \times (-0.296) \times 1.78a_2) + 2.568^2 + (5.13a_2)^2 - 2 \times 2.568 \times 5.13a_2$$

$$f(X_3) = 0.27 + 9.5a_2^2 + 3.16a_2 + 6.6 + 26.31a_2^2 - 26.34a_2$$

$$f(X_2) = 35.81a_2^2 - 23.18a_2 + 6.87$$

Now, to find the value of a_2 you differentiate $f(X_3)$ with respect to a_2 and minimise the derivative.

$$\frac{\partial f}{da_2} = 71.62a_2 - 23.18$$

To minimise the derivative, you can write

$$71.62a_2 - 23.18 = 0$$

$$a_2 = 0.32$$

Substituting the value of a_2 you get X_3 as

$$X_3 = (-0.296 - 1.78a_2, 2.568 - 5.13a_2)^T$$

$$X_3 = (-0.296 - 1.78 \times 0.32, 2.568 - 5.13 \times 0.32)^T$$

$$X_3 = (-0.87, 0.93)^T$$

So, $f(X_3) = 3x_1^2 + x_2^2$

$$f(X_3) = 3 \times (-0.87)^2 + (0.93)^2$$

$$f(X_3) = 2.27 + 0.86 = 3.13$$

Function $f(X)$ is further reducing as you moved from X_2 to X_3.

$$f(X_2) = 6.85 \text{ whereas } f(X_3) = 3.13$$

Q.18 Minimise $f(x_1, x_2) = 4x_1 - 2x_2 + 2x_1^2 + 2x_1x_2 + x_2^2$ with starting point $X_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ using the Newton's method.

(6 Marks)

Ans. :

$$\nabla f(X) = \begin{bmatrix} 4x_1 + 2x_2 + 4 \\ 2x_2 + 2x_1 - 2 \end{bmatrix}$$

To find second order derivative of $\nabla f(X)$ matrix, you differentiate each term twice, once for x_1 and once for x_2.

Differentiate $4x_1 + 2x_2 + 4$ once with respect to x_1, you get 4 and with respect to x_2.

$$\nabla^2 f(X) = \begin{pmatrix} 4 \text{ yet to be filled} \\ 2 \text{ yet to be filled} \end{pmatrix}$$

Now, differentiate $2x_2 + 2x_1 - 2$ once with respect to x_1, you get 2 and with respect to x_2. Then, complete the matrix as $\nabla^2 f(X) = \begin{pmatrix} 4 & 2 \\ 2 & 2 \end{pmatrix}$

Iteration 1

Now, substitute the initial value (starting point) at $k = 1$ for $\nabla f(X)$ taking the given values for x_1, x_2 as and 0 respectively.

Hence, $$\nabla f(X_1) = \begin{pmatrix} 4x_1 + 2x_2 + 4 \\ 2x_2 + 2x_1 - 2 \end{pmatrix}$$

$$\nabla f(X_1) = \begin{pmatrix} 4 \times 0 + 2 \times 0 + 4 \\ 2 \times 0 + 2 \times 0 - 2 \end{pmatrix}$$

$$\nabla f(X_1) = \begin{bmatrix} 4 \\ -2 \end{bmatrix}$$

Now, according to Newton's method, the next point X_2 could be found as

$$X_2 = X_1 - \frac{\nabla f(X_1)}{\nabla^2 f(X_1)}$$

$$X_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} - \begin{pmatrix} 4 & 2 \\ 2 & 2 \end{pmatrix}^{-1} \times \begin{pmatrix} 4 \\ -2 \end{pmatrix}$$

Inverse of a 2×2 matrix $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$ is given as

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{ad - bc} \times \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

So, $$\begin{pmatrix} 4 & 2 \\ 2 & 2 \end{pmatrix}^{-1} = \frac{1}{4 \times 2 - 2 \times 2} \begin{pmatrix} 2 & -2 \\ -2 & 4 \end{pmatrix}$$

$$= \frac{1}{4} \begin{pmatrix} 2 & -2 \\ -2 & 4 \end{pmatrix} = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & 1 \end{bmatrix}$$

$$X_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & 1 \end{bmatrix} \times \begin{bmatrix} 4 \\ -2 \end{bmatrix}$$

$$X_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 3 \\ -4 \end{bmatrix} = \begin{bmatrix} -3 \\ 4 \end{bmatrix}$$

So, $f(X_2) = 4x_1 - 2x_2 + 2x_1^2 + 2x_1x_2 + x_2^2$

$f(X_2) = 4 \times -3 - 2 \times 4 + 2 \times (-3)^2 + 2 \times -3 \times 4 + 4^2$

$f(X_2) = -12 - 8 + 18 - 24 + 16 = -10$

Iteration 2

$$\nabla f(X_2) = \begin{pmatrix} 4x_1 + 2x_2 + 4 \\ 2x_2 + 2x_1 - 2 \end{pmatrix}$$

$$\nabla f(X_2) = \begin{pmatrix} 4 \times -3 + 2 \times 4 + 4 \\ 2 \times 4 + 2 \times -3 - 2 \end{pmatrix}$$

$$\nabla f(X_1) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Now, according to Newton's method, the next point X_3 could be found as

$$X_3 = X_2 - \frac{\nabla f(X_2)}{\nabla^2 f(X_2)}$$

$$X_3 = \begin{pmatrix} -3 \\ 4 \end{pmatrix} - \begin{pmatrix} 4 & 2 \\ 2 & 2 \end{pmatrix}^{-1} \times \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$X_3 = \begin{bmatrix} -3 \\ 4 \end{bmatrix}$$

Q.19 Minimise $f(x_1, x_2) = x_1^2 - x_1x_2 + 3x_2^2$ with starting point $X_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$ using the Newton's method. **(6 Marks)**

Ans. :

$$\nabla f(X) = \begin{bmatrix} 2x_1 - x_2 \\ -x_1 + 6x_2 \end{bmatrix}$$

Differentiate $2x_1 - x_2$ once with respect to x_1, you get 2 and with respect to x_2, you get -1.

$$\nabla^2 f(X) = \begin{pmatrix} 2 \text{ yet to be filled} \\ -1 \text{ yet to be filled} \end{pmatrix}$$

Now, differentiate $-x_1 + 6x_2$ once with respect to x_1, you get -1 and with respect to x_2, you get 6. Then, complete the matrix as $\nabla^2 f(X) = \begin{bmatrix} 2 & -1 \\ -1 & 6 \end{bmatrix}$

Now, substitute the initial value (starting point) at $k = 1$ for $\nabla f(X)$ taking the given values for x_1, x_2 as and 2 respectively.

Hence,

$$\nabla f(X_1) = \begin{bmatrix} 2x_1 - x_2 \\ -x_1 + 6x_2 \end{bmatrix}$$

$$\nabla f(X_1) = \begin{bmatrix} 2 \times 1 - 2 \\ -1 + 6 \times 2 \end{bmatrix}$$

$$\nabla f(X_1) = \begin{bmatrix} 0 \\ 11 \end{bmatrix}$$

Now, according to Newton's method, the next point X_2 could be found as

$$X_2 = X_1 - \frac{\nabla f(X_1)}{\nabla^2 f(X_1)}$$

$$X_2 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \begin{bmatrix} 2 & -1 \\ -1 & 6 \end{bmatrix}^{-1} \times \begin{bmatrix} 0 \\ 11 \end{bmatrix}$$

Inverse of a 2×2 matrix $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$ is given as

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{ad - bc} \times \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

So,

$$\begin{bmatrix} 2 & -1 \\ -1 & 6 \end{bmatrix}^{-1} = \frac{1}{2 \times 6 - (-1) \times (-1)} \begin{bmatrix} 6 & 1 \\ 1 & 2 \end{bmatrix}$$

$$= \frac{1}{11} \begin{bmatrix} 6 & 1 \\ 1 & 2 \end{bmatrix}$$

$$X_2 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \frac{1}{11} \begin{bmatrix} 6 & 1 \\ 1 & 2 \end{bmatrix} \times \begin{bmatrix} 0 \\ 11 \end{bmatrix}$$

$$X_2 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \begin{bmatrix} 6 & 1 \\ 1 & 2 \end{bmatrix} \times \begin{bmatrix} \frac{0}{11} \\ \frac{11}{11} \end{bmatrix}$$

$$= \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \begin{bmatrix} 6 & 1 \\ 1 & 2 \end{bmatrix} \times \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \begin{bmatrix} 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Hence, $X_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ is the optimal solution for $f(x_1, x_2) = x_1^2 - x_1x_2 + 3x_2^2$

Q.20 Write a short note on Overfitting. (4 Marks)

Ans. :

- **Definition :** Overfitting occurs when the model matches the training data too closely, causing it to perform poorly on new data.
- Overfitting happens when a model learns the detail and noise in the training data to the extent that it negatively impacts the performance of the model on new data. This means that the noise or random fluctuations in the training data is picked up and learned as concepts by the model.
- Overfitting is more likely with nonparametric and nonlinear models that have more flexibility when learning a target function. As such, many nonparametric machine learning algorithms also include parameters or techniques to limit and constrain how much detail the model learns.
- For example, decision trees are a nonparametric machine learning algorithm that is very flexible and is subject to overfitting training data.
- This problem can be addressed by pruning a tree after it has learned in order to remove some of the detail it has picked up.

Q. 21 Write a short note on Underfitting. (4 Marks)

Ans. :

- **Definition :** Underfitting occurs when the model can neither model the training data nor work with the new data, causing it to perform poorly.
- If the target function is kept too simple, it may not be able to learn the essential features and characteristics of the training data. Underfitting could potentially happen due to unavailability of sufficient training data or incorrect selection of features.
- An underfit machine learning model is not a suitable model and it has a poor performance on the training data. Underfitting is easy to detect given a good performance metric. Underfitting can also be avoided by using more training data and also by carefully selecting the features on which the model is trained.

Q. 22 With a diagram, illustrate bias-variance trade-off. (6 Marks)

Ans. :

- There is a trade-off (need to balance bias and variance) to get the models just right for the job. If you denote the variable you are trying to predict as Y and the dependent variables as X, then you may assume that there is a relationship such that $Y = f(X) + \in$ where the error term $\in$ is normally distributed.
- The error term is an aggregation of reducible error and irreducible error.

  Total Error = Reducible Error + Irreducible Error
- The Fig. 3.5 illustrates what it looks like to balance bias and variance.

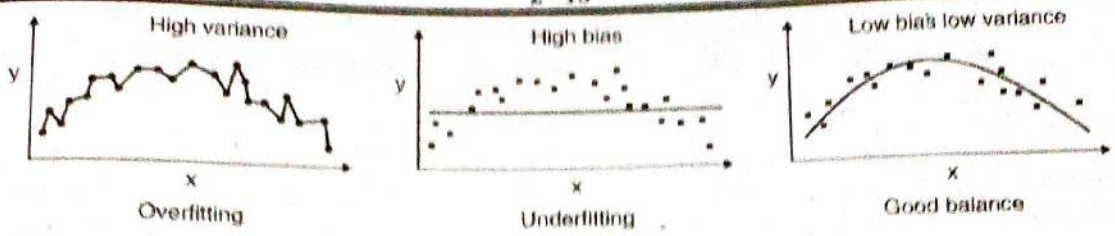


Fig. 3.5

- Bias and variance errors can, however, be minimised. Mathematically, the error function could be described as following.

Err (x) = Total Error

= $Bias^2$ + variance + Irreducible Error

- That third term, irreducible error, is the noise term in the true relationship that cannot fundamentally be reduced by any model. Given the true model and infinite data to calibrate it, you should be able to reduce both the bias and variance terms to 0. However, in a world with imperfect models and finite data, there is a trade-off between minimising the bias and minimising the variance.
- To build a good model, you need to find a good balance between bias and variance such that it minimises the total error.

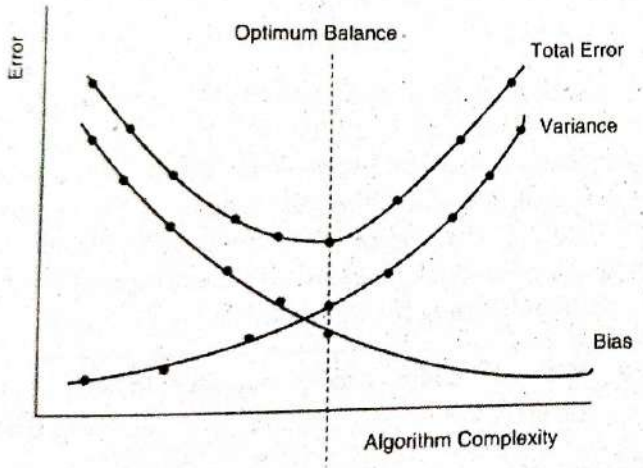


Fig. 3.5(a)

- Bias is high for a simpler model and decreases with an increase in model complexity, the line representing bias exponentially decreases as the model complexity increases.
- Variance is high for a more complex model and is low for simpler models. Hence, the line representing variance increases exponentially as the model complexity increases.
- The most optimal complexity of the model is right in the middle, where the bias and variance intersect. These values of bias and variance would produce the least error and are preferred. An optimal balance of bias and variance would not overfit or underfit the model.
- Use the flow chart shown in Fig. 3.5(b) for balancing bias and variance.

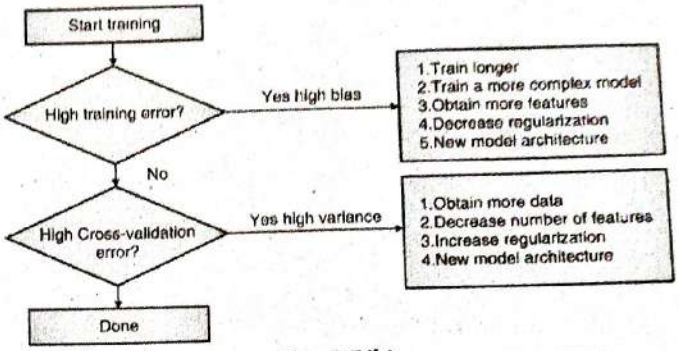


Fig. 3.5(b)

Q.23 What are the characteristics of a high bias model? **(4 Marks)**

Ans. : Characteristics of a high bias model are as following.

1. Failure to capture proper data trends
2. Low training accuracy
3. Potential towards underfitting
4. More generalised or overly simplified
5. High error rate

Q.24 What are the characteristics of a high variance model? **(4 Marks)**

Ans. : Characteristics of a high variance model are as following.

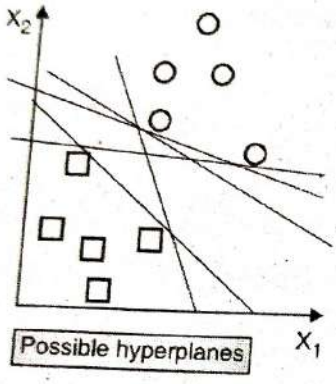
1. Noise in the data set
2. Low testing accuracy
3. Potential towards overfitting
4. Complex models
5. Trying to put all data points as close as possible

Chapter 4 : Supervised Learning : Classification

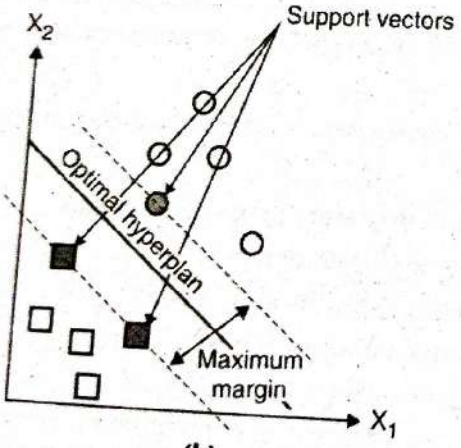
Q. 1 Write a short note on SVM. **(4 Marks)**

Ans. :

- **Definition :** Support Vector Machines (SVM) is a classification technique that uses critical boundary datapoints to create a hyperplane that differentiates the data points.
- SVM is a supervised learning method. Plot each datapoint as a point in a n-dimensional space (where n is the number of datapoint attributes you have). The higher the gap between the datapoints (highest boundary datapoints of one class and lowest boundary datapoints of another class), the better.



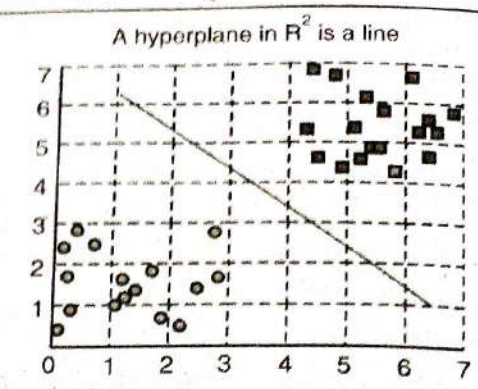
(a)



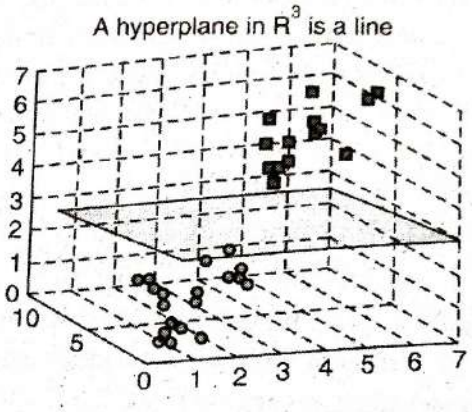
(b)

Fig. 4.1

- Hyperplanes are decision boundaries that help in classifying the datapoints. Datapoints falling on either side of the hyperplane can be attributed to different classes. Also, the dimension of the hyperplane depends upon the number of data attributes. If the number of input data attributes is 2, then the hyperplane is just a line. If the number of input data attributes is 3, then the hyperplane becomes a two-dimensional plane, and likewise.



(a)



(b)

Fig. 4.1

- Support vectors are datapoints that are closer to the hyperplane and influence the position and orientation of the hyperplane. Using these support vectors, you maximise the margin of the classifier. Deleting the support vectors will change the position of the hyperplane. These are the points that help in building the SVM model.

Q. 2 With a diagram, explain maximum margin concept behind SVM. **(6 Marks)**

Ans. :

- The larger the margin between the separating hyperplane the better. Typically, for linearly separable hyperplanes (which is a line), the decision boundary is given as $w \times x - b = 0$.
- Any positive point above the decision boundary classifies the input as one class and any negative point below the decision boundary classifies the input as another class. Select two parallel hyperplanes that separate the two classes of data, so that the distance between them is as large as possible.

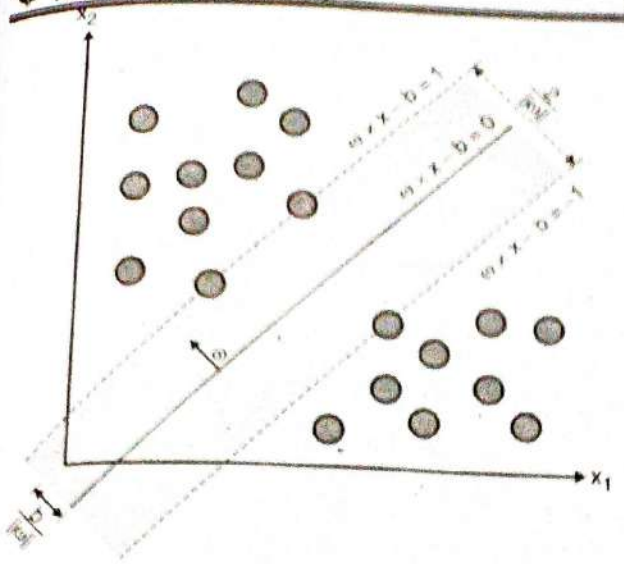


Fig. 4.2

- The region bounded by these two hyperplanes is called the "margin", and the maximum-margin hyperplane is the hyperplane that lies halfway between them. With a normalised or standardised dataset, these hyperplanes can be described by the following equations.

Hyperplane 1 is described as $w \times x - b = 1$ which classifies the data into the first class.

Hyperplane 2 is described as $w \times x - b = -1$ which classifies the data into the second class.

- Geometrically, the distance between these two hyperplanes(margin) is given as $m = \frac{2}{||w||}$. So, to maximise the distance between the planes you want to minimise $||w||$.

w and x in the above equations are vectors.

- SVM is a supervised learning method, x denotes input sample features such as $x_1, x_2, x_3, \ldots, x_n$. w denotes the set of weights w_i for each feature. b is the bias that can be added to the hyperplane to shift (adjust) it towards a particular class as required. y_i is the resultant classification based on the SVM hyperplanes. So, the datapoints must lie on the correct side of the margin for classifying them correctly.

So, If $w \times x - b \geq 1$, then $y_i = 1$ (Positive classification)

If $w \times x - b \leq -1$, then $y_i = -1$ (Negative classification)

To minimise $||w||$ (the absolute value of w without sign) subject to $y_i (w \times x_i - b) \geq 1$ for I = 1, 2 ... n

Q. 3 Given the following data, calculate hyperplane. Also, classify (0.6, 0.9) based on the calculated hyperplane.

A1	A2	y	α
0.38	0.47	+	65.52
0.49	0.61	-	65.52
0.92	0.41	-	0
0.74	0.89	-	0
0.18	0.58	+	0
0.41	0.35	+	0
0.93	0.81	-	0
0.21	0.1	+	0

Ans. :

The value of α is non-zero for only first two training examples. Hence, the first two training examples are support vectors.

Calculate the value of w and b as required to calculate the SVM hyperplanes.

Let $w = (w_1, w_2)$

$$w = \sum_{n=1}^{N} a_n y_n x_n$$

$$w_1 = a_1 * y_1 * A1x_1 + a_2 * y_2 * A1x_2$$
$$= 65.52 * 1 * 0.38 + 65.52 * -1 * 0.49$$
$$= -7.2$$

$$w_2 = a_1 * y_1 * A2x_1 + a_2 * y_2 * A2x_2$$
$$= 65.52 * 1 * 0.47 + 65.52 * -1 * 0.61$$
$$= -9.2$$

The parameter b can be calculated for each support vector as follows.

$$b_1 = 1 - w * x_1$$
$$= 1 - (-7.2) * 0.38 - (-9.2) * 0.47$$
$$= 8.06$$

$$b_2 = 1 - w * x_2$$
$$= 1 - (-7.2) * 0.47 - (-9.2) * 0.61$$
$$= 10$$

Averaging b_1, b_2 you get b = 9.03

Hence, the hyperplane line is defined as
$-7.2 x_1 - 9.2 x_2 + 9.03 = 0$

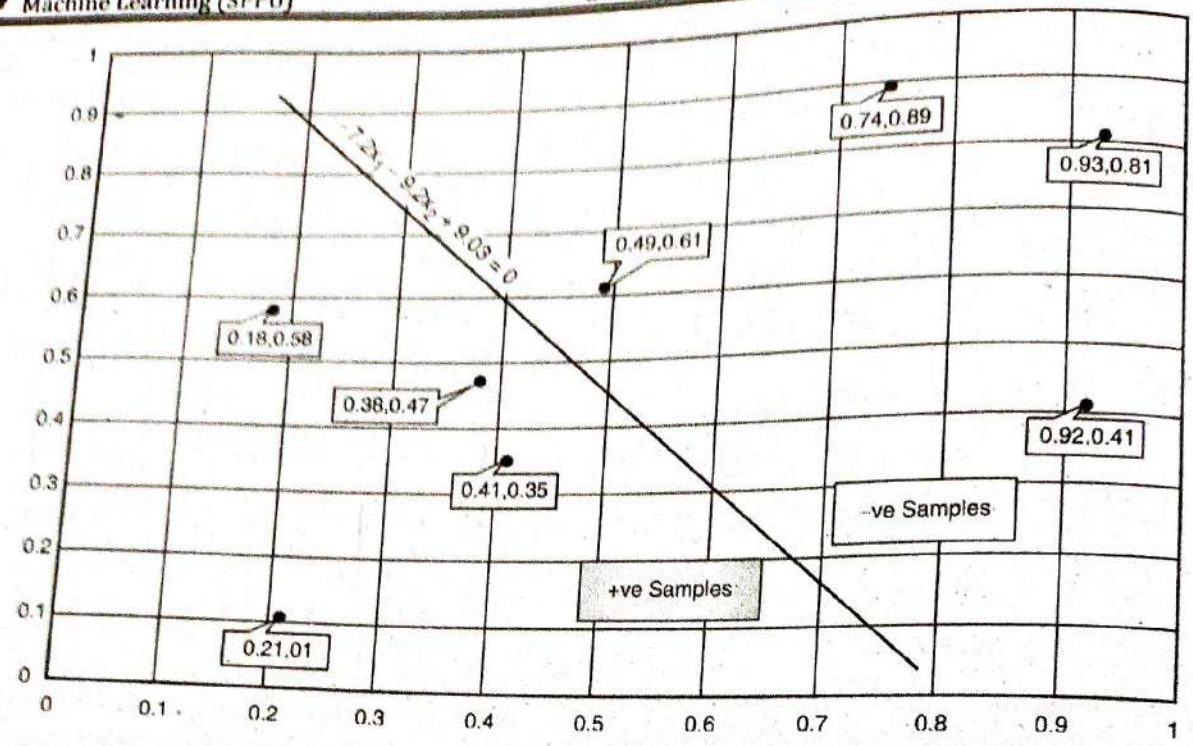


Fig. 4.3

Now, suppose that there is a new data point (0.6, 0.9) that you want to classify.

Putting the values in the equation $-7.2\,x_1 - 9.2\,x_2 + 9.03$ you get $-7.2 * 0.6 - 9.2 * 0.9 + 9.03 = -3.57$.

Hence, this is classified as negative sample.

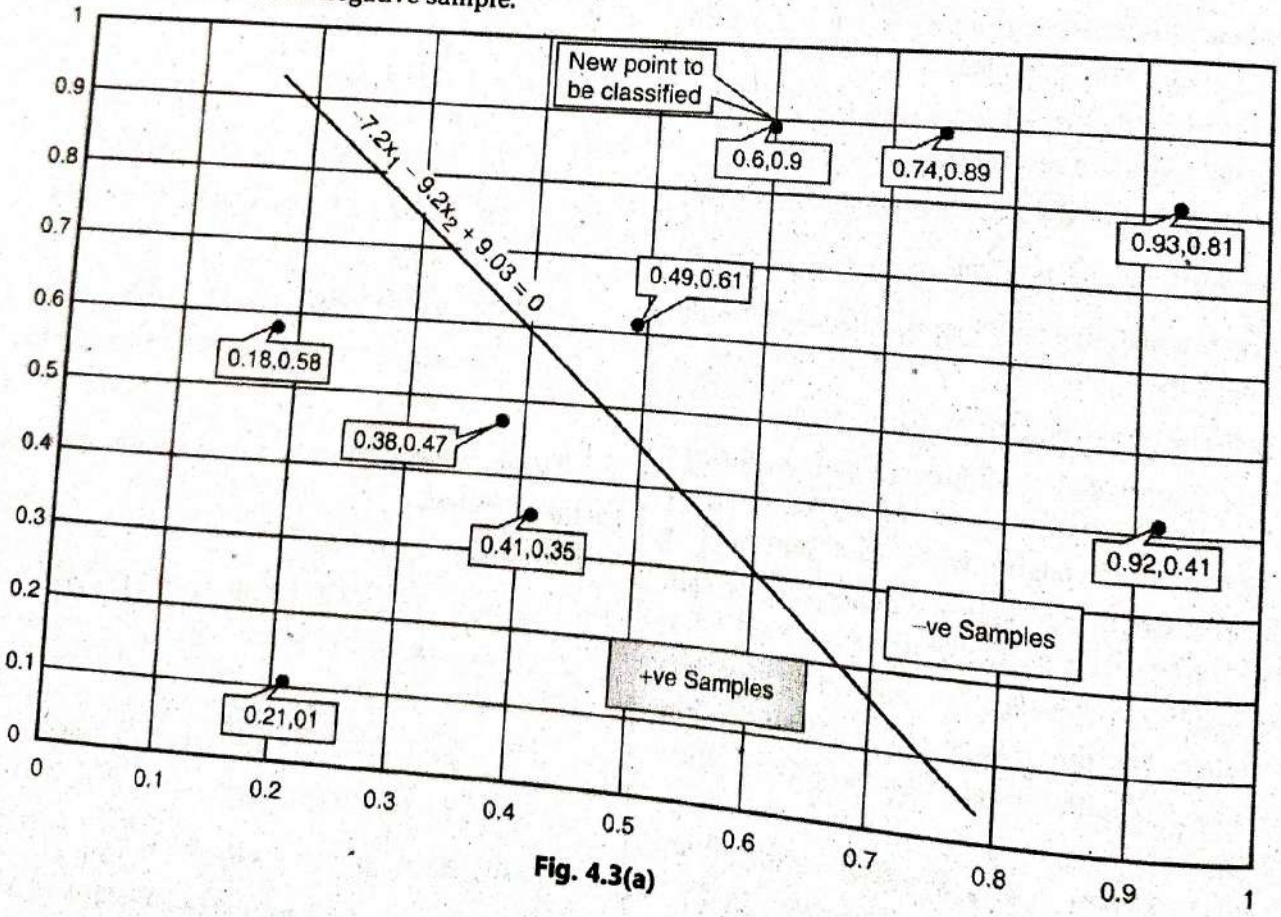


Fig. 4.3(a)

Q. 4 With a diagram, explain Kernel function. **(6 Marks)**

Ans. : You cannot really draw a line separating out the classes. In such scenarios, you use a kernel function to map the datapoints such that they can be "uplifted" to higher dimensions and then could possibly be classified.

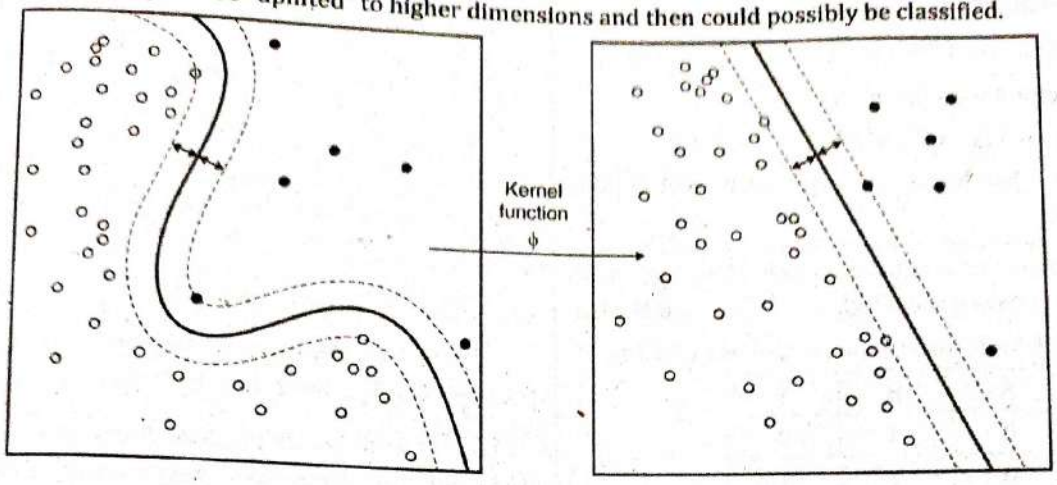


Fig. 4.4

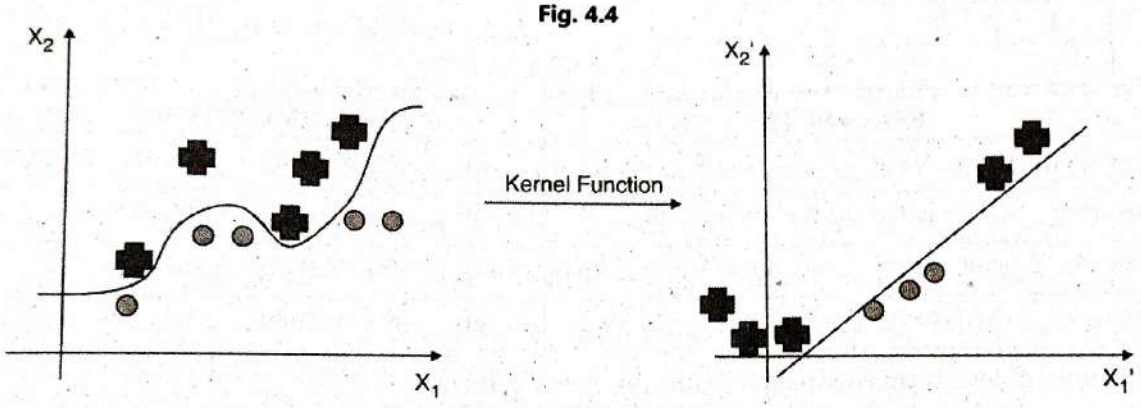


Fig. 4.4(a)

- The kernel function is often denoted by $\phi(x)$. You can choose the type of kernel function as appropriate for problem in hand. Some of the commonly used kernel functions are polynomial function, sigmoid function, and radial basis function. This technique of applying a mapping kernel function to adjust the data is also called as kernel trick.
- The Table 4.1 summarises various commonly used kernels.

Table 4.1

Kernel	Function
Linear	$K(x, y) = (1 + x^T y)$
Polynomial	$K(x, y) = (1 + x^T y)^s$
Sigmoid	$K(x, y) = \tanh(kx^T y - \delta)$
Radial Basis Function	$K(x, y) = \exp\left(-\frac{(x-y)^2}{2\sigma^2}\right)$

Q. 5 Compare logistic regression and SVM. **(4 Marks)**

Ans. :

Table 4.2

Comparison Attribute	Logistic Regression	SVM
Good for	Linear classification	Both linear and non-linear classification
Decision boundary	Multiple	One (best one)
Approach	Statistical	Geometrical
Errors	Comparatively higher	Comparatively Lower

Q. 6 Calculate the radial distance of point A having value of 6 with respect to point B and C having value of 8 and 12 respectively. **(4 Marks)**

Ans. :

Assume $\gamma = 1$

Distance of point A with respect to

$$B = e^{-\gamma(x-y)^2} = e^{-1(6-8)^2} = e^{-4} = 0.018$$

Distance of point A with respect to

$$C = e^{-\gamma(x-y)^2} = e^{-1(6-12)^2} = e^{-36} = 0$$

So, point B has higher influence over point A than point C.

Q. 7 Demonstrate how you could use RBF over XOR function to draw a linear SVM. **(8 Marks)**

Ans. : The truth table for XOR function is as following.

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

If you plot these points, you get the layout as shown in Fig. 4.5.

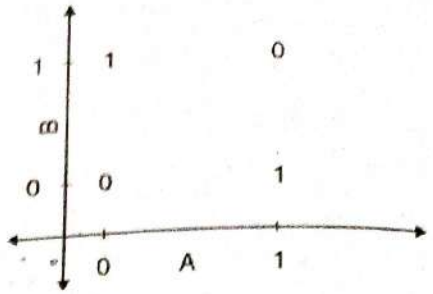


Fig. 4.5

There are two classes here $y = \{0, 1\}$ and four training samples, two for each class.

{0, 1} and {1, 0} produce 1 (+ ve class) whereas {0, 0} and {1, 1} produce 0 (– ve class). Assume $\gamma = 1$.

Let's choose elements {0, 1} and {1, 0} as the basis for RBF.

X (Input)	Basis = {0, 1}	Basis = {1, 0}
{0, 0}	$e^{-\gamma(x-y)^2} = e^{-1(\{0,0\}-\{0,1\})^2} = e^{-1(0-0\,0-1)^2} = e^{-1} = 0.37$	$e^{-\gamma(x-y)^2} = e^{-1(\{0,0\}-\{1,0\})^2} = e^{-1(0-1\,0-0)^2} = e^{-1} = 0.37$
{0, 1}	$e^{-\gamma(x-y)^2} = e^{-1(\{0,1\}-\{0,1\})^2} = e^{-1(0-0\,1-1)^2} = e^{0} = 1$	$e^{-\gamma(x-y)^2} = e^{-1(\{0,1\}-\{1,0\})^2} = e^{-1(0-1\,1-0)^2} = e^{-4} = 0.018$
{1, 0}	$e^{-\gamma(x-y)^2} = e^{-1(\{1,0\}-\{0,1\})^2} = e^{-1(1-0\,0-1)^2} = e^{-4} = 0.018$	$e^{-\gamma(x-y)^2} = e^{-1(\{1,0\}-\{1,0\})^2} = e^{-1(1-1\,0-0)^2} = e^{0} = 1$
{1, 1}	$e^{-\gamma(x-y)^2} = e^{-1(\{1,1\}-\{0,1\})^2} = e^{-1(1-0\,1-1)^2} = e^{-1} = 0.37$	$e^{-\gamma(x-y)^2} = e^{-1(\{1,1\}-\{1,0\})^2} = e^{-1(1-1\,1-0)^2} = e^{-1} = 0.37$

So, RBF does the following transformations to the respective points.

Old Point	New Point
{0, 0}	{0.37, 0.37}
{0, 1}	{1, 0.018}
{1, 0}	{0.018, 1}
{1, 1}	{0.37, 0.37}

If you plot these points, you see that they are now clearly separatable as in the case of a linear SVM.

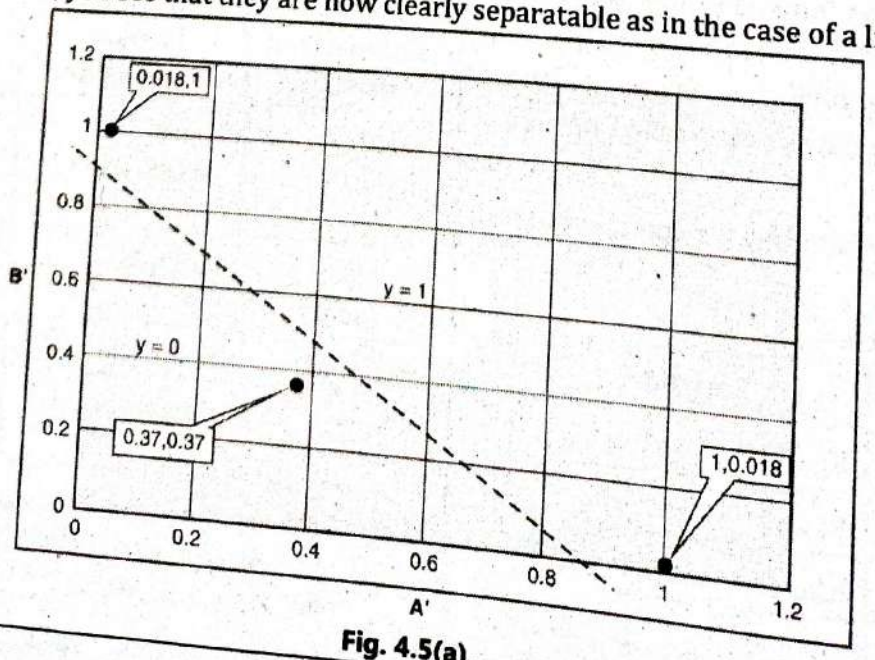


Fig. 4.5(a)

Points {1, 0.018} and {0.018, 1} give output y = 1.

Points {0.37, 0.37} and {0.37, 0.37} give output y = 0.

Hence, RBF transformed the XOR points to make them linearly separable.

Q. 8 Explain Radial Basis Function Network. **(6 Marks)**

Ans. :

- The core concept behind RBF is that inputs that are close together should generate the same output.
- Points that are close together exercise more influence over each other than the points that are far apart.
- Neural networks, for any input that you present to a set of the neurons, some of them will fire strongly, some weakly, and some will not fire at all, depending upon the distance between the weights and the particular input in weight space.
- This simply requires adding weights from each hidden (RBF) neuron to a set of output nodes. This is known as an RBF network.

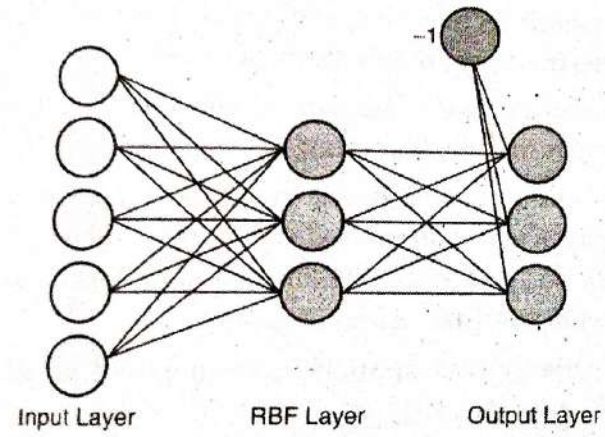


Fig. 4.6

- The Radial Basis Function network consists of input nodes connected by weights to a set of RBF neurons, which fire proportionally to the distance between the input and the neuron in weight space.
- The activations of these nodes are used as inputs to the second layer, which consists of linear nodes. RBF networks never have more than one layer of non-linear neurons.
- In an RBF network, input nodes will activate according to how close they are to the input, and the combination of these activations will enable the network to decide how to respond.

Q. 9 Write a short note on Support Vector Regression (SVR). **(4 Marks)**

Ans. :

- Support Vector Machines (SVM) are popularly and widely used for classification problems in machine learning. Support Vector Regression (SVR) uses the same principle as SVM, but for regression problems.
- The goal of a regression problem is to find a function that approximates mapping from an input domain to real numbers on the basis of a training sample.

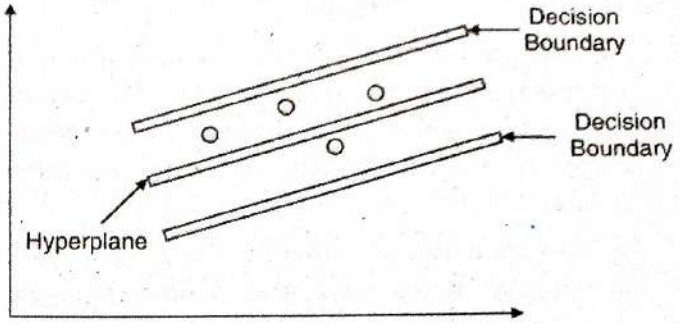


Fig. 4.7

- In case of SVM, consider the two decision boundaries and a hyperplane. Your objective is to consider the points that are within the decision boundary line. The best fit line (or regression line) is the hyperplane that has a maximum number of points. Assume that the decision boundaries are at any distance, say 'a', from the hyperplane. So, these are the lines that you draw at distance '+a' and '-a' from the hyperplane. Based on SVM, the equation of the hyperplane is as following.

$$Y = wx - b,$$

- The equations of decision boundaries are as following.

$$wx - b = a$$

$$wx - b = -a$$

- Thus, any hyperplane that satisfies SVR should satisfy $-a \leq wx - b \leq a$

Q. 10 Write a short note on multi-class classification techniques. **(4 Marks)**

Ans. :

- **Definition** : Multi-class or multinomial classification is the problem of classifying instances (data points) into one of three or more classes.
- Multi-class classification is a task with more than two classes.

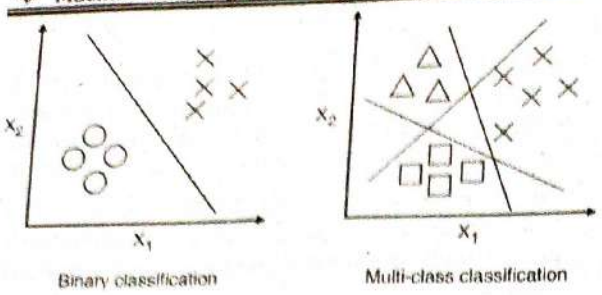


Fig. 4.8

- For example, classifying a set of images of fruits which may be oranges, apples, or pears. Multi-class classification makes the assumption that each sample is assigned to one and only one class (or label).
- For example, a fruit can either be an apple or a pear, but not both at the same time. Another example could be recognising alphabets in an optical character recognition type of problems where a given alphabet could be one of the 26 alphabets.
- Multi-class classification techniques are classified as :
  1. One vs One (OvO)
  2. One vs Rest (OvR) (Ove vs All)

Q. 11 Explain One vs One (OvO) multi-class classification technique. **(4 Marks)**

Ans. :

One vs One (OvO in short) is a heuristic method for using binary classification algorithms for multi-class classification. One vs One technique splits a multi-class classification dataset into binary classification problems. In this approach, the entire dataset is split into one dataset for each class versus every other class.

For example, consider a multi-class classification problem with four classes – 'red', 'blue', 'green', and 'yellow'. This could then be divided into six binary classification datasets as following.

- Binary Classification Problem 1 : red vs blue
- Binary Classification Problem 2 : red vs green
- Binary Classification Problem 3 : red vs yellow
- Binary Classification Problem 4 : blue vs green
- Binary Classification Problem 5 : blue vs yellow
- Binary Classification Problem 6 : green vs yellow

No. of binary datasets (or models) = $\frac{C \times (C-1)}{2}$ where C is the number of classes

Each binary classification model may predict one class label and the model with the most predictions or votes is predicted. Similarly, if the binary classification models predict a numerical class membership, such as a probability, then the argmax of the sum of the scores is predicted as the class label.

Q. 11 Explain One vs Rest (OvR) multi-class classification technique. **(4 Marks)**

Ans. :

- One vs Rest (OvR in short, also referred to as One-vs-All or OvA) is a heuristic method for using binary classification algorithms for multi-class classification.
- It involves splitting the multi-class dataset into multiple binary classification problems.
- Each binary classifier is then trained on each binary classification problem and predictions are made using the model that is the most confident.
- Basically, for each split dataset, you take one class as positive and all other classes as negative.
- For example, consider a multi-class classification problem with four classes – 'red', 'blue', 'green', and 'yellow'. This could be divided into four binary classification datasets as following.
  - Binary Classification Problem 1: red vs [blue, green, yellow]
  - Binary Classification Problem 2: blue vs [red, green, yellow]
  - Binary Classification Problem 3: green vs [red, blue, yellow]
  - Binary Classification Problem 4: yellow vs [red, blue, green]
- One machine learning model is created for each class. For example, four classes require four models. This approach requires that each model predicts a class membership probability or a probability-like score. The argmax of these scores (class index with the largest score) is then used to predict a class.
- This approach is commonly used for algorithms that naturally predict numerical class membership probability or score, such as Logistic Regression and Perceptron.

- One advantage of this approach is its interpretability. Since each class is represented by one and only one classifier, it is possible to gain knowledge about the class by inspecting its corresponding classifier. This is the most commonly used strategy and is a fair default choice.

Q. 12 Compare One vs One (OvO) and One vs Rest (OvR) multi-class classification techniques. **(4 Marks)**

Ans. :

Comparison Attribute	One vs One (OvO)	One vs Rest (OvR)
Speed	Slower than OvR	Faster than OvO
Computation Complexity	High	Low
Suitable for	Algorithms that don't scale	Algorithms that scale
No. of binary datasets or models for C classes	$\frac{C \times (C-1)}{2}$	C
Interpretability	Low	High
Used	Less commonly	More Commonly

Q. 13 What are the causes of class imbalance? **(4 Marks)**

Ans. :

- The imbalance in the class distribution may have many causes. The two main causes are as following.

1. **Sampling error**

   You might have class imbalance due to sampling errors. For example, you could have collected the data points from a very narrowly selected population or there could be bias in sampling along with data collection errors (such as incorrectly labelling samples).

2. **Problem domain**

   Based on the domain of your classification problem, there might be natural class imbalance. Suppose, that you are building a model based on weather conditions to predict next earthquake. Now, earthquake events may be very rare whereas non-earthquake events may dominate the majority of weather condition related datapoints. You cannot really do anything about it (natural events, such as an earthquake, are beyond your control) and your overall earthquake dataset might be imbalanced.

- The natural occurrence or presence of one class may dominate other classes. This may be because the process that generates observations in one class is more expensive in time, cost, computation, or other resources. As such, it is often infeasible to simply collect more samples from the domain in order to improve the class distribution.

Q. 14 Describe a few techniques to handle class imbalance. **(6 Marks)**

OR Write a short note on Tomek Links. **(4 Marks)**

OR Write a short note on SMOTE. **(4 Marks)**

Ans. :

A few techniques to handle class imbalance.

1. **Random Resampling :** Resampling is a widely used technique for handling imbalanced classes in a dataset. It consists of removing samples from the majority class (under-sampling) and/or adding more samples (artificially) from the minority class (over-sampling). The Fig. 4.9 illustrates this concept.

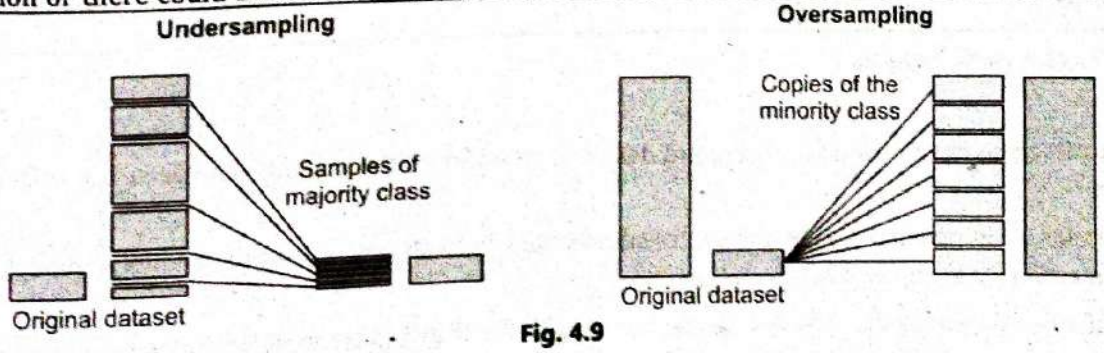


Fig. 4.9

- In under-sampling, you remove many observations of the majority class randomly. This is done until the majority and minority classes are balanced out. Under-sampling can be a good choice when you have a lot of majority class data.

- A drawback of under-sampling is that you randomly remove information which might have been useful. In over-sampling, you add more copies of data points to the minority class. Over-sampling can be a good choice when you do not have a lot of data to work with. A drawback of over-sampling is that it can cause overfitting and poor generalisation.

2. **Tomek Links :** Tomek links are pairs of very close instances but of opposite classes. Removing the instances of the majority class of each pair increases the space between the two classes and thus facilitating the classification process better. Tomek links exist if the two samples are the nearest neighbours of each other. It is an under-sampling technique to remove datapoints of the majority class. The Fig. 4.9(a) illustrates how Tomek links work.

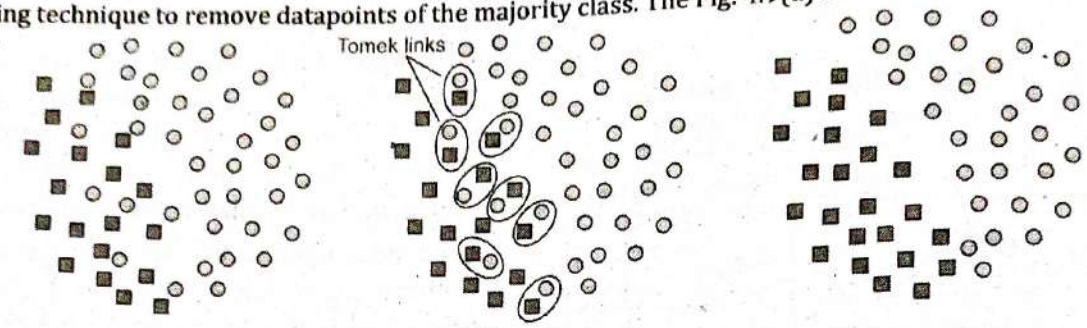


Fig. 4.9(a)

3. **SMOTE (Synthetic Minority Oversampling Technique) :** SMOTE generates synthetic (artificial) data for the minority class. It is an over-sampling technique. It consists of synthesising elements for the minority class, based on those that already exist. It works by randomly picking a point from the minority class and computing the k-nearest neighbours for this point. The synthetic points are added between the chosen point and its neighbours. The Fig. 4.9(b) illustrates how SMOTE works.

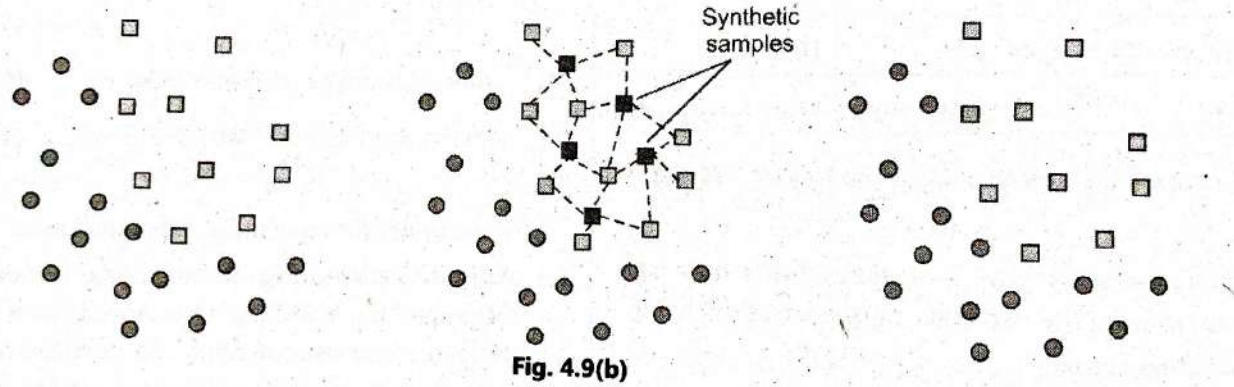


Fig. 4.9(b)

4. **Class Weights :** Most of the machine learning models provide a parameter to adjust class weights. You can use class weights to add more weightage to the minority class and less weightage to the majority class. You can also use penalised learning algorithms, such as Penalised-SVM, that increase the cost of classification mistakes on the minority class.

Q. 15 Explain the concept of Bagging.

Ans. : **(4 Marks)**

- ***Definition :*** Bagging is an ensemble technique designed to improve the stability and accuracy of classification algorithms.
- It is a short form for Bootstrap aggregating. It reduces bias and variance from the predictive models. The way bootstrap method (bagging) works is as following.
- A sample of 100 values (say distribution *d*), and you would like to get an estimate of the mean of the sample. You could simply calculate the mean directly from the sample as

$$\text{Mean (d)} = \frac{\text{Sum of values in d}}{100}$$

- Your samples have very low as well as very high values. Since the sample size is small it could be that your mean calculation might not be an accurate representation of the distribution. You could improve the estimate of the mean using the bootstrap procedure by

1. Creating many (say 1000) random sub-datasets of your original dataset. You may select a datapoint in various sub-datasets multiple times.
2. Calculating the mean of each sub-dataset
3. Calculating the average of all the means for sub-datasets

Q. 16 Write a short note on Stumping. **(4 Marks)**

Ans. :

- There is a very extreme form of boosting that is applied to trees. In real world, the stump of a tree is the tiny piece that is left over when you chop off the rest of the tree.
- In machine learning, stumping consists of simply taking the root of the tree and using that as the decision maker. So, for each classifier, use the very first question that makes up the root of the tree, and that is it. The decision tree is not further grown. It is also called as a decision stump.
- **Definition :** A decision stump is a machine learning model consisting of a one-level decision tree.
- It is a decision tree with one internal node (the root) which is immediately connected to the terminal nodes (its leaves). A decision stump makes a prediction based on the value of just a single input feature.
- Following is an example of a decision stump.

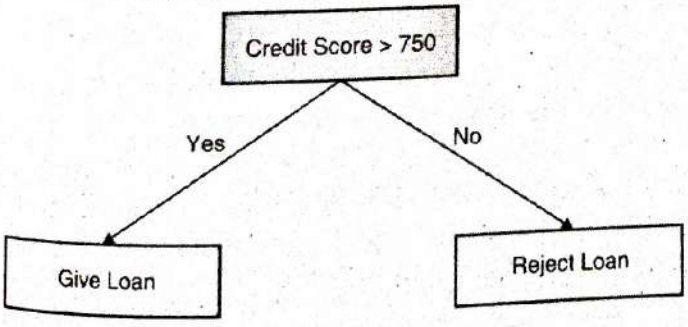


Fig. 4.10

- The overall performance of stumping can be very high if the dataset is majorly skewed around a particular feature.

Q. 17 Explain AdaBoost algorithm. **(6 Marks)**

Ans. :

- AdaBoost is the short name used for adaptive boosting. This boosting method is adaptive in the sense that it uses a concept of weights to train the classifier model and adjusts these weights dynamically based on how the model performs in subsequent iterations of training.
- At a high-level, the AdaBoost algorithm works as following.

1. A weight is applied to every example in the training data. Initially, these weights are all equal.
2. A weak classifier is first trained on the training data. The errors from the weak classifier are calculated, and the weak classifier is trained a second time with the same dataset.
3. The second time the weak classifier is trained, the weights of the training set are adjusted such that the examples properly classified the first time are weighted less and the examples incorrectly classified in the first iteration are weighted more.
4. To get uniform output from all of these weak classifiers, AdaBoost assigns α values to each of the classifiers. The α values are based on the error of each weak classifier. The error $\in$ is given as

$$\in = \frac{\text{number of incrorrectly classified examples}}{\text{total number of examples}}$$

α is calculated as

$$\alpha = \frac{1}{2}\ln\left(\frac{1-\in}{\in}\right)$$

5. After you calculate α, you can update the weight vector D so that the examples that are correctly classified will decrease in weight and the misclassified examples will increase in weight.

If the classification was correct, then D is calculated as

$$D_i^{t+1} = \left(D_i^t \frac{e^{-\alpha}}{\text{Sum (D)}}\right)$$

Else if the classification was incorrect, then D is calculated as

$$D_i^{t+1} = \left(D_i^t \frac{e^{\alpha}}{\text{Sum (D)}}\right)$$

6. After D is calculated, AdaBoost starts on the next iteration. The AdaBoost algorithm repeats the training and weight-adjusting iterations until the training error is 0or until the number of weak classifiers reaches a user-defined value.

The AdaBoost algorithm is illustrated as following.

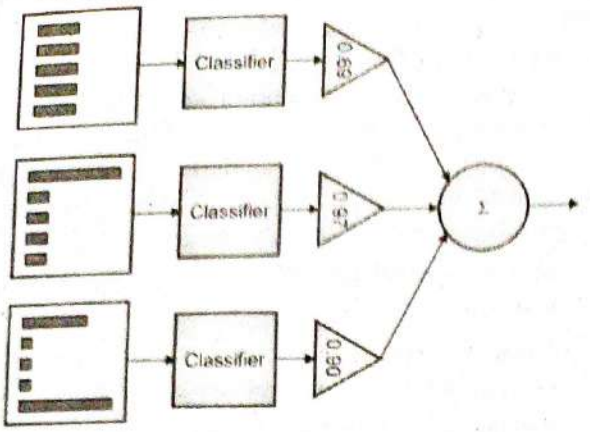


Fig. 4.11

Q. 18 Compare bagging and boosting. **(4 Marks)**

Ans. :

Table 4.3

Comparison Attribute	Bagging	Boosting
Weak models work	Parallelly	Sequentially
All models have	Equal weight	Adjusted weights
Reduces	Variance	Bias
Overfitting	Addressed	Not addressed
Merge predictions of	Same type	Different types

Q. 19 Write a short note on Random Forests. **(4 Marks)**

Ans. :

- ***Definition :*** Random Forests is an ensemble technique designed to combine several decision trees to reduce errors and to build a more accurate prediction model.

In random forest, instead of building one decision tree, multiple decision trees are built.

1. Each tree in the forest is built using random datapoints drawn from the dataset.
2. The trees in the forest are split such that a fixed subset of input variables (attributes) are taken each time and the best possible split is performed. For example, if there are 5 attributes in the dataset, you could choose to take 2 attributes at a time and split based on those attributes to create a tree in the forest.
3. Finally, the result from each tree is aggregated to give the outcome. Usually, you count the total number of votes (based on the classification outcome from each tree in the forest) for classification problems and average (based on the outcome value from each tree in the forest) for regression problems.

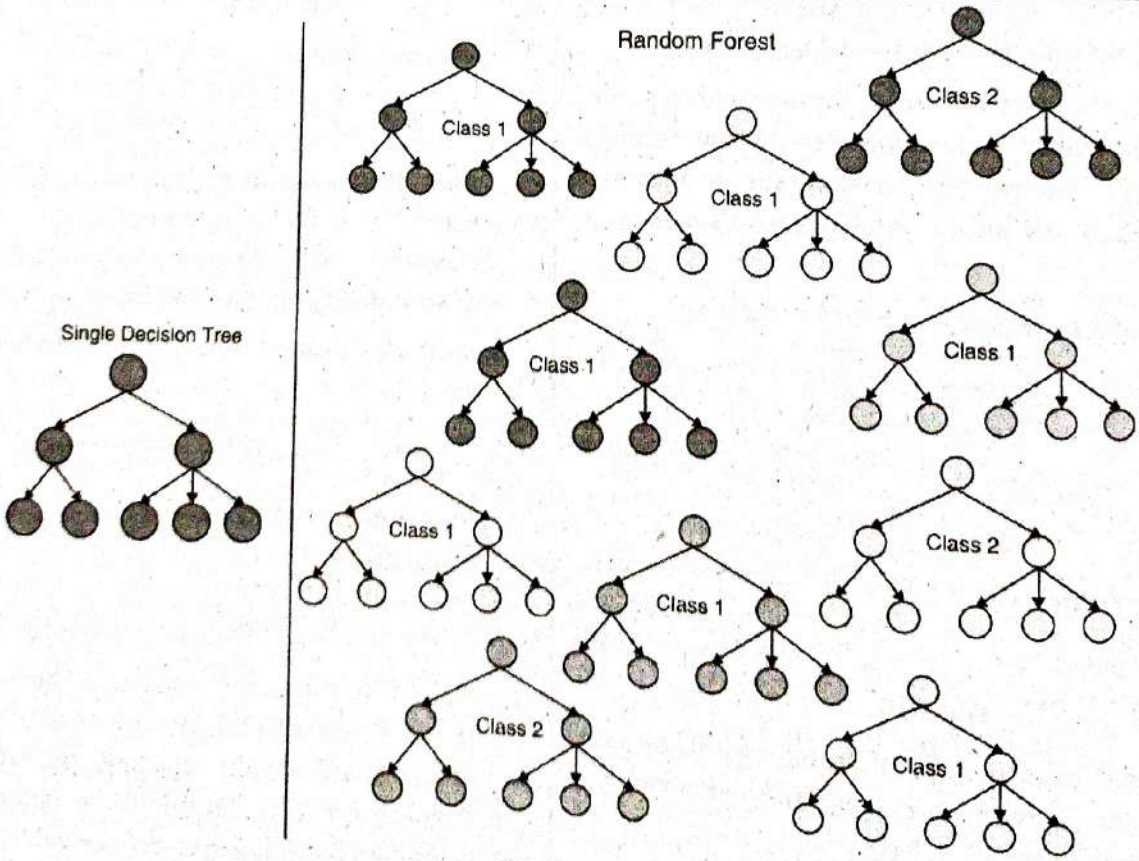


Fig. 4.12

Q. 20 Explain confusion matrix with an example. **(6 Marks)**

Ans. :

- ***Definition*** : A confusion matrix lays out the comparison between the predicted classification and actual classification to provide various performance measures.
- The number of correct and incorrect predictions are written for each class. Confusion matrix provides calculations for various types of errors in prediction and helps you to improve your classification model.
- Following is how a confusion matrix looks like for two possible classifications.

Confusion Matrix for Class 1 and Class 2		Actual Class	
		Class 1	**Class 2**
Predicted Class	**Class 1**	**True Positives (TP)**	False Positives (FP)
	Class 2	False Negative (FN)	**True Negative (TN)**

- **True Positive** = Correct True Prediction for Class 1. This is desired.

Predicted Class = Actual Class : Correct classification

- **True Negative** = Correct False Prediction for Class 1 (True Prediction for Class 2). This is desired.

Predicted Class = Actual Class : Correct classification

- **False Positive** = Incorrect True Prediction for Class 2. This is Type I error.

Predicted Class ≠ Actual Class : Incorrect classification

- **False Negative** = Incorrect False Prediction for Class 2. This is Type II error.

Predicted Class ≠ Actual Class : Incorrect classification

For example, assume that you have built a model for fruit classifier that can classify the pictures into banana and apple. Assume that there were 13 pictures of fruits that were classified using your model. In reality, there were 8 banana pictures and 5 apple pictures. A confusion matrix could look like the following.

Confusion Matrix for Apples and Bananas		Actual Class	
		Banana	**Apple (Not Banana)**
Predicted Class	**Banana**	5	2
	Apple (Not Banana)	3	3

So, the confusion matrix tells you that

- Your model correctly classified 5 out of 13 pictures as banana. (True Positive)
- Your model correctly classified 3 out of 13 pictures as apple (which were not banana) pictures. (True Negative)
- Your model incorrectly classified 2 out of 13 pictures. They were actually apples which were classified as bananas. (False Positive)
- Your model incorrectly classified 3 out of 13 pictures. They were actually bananas which were classified as apples. (False Negative)

So,

Overall 8 out of 13 classifications were correct.

Overall 5 out of 13 classifications were incorrect.

Q. 21 For the following confusion matrix, calculate accuracy, TPR and FPR. **(6 Marks)**

Confusion Matrix for Apples and Bananas		Actual Class	
		Banana	**Apple (Not Banana)**
Predicted Class	**Banana**	**5**	2
	Apple (Not Banana)	3	**3**

Ans. : Based on the given confusion matrix

- True Positive (TP) = 5
- True Negative (TN) = 3
- False Positive (FP) = 2
- False Negative (FN) = 3

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \times 100$$

$$\text{Accuracy} = \frac{5+3}{5+3+2+3} \times 100 = \frac{10}{13} \times 100 = 76.92\ \%$$

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

$$TPR = \frac{5}{5+3} = \frac{5}{8} = 0.625$$

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

$$FPR = \frac{2}{2+3} = \frac{2}{5} = 0.4$$

Q. 22 For the following confusion matrix, calculate F1 score and MCC. What would you comment about the classifier based on these values? **(6 Marks)**

TP = 100	FP = 10
FN = 5	TN = 50

Ans. :

$$\text{F1 Score} = \frac{2TP}{2TP + FP + FN}$$

$$\text{F1 Score} = \frac{2 \times 100}{2 \times 100 + 10 + 5} = \frac{200}{215} = 0.93$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

$$MCC = \frac{100 \times 50 - 10 \times 5}{\sqrt{(100 + 10)(100 + 5)(50 + 10)(50 + 5)}}$$

$$= \frac{4950}{6173.73} = 0.80$$

The values of F1 score and MCC are very close to 1. These performance measures suggest that the classifier's accuracy is pretty good.

Q. 23 Following are the scores for a multi-class classification model. Find out micro and macro precision, recall, and F1-score. **(6 Marks)**

Class	TP	FP	FN	Precision	Recall
A	5	2	1	0.71	0.83
B	10	90	7	0.1	0.58
C	15	11	2	0.57	0.88

Ans. :

Micro-average precision

$$= \frac{TP_1 + TP_2 \ldots TP_n}{TP_1 + TP_2 \ldots + TP_n + FP_1 + FP_2 \ldots + FP_n}$$

$$\text{Micro-average precision} = \frac{5 + 10 + 15}{5 + 10 + 15 + 2 + 90 + 11}$$

$$= \frac{30}{133} = 0.22$$

$$\text{Macro-average precision} = \frac{P_1 + P_2 + \ldots P_n}{n}$$

$$\text{Macro-average precision} = \frac{0.71 + 0.1 + 0.57}{3} = \frac{1.38}{3}$$

$$= 0.46$$

Micro-average recall

$$= \frac{TP_1 + TP_2 \ldots TP_n}{TP_1 + TP_2 \ldots + TP_n + FN_1 + FN_2 \ldots + FN_n}$$

$$\text{Micro-average Recall} = \frac{5 + 10 + 15}{5 + 10 + 15 + 1 + 7 + 2}$$

$$= \frac{30}{40} = 0.75$$

$$\text{Macro-average Recall} = \frac{R_1 + R_2 + \ldots R_n}{n}$$

$$\text{Macro-average Recall} = \frac{0.83 + 0.58 + 0.88}{3} = \frac{2.29}{3}$$

$$= 0.76$$

$$\text{Micro-average F1 score} = \frac{TP_1 + TP_2 \ldots TP_n}{TP_1 + TP_2 + TP_n + \frac{1}{2}(FP_1 + FP_2 + \ldots FP_n + FN_1 + FN_2 + \ldots FN_n)}$$

Micro-average F1 score

$$= \frac{5 + 10 + 15}{5 + 10 + 15 + \frac{1}{2}(2 + 90 + 11 + 1 + 7 + 2)}$$

$$= \frac{30}{86.5} = 0.35$$

$$\text{F1 score} = \frac{2TP}{2TP + FP + FN}$$

$$\text{F1score}_1 = \frac{2 * 5}{2 * 5 + 2 + 1} = \frac{10}{13} = 0.77$$

$$\text{F1score}_2 = \frac{2 * 10}{2 * 10 + 90 + 7} = \frac{20}{117} = 0.17$$

$$\text{F1score}_3 = \frac{2 * 15}{2 * 15 + 11 + 2} = \frac{30}{43} = 0.70$$

Macro-average F1score

$$= \frac{\text{F1score}_1 + \text{F1score}_2 + \ldots \text{F1score}_n}{n}$$

$$\text{Macro-average F1 score} = \frac{0.77 + 0.17 + 0.70}{3}$$

$$= \frac{1.64}{3} = 0.55$$

Q. 24 Write a short note on ROC Curve. **(4 Marks)**

Ans. :

- ROC is an acronym for Receiver Operating Characteristic.
- ***Definition :*** A ROC curve evaluates the performance of a classifier based on the True Positives (TP) and False Positives (FP) regardless of other factors such as class distribution and error costs.
- On the Y-axis, you plot the True Positive Rate (TPR) and on the X-axis you plot the False Positive Rate (FPR). As you adjust your classifier model, a ROC curve can help you to visualise all possible classification thresholds.
- A classifier that does a good job will have ROC curve that is bulging outside the central ROC = 0.5 line. ROC = 0.5 line represents a classifier that is as good as someone randomly guessing (TPR = FPR).

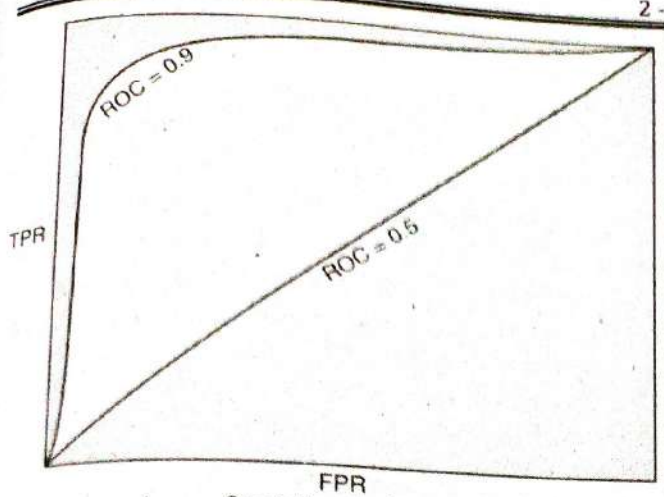


Fig. 4.13

- A classifier that does not classify well will have its ROC curve plot very close to ROC = 0.5 line.

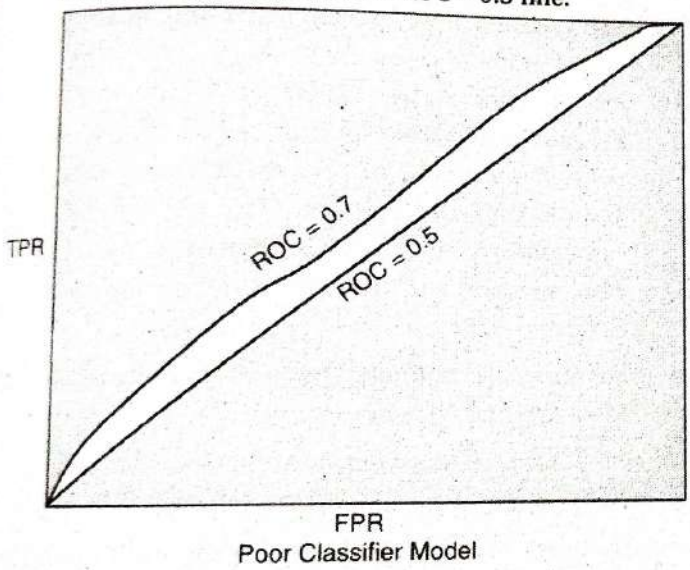
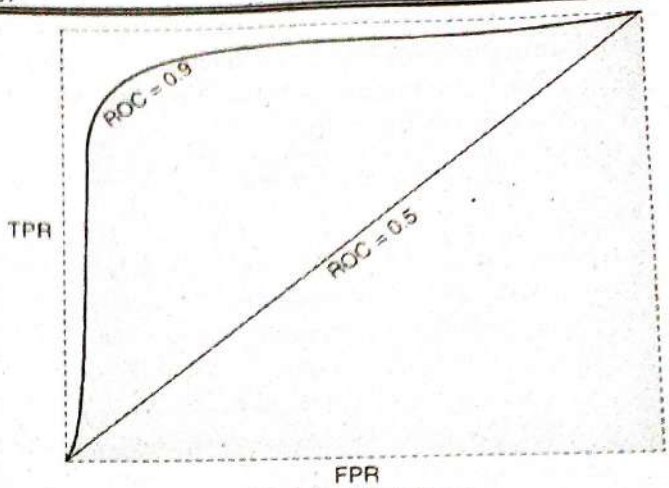


Fig. 4.13(a)

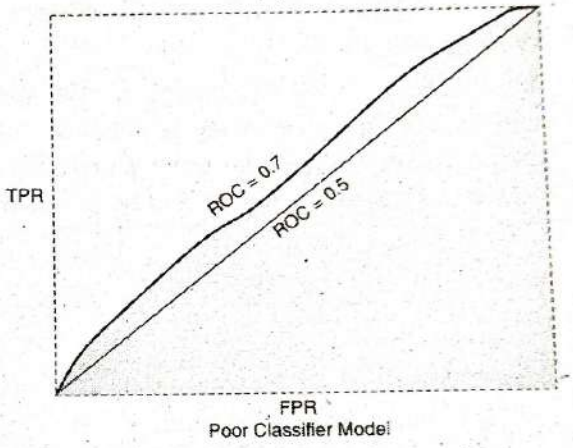
Q. 25 Write a short note on AUC. **(4 Marks)**

Ans. :

- ***Definition :*** Area Under the Curve (AUC) is calculated by measuring the area under the ROC curve.
- Imagine an illusory square box around the ROC curve plot. AUC represents the area or the amount of box that the ROC curve fills in.
- The higher the area filled, the better is the classifier. The score can range from 0.5 (for the diagonal line TPR = FPR) to 1.0 (with ROC passing through the top-left corner).



(a)



(b)

Fig. 4.14

Q. 26 What is cross-validation? **(4 Marks)**

Ans. :

- **Definition :** Cross-validation is a technique for evaluating the performance of machine learning models by training several machine learning models on subsets of the available input data and evaluating them on the complementary subset of the data.
- At a high-level, from a given dataset, you randomly create training and testing samples and test out various models to figure out which model works best. You can use cross-validation to detect overfitting. There are various popular techniques for performing cross-validation. Broadly speaking, they could be classified into two types as following.

1. **Non-exhaustive :** In this method, you split the original dataset and leave out a subset for validation. You do not attempt to split the dataset in all possible ways.

2. **Exhaustive :** In this method, you split the original data set in all possible ways into a training and a validation set and carry out multiple iterations of cross-validation.

Q. 27 Describe k-fold cross-validation. **(6 Marks)**

Ans. :

- k-Fold cross-validation is another non-exhaustive approach for cross-validation. In k-fold cross-validation technique, you split the input data into k subsets of data (also known as folds). You train a machine learning model on all but one (k – 1) of the subsets, and then evaluate the model on the subset that was not used for training.
- This process is repeated k times, with a different subset reserved for evaluation (and excluded from training) each time.
- The Fig. 4.15 shows an example of the training subsets and complementary evaluation subsets generated for each of the four models that are created and trained during a 4-fold cross-validation.

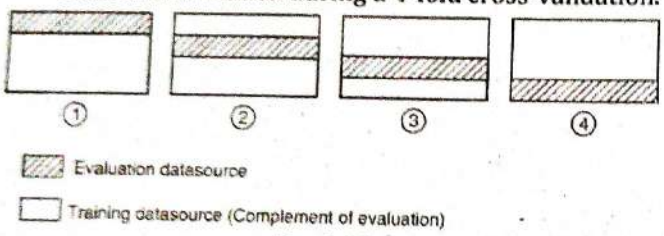


Fig. 4.15

- Model one uses the first 25 percent of data for evaluation, and the remaining 75 percent for training. Model two uses the second subset of 25 percent (25 percent to 50 percent) for evaluation, and the remaining three subsets of the data for training, and so on. Each model is trained and evaluated using complementary datasources - the data in the evaluation datasource includes and is limited to all of the data that is not in the training datasource.
- Performing a 4-fold cross-validation generates four models, four datasources to train the models, four datasources to evaluate the models, and four evaluations, one for each model. You can generate a model performance metric for each evaluation. For example, in a 4-fold cross-validation for a binary classification problem, each of the evaluations could report an area under curve (AUC) metric. You can get the overall performance measure by computing the average of the four AUC metrics.
- k-Fold cross-validation approach has two major problems.
  1. To keep the training set large, you need to allow validation sets that are small.
  2. The training sets overlap considerably, namely, any two training sets share k – 2 parts.
- k is typically set to 10 or 30. As k increases, the percentage of training instances increases and you get more robust estimators, but the validation set becomes smaller. Furthermore, there is the computation cost of training the classifier k times, which increases as k is increased.

Q. 28 Describe Leave-P-Out Cross-Validation (LpOCV). **(6 Marks)**

Ans. :

- Leave-P-Out Cross-Validation is an exhaustive approach for cross-validation. In this method, you take p number of points out from the total number of data points in the dataset (say n). While training the model, you train it on these (n – p) data points and test the model on the remaining p data points. You repeat this process for all the possible combinations of p from the original dataset. Finally, you average out the accuracies from all the iterations of p.
- For example, suppose that you have a dataset as following and you choose p = 2.

Fruit = {apple, orange, guava, grapes, papaya, banana}

- For each iteration, you keep 2 data points out of the training dataset. So, to ensure all possible combinations of p, you could split the dataset and iterations as following :

Iteration 1 – Training = {guava, grapes, papaya, banana} validation = {apple, orange}

Iteration 2 – Training = {apple, grapes, papaya, banana} validation = {orange, guava}

Iteration 3 – Training = {apple, orange, papaya, banana} validation = {guava, grapes}

Iteration 4 – Training = {apple, orange, guava, banana} validation = {grapes, papaya}

Iteration 5 – Training = {apple, orange, guava, grapes} validation
= {papaya, banana}

Iteration 6 – Training = {orange, guava, grapes, papaya} validation
= {banana, apple}
[various others] ...

- The total number of possible combinations is given by the well-known combination formula as following.

$$C = \frac{n!}{p!\,(n-p)!}$$

So, for the example that I gave earlier, there could be 15 combinations!

$$C = \frac{6!}{2!\,(6-2)!} = 15$$

- This is a very simplistic example. But, note that for large datasets, this method can become computationally infeasible. For example, with n = 100 and p = 30 there would be approximately 3×10^{25} combinations! A variant of LpOCV with p = 2, known as leave-pair-out cross-validation, has been recommended as a nearly unbiased method for practical use. Another variant, where p = 1 is called Leave-one-out Cross-Validation (LOOCV). This makes the method much less exhaustive as now for n data points and p = 1, you have at max n number of combinations.

❑❑❑

Chapter 5 : Unsupervised Learning

Q. 1 Write a short note on clustering. **(4 Marks)**

Ans. :

- **Definition :** Clustering is a technique in which the data points are arranged in similar groups dynamically without any pre-assignment of groups.
- It is the task in which the data points are grouped in such a way that objects in the same group (called a cluster) are more similar to each other than to those in other groups (clusters).
- It is a data exploration technique commonly used to understand how the data should be interpreted and grouped to be best analysed. There is no prior learning of groupings required. Instead, groups are implicitly arranged (or created) based on the data attributes.
- The same dataset can be used to form various clusters depending upon the data attributes and then you can decide which clusters make sense to be used for further data analytics.
- For example, here is a simple plot of data points. Some set of points are closer to each other when compared with others. These sets could possibly form a group (or a cluster) for further data analytics.

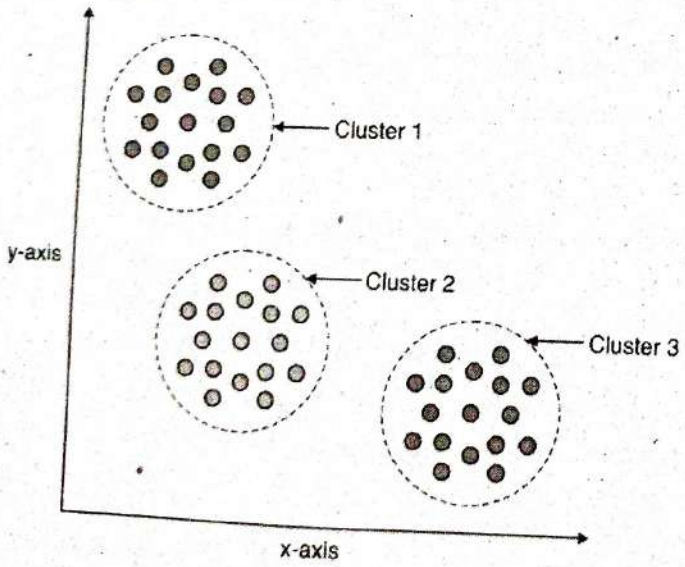


Fig. 5.1

- Clustering is used in recommendation and search engines, marketing, economics, and other branches of science.

Q. 2 Explain the properties of a cluster. **(4 Marks)**

Ans. :

Typically clusters have the following properties.

1. **All the data points in a cluster should be similar to each other**

   By definition, a cluster is a grouping of similar objects. So, a good cluster must have data points that indeed have some convincing similarities on the basis of which they are grouped. For example, phone preference cluster mentioned in the example in the previous section could be a convincing enough cluster.

2. **The data points from different clusters should be as different as possible**

   To make clusters distinct enough, the data points from different clusters should be far apart or, in other words, must be clearly distinguishable. For example, a cluster containing all customers whose preference is iPhone Model X is clearly distinguishable from another cluster containing customers preferring Motorola X.

Q. 3 Explain types of clustering. **(4 Marks)**

Ans. :

At a high-level, there are two types of clustering.

1. **Hard Clustering** : In this, each data point belongs to only one cluster at a time. For example, customer A (in the previously given example) could only be in iPhone X cluster if you clustered based on phone model preference.
2. **Soft Clustering :** In this, instead of putting each data point into a separate cluster, a probability or likelihood of that data point to be in those clusters is assigned. For example, customer A would be in both the clusters -iPhone X and Motorola X with respective probabilities of 1 and 0.

Q. 4 Explain 3 Applications of Clustering. **(6 Marks)**

Ans. :

Some of the common applications of clustering are as following.

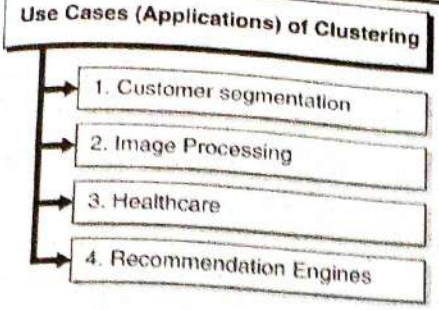


Fig. 5.2

1. Customer Segmentation

- One of the most common applications of clustering is customer segmentation. Various companies build customer segment profiles to understand the customer's shopping behaviour and accordingly build strategies for predicting demand and improving sales.
- Various customers could be clustered to understand their purchase behaviours and appropriate and targeted marketing campaign could be run for different segments. For example, customers who own iPhone Model X could be shown other iPhone related accessories or other products from the same company.

2. Image Processing

- Using clustering technique, you could identify objects in an image. You can cluster similar pixels in the image together and then match that with known objects to identify the objects in the image.

3. Healthcare

- Patient data, containing attributes such as blood pressure, height, weight, cholesterol level and glucose level, could be used to create clusters and detect any early signs of health problems based on historical records of other patients who had similar characteristics and then they were diagnosed with a particular health problem. For example, if someone has high cholesterol level and weight, it might be possible to detect if she is closer to getting a heart attack. There could be several other uses of clustering in the healthcare industry.

4. Recommendation Engines

- Video or song recommendations on favourite media apps or the product recommendations as you browse through an e-commerce portal.

Q. 5 Explain the steps involved in the K-means algorithm. **(6 Marks)**

Ans. :

- At a high-level, the following steps are taken for clustering the data using K-means clustering algorithm.

1. Decide the number of clusters, k, that you desire to group your data points into.
2. Select k random data points as centroids.
3. Compute the distance from each data point to each centroid. Assign all the data points to the closest cluster centroid. The distance d between any two points, (x_1, y_1) and (x_2, y_2) is calculated as

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

4. Recompute the centroids of newly formed clusters. The centroid (X_c, Y_c) of the m data points in a cluster is calculated as

$$(X_c, Y_c) = \left(\frac{\sum_{i=1}^{m} X_i}{m}, \frac{\sum_{i=1}^{m} Y_i}{m} \right)$$

It is a simple arithmetic mean of all X coordinates and Y coordinates of the m data points in the cluster.

5. Repeat steps 3 and 4 until any of the following criteria is met
   a. Centroids of newly formed clusters do not change.
   b. Points remain in the same cluster.
   c. Maximum number of iterations are reached as desired.

Q. 6 Use the following data and group them using K-means clustering algorithm. Show calculation of centroids. **(6 Marks)**

Height	Weight
185	72
170	56
168	60
179	68
182	72
188	77
180	71
180	70
183	84
180	88
180	67
177	76

Ans. :

Height on X-axis and weight on Y-axis. Hence, k = 2. Let's decide that we would do centroid calculation and data points to cluster assignment twice (number of iterations) in the algorithm. The first plot of the given data points is as shown in Fig. 5.3

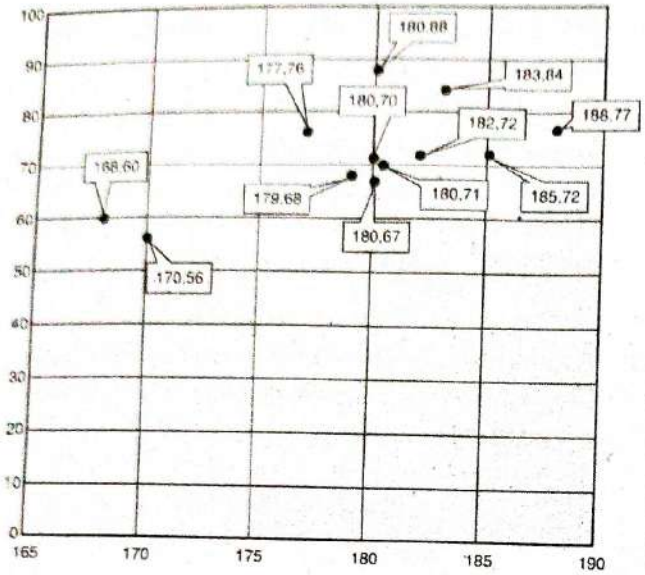


Fig. 5.3

Iteration 1

Let's randomly choose two centroids.

Centroid 1 = (170, 56) and Centroid 2 = (182, 72).

Now, let's calculate the distance of the data points from the chosen centroids and complete the data points table. The data point is assigned to the cluster based on the closest centroid.

A sample distance calculation is as following for the first data point.

Distance from Centroid 1 (170, 56) for (185, 72) =

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

$$d = \sqrt{(170 - 185)^2 + (56 - 72)^2}$$

$$d = 21.93$$

Height	Weight	Distance from Centroid 1	Distance from Centroid 2	Assigned Cluster
185	72	21.93	3.00	2
170	56	0.00	20.00	1
168	60	4.47	18.44	1
179	68	15.00	5.00	2
182	72	20.00	0.00	2
188	77	27.66	7.81	2
180	71	18.03	2.24	2
180	70	17.20	2.83	2
183	84	30.87	12.04	2
180	88	33.53	16.12	2
180	67	14.87	5.39	2
177	76	21.19	6.40	2

Let's re-calculate the centroids for the next iteration.

For Centroid 1 calculation, you have two data points that fall in cluster 1. They are (170, 56) and (168, 60).

Hence, Centroid 1 is the mean of the data points which is $\left(\frac{170 + 168}{2}, \frac{56 + 60}{2}\right)$ = (169, 58).

For Centroid 2 calculation, take the remaining 10 data points that fall in cluster 2.

Hence, Centroid 2 is the mean of these 10 data points which is

$$\left(\frac{185+179+182+188+180+180+183+180+180+177}{10}, \frac{72+68+72+77+71+70+84+88+67+76}{10}\right)$$

= (181.4, 74.5)

Now, you have both the centroids ready for the next and final iteration.

Iteration 2

Centroid 1 = (169, 58) and Centroid 2 = (181.4, 74.5).

Now, let's calculate the distance of the data points from the centroids and complete the data points table. The data point is assigned to the cluster based on the closest centroid.

Height	Weight	Distance from Centroid 1	Distance from Centroid 2	Assigned Cluster
185	72	21.26	4.38	2
170	56	2.24	21.73	1
168	60	2.24	19.74	1
179	68	14.14	6.93	2
182	72	19.10	2.57	2
188	77	26.87	7.06	2
180	71	17.03	3.77	2
180	70	16.28	4.71	2
183	84	29.53	9.63	2
180	88	31.95	13.57	2
180	67	14.21	7.63	2
177	76	19.70	4.65	2

You stop here because the number of iterations that you decided are completed and also you see that the clusters assigned in the iteration 1 for the data points did not change.

Hence, you got the two clusters as shown in Fig. 5.3(a).

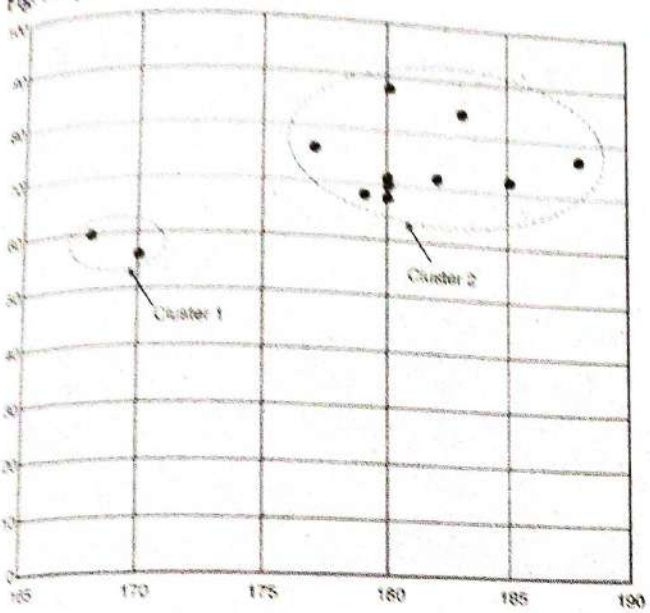


Fig. 5.3(a)

Q. 7 In a survey, the following data was collected for TV viewership. Is there a group which watches more TV than the other? **(6 Marks)**

Age	Number of TV watching hours
23	3
25	2
28	2
35	4
37	5
40	4
34	3
41	5
39	4
38	3

Ans. : Use K-means clustering for grouping the survey results. Let's choose the desired number of clusters, k = 2 and the number of iterations for centroid calculation and and the number of iterations for centroid calculation and cluster assignment to be 2 as well. Let's put age on X-axis and TV watching hours on Y-axis.

The first plot of data points is as shown in Fig. 5.4.

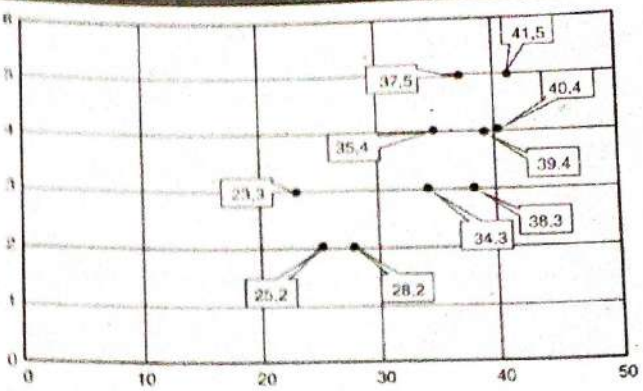


Fig. 5.4

Iteration 1

Let's randomly choose two centroids.

Centroid 1 = (25, 2) and Centroid 2 = (35, 4).

Now, let's calculate the distance of the data points from the chosen centroids and complete the data points table. The data point is assigned to the cluster based on the closest centroid.

A sample distance calculation is as following for the first data point.

Distance from Centroid 1 (25, 2) for (23, 3) =

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

$$d = \sqrt{(25 - 23)^2 + (2 - 3)^2}$$

$$d = 2.24$$

Age	Hours	Distance from Centroid 1	Distance from Centroid 2	Assigned Cluster
23	3	2.24	12.04	1
25	2	0.00	10.20	1
28	2	3.00	7.28	1
35	4	10.20	0.00	2
37	5	12.37	2.24	2
40	4	15.13	5.00	2
34	3	9.06	1.41	2
41	5	16.28	6.08	2
39	4	14.14	4.00	2
38	3	13.04	3.16	2

Now re-calculate the centroids for the next iteration.

For Centroid 1 calculation, you have three data points that fall in cluster 1. They are (23, 3), (25, 2) and (28, 2).

Hence, Centroid 1 is the mean of the data points which is $\left(\frac{23 + 25 + 28}{3}, \frac{3 + 2 + 2}{3}\right)$ = (25.33, 2.33).

For Centroid 2 calculation, take the remaining seven data points that fall in cluster 2.

Hence, Centroid 2 is the mean of these seven data points which is

$$\left(\frac{35+37+40+34+41+39+38}{7}, \frac{4+5+4+3+5+4+3}{7}\right)$$

$$= (37.71, 4)$$

Now, you have both the centroids ready for the next and final iteration.

Iteration 2

Centroid 1 = (25.33, 2.33) and Centroid 2 = (37.71, 4).

Now, let's calculate the distance of the data points from the centroids and complete the data points table. The data point is assigned to the cluster based on the closest centroid.

Age	Hours	Distance from Centroid 1	Distance from Centroid 2	Assigned Cluster
23	3	2.42	14.74	1
25	2	0.47	12.87	1
28	2	2.69	9.91	1
35	4	9.81	2.71	2
37	5	11.97	1.23	2
40	4	14.76	2.29	2
34	3	8.70	3.84	2
41	5	15.90	3.44	2
39	4	13.77	1.29	2
38	3	12.69	1.04	2

You stop here because the number of iterations that you decided are completed and also you see that the clusters assigned in the iteration 1 for the data points did not change.

Hence, you got the two clusters as shown in Fig. 5.4(a).

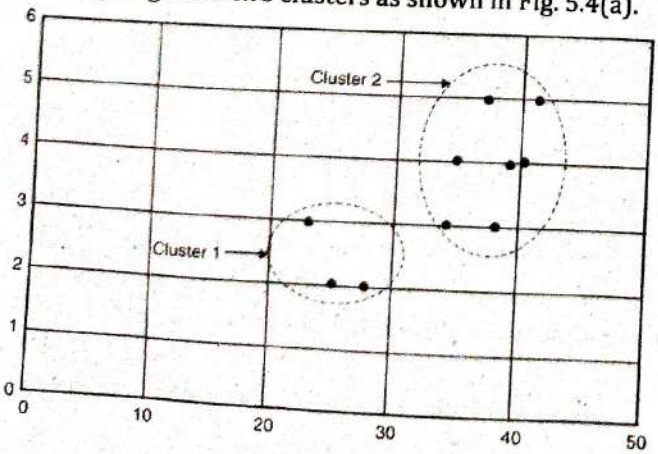


Fig. 5.4(a)

From the clusters formed, it can be concluded that people in cluster 2 seem to be watching more TV than people in cluster 1.

Q. 8 A bank has received the following loan applications. Which of the applications could be risky to approve? **(6 Marks)**

Credit Score (out of 1000)	Amount in Lakhs
500	10
726	25
430	5
678	15
780	30
380	10
645	15
890	50
900	65
450	10

Ans. :

Let us use K-means clustering for grouping the loan applications. Let's choose the desired number of clusters, k = 2 and the number of iterations for centroid calculation and cluster assignment to be 2 as well. Let's put credit score on X-axis and loan amount on Y-axis.

The first plot of data points is as shown in Fig. 5.5.

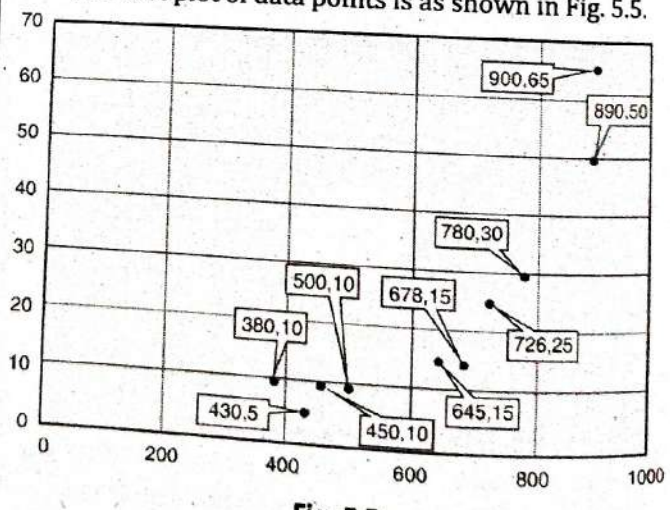


Fig. 5.5

Iteration 1

Let's randomly choose two centroids.

Centroid 1 = (430, 5) and Centroid 2 = (726, 25).

Now, let's calculate the distance of the data points from the chosen centroids and complete the data points table. The data point is assigned to the cluster based on the closest centroid.

A sample distance calculation is as following for the first data point.

Distance from Centroid 1 (430, 5) for (500, 10) =

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

$$d = \sqrt{(430 - 500)^2 + (5 - 10)^2}$$

$$d = 70.18$$

Credit Score	Amount	Distance from Centroid 1	Distance from Centroid 2	Assigned Cluster
500	10	70.18	226.50	1
726	25	296.67	0.00	2
430	5	0.00	296.67	1
678	15	248.20	49.03	2
780	30	350.89	54.23	2
380	10	50.25	346.32	1
645	15	215.23	81.61	2
890	50	462.20	165.89	2
900	65	473.81	178.54	2
450	10	20.62	276.41	1

Now re-calculate the centroids for the next iteration.

For Centroid 1 calculation, you have four data points that fall in cluster 1.

Hence, Centroid 1 is the mean of the data points which is

$$\left(\frac{500 + 430 + 380 + 450}{4}, \frac{10 + 5 + 10 + 10}{4}\right) = (440, 8.75).$$

For Centroid 2 calculation, take the remaining six data points that fall in cluster 2.

Hence, Centroid 2 is the mean of these six data points which is

$$\left(\frac{726 + 678 + 780 + 645 + 890 + 900}{6}, \frac{25 + 15 + 30 + 15 + 50 + 65}{6}\right)$$

$= (769.83, 33.33)$

Now, you have both the centroids ready for the next and final iteration.

Iteration 2

Centroid 1 = (440, 8.75) and Centroid 2 = (769.83, 33.33).

Now, let's calculate the distance of the data points from the centroids and complete the data points table. The data point is assigned to the cluster based on the closest centroid.

Credit Score	Amount	Distance from Centroid 1	Distance from Centroid 2	Assigned Cluster
500	10	60.01	270.84	1
726	25	286.46	44.61	2
430	5	10.68	341.01	1
678	15	238.08	93.64	2
780	30	340.66	10.70	2
380	10	60.01	390.53	1
645	15	205.10	126.17	2
890	50	451.89	121.32	2
900	65	463.43	133.97	2
450	10	10.08	320.68	1

Hence, you got the two clusters as shown in Fig. 5.5(a).

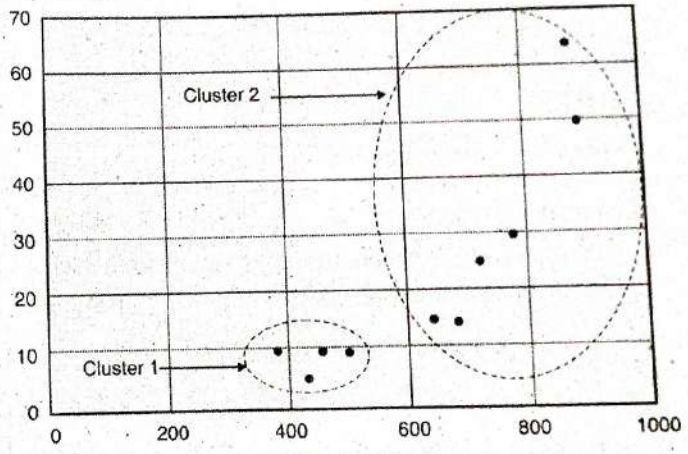


Fig. 5.5(a)

From the clusters formed, it can be concluded that people in cluster 2 seem to be less risky for granting loan.

Q. 9 A cluster has the following data points. Calculate its inertia. Assume 2nd data point to be the centroid for the cluster. **(6 Marks)**

Age	Income in Thousand
33	12
33	15
35	13
34	14
32	16

Ans :

To calculate inertia of a cluster, you need to find the distance of all data points from the centroid of the cluster and add them.

Let's assume age on X-axis and income on Y-axis. Now, let's calculate the distance of the data points from the cluster centroid and complete the data points table.

A sample distance calculation is as following for the first data point

Distance from the Cluster Centroid (33, 15) for (33, 12) =

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

$$d = \sqrt{(33 - 33)^2 + (15 - 12)^2}$$

$$d = 3$$

Age	Income	Distance from Cluster Centroid
33	12	3.00
33	15	0.00
35	13	2.83
34	14	1.41
32	16	1.41
Total		**8.66**

Hence, the cluster inertia is 8.66.

Q. 10 Two clusters have the following data points. Calculate their intracluster distance. Also, calculate total inertia of the data point assignment. **(6 Marks)**

Height	Weight	Assigned Cluster
185	72	2
170	56	1
168	60	1
179	68	2
182	72	2
188	77	2
180	71	2
180	70	2
183	84	2
180	88	2
180	67	2
177	76	2

Ans. :

To calculate inertia (or intracluster distance) of a cluster, you need to find the distance of all data points from the centroid of the cluster and add them. In the question, centroids for the respective clusters are not given. Hence, first you need to calculate the centroids for the respective clusters.

For Centroid 1 calculation, you have two data points that fall in cluster 1. They are (170, 56) and (168, 60).

Hence, Centroid 1 is the mean of the data points which is $\left(\frac{170 + 168}{2}, \frac{56 + 60}{2}\right) = (169, 58)$.

For Centroid 2 calculation, take the remaining 10 data points that fall in cluster 2.

Hence, Centroid 2 is the mean of these 10 data points which is

$$\left(\frac{185+179+182+188+180+180+183+180+180+177}{10}, \frac{72+68+72+77+71+70+84+88+67+76}{10}\right)$$

$$= (181.4, 74.5)$$

Now, you have both the centroids ready for calculating the respective cluster inertia.

Centroid 1 = (169, 58) and Centroid 2 = (181.4, 74.5).

Now, let's calculate the respective distance of the data points from the centroids and complete the data points table.

Height	Weight	Distance from Centroid 1
170	56	2.24
168	60	2.24
Total		**4.48**

Height	Weight	Distance from Centroid 2
185	72	4.38
179	68	6.93
182	72	2.57
188	77	7.06
180	71	3.77
180	70	4.71
183	84	9.63
180	88	13.57
180	67	7.63
177	76	4.65
Total		**64.91**

Hence, The inertia for cluster 1 = 4.48

The inertia for cluster 2 = 64.91

Total inertia is sum of inertia of all clusters

= 4.48 + 64.91 = 69.39

Q. 11 What are some of the drawbacks of K-means algorithm. **(6 Marks)**

Ans. :

A few things to consider for K-means clustering are as following.

1. **Object attributes**

The data points could have several attributes such as age, weight, height, income, etc. Once your clustering is complete, you need to ensure that the attributes would be available for new data points as and when added for future analysis. For example, if your existing clustering is based on customer ratings on a particular product, such rating may not be immediately available for a customer who has just completed the purchase. It might require a week or a month before the customer is comfortable rating the product judiciously.

2. **Units of measurement and scaling**

You need to be careful in choosing the units for your data point attributes. While it may not impact K-means clustering a lot, it could actually shift a few data points here and there. For example, suppose you have two data points age and income. Age is expressed in double digits. But, if you choose income to be in thousands and someone earning 10,000 is represented as 10, then both age and income are in double digits and equally contribute towards distance calculation. But, suppose the income was not in thousands, then the income data attribute would dominate the distance calculation as it has many more digits than age. This might shift the data points more with respect to income than age and hence skew (bias or distort) the cluster assignment. For example, look at the following distance calculation.

In the first calculation, Centroid 1 = (33, 15) and Centroid 2 = (34, 14)

In the second calculation, Centroid 1 = (33, 15000) and Centroid 2 = (34, 14000)

Age	Income in thousand	Distance from Centroid 1	Distance from Centroid 2	Assigned Cluster
33	12	3.00	2.24	2
33	15	0.00	1.41	1
35	13	2.83	1.41	2
34	14	1.41	0.00	2
32	16	1.41	2.83	1

Age	Income	Distance from Centroid 1	Distance from Centroid 2	Assigned Cluster
33	12000	3000.00	2000.00	2
33	15000	0.00	1000.00	1
35	13000	2000.00	1000.00	2
34	14000	1000.00	0.00	2
32	16000	1000.00	2000.00	1

While you might argue that the cluster assignment has not changed, but if you look closely the distance between the centroids is more noticeable due to income attribute. This could have impact on inertia and might lead you to believe that you need more clusters.

3. **Initial Centroid position**

K-means is sensitive to initial centroid position. Hence, you should run the K-means analysis several times to ensure that clusters created are optimum.

4. **Number of clusters**

You must be careful while deciding on the number of clusters required for optimum placement of data points. Cluster Inertia is a key parameter to consider for ensuring that you have the right number of clusters.

Q. 12 Write a short note on k-Nearest Neighbours (kNN) classification algorithm. **(4 Marks)**

Ans. :

- One of the popular variations of K-means clustering algorithm is k-Nearest Neighbours (kNN) classification algorithm.
- It works on the same principle as K-means clustering algorithm that a data point is likely to resemble its neighbours and would possibly have the same classification. kNN is a supervised learning method.

- The way it works is simple and straightforward.
  1. Assume that you have already assigned labels to the existing data points in a given data set.
  2. Then you are given a new data point to classify.
  3. You calculate the distance of the new data point with respect to its k neighbours. The number k is chosen based on your data set and requirements but in general higher the better to reduce the noise and avoid incorrect labelling.
  4. The new data point is classified based on the classification of the majority of the surrounding neighbour's classification.

Q. 13 Following is a dataset for weight of teens and whether they like Pizza.

Weight (Kgs)	Like Pizza?
78	Yes
54	No
69	Yes
73	Yes
59	No
48	No
82	No
65	Yes

Using k-Nearest Neighbours (kNN) Classification Algorithm determine if a teen weighing 63 Kgs likely to like Pizza?

Use k = 3. **(6 Marks)**

Ans. : Calculate the distance of the new data point (63 Kgs) with respect to other data points in the data set.

A sample distance calculation is as following for the first data point.

$$d = \sqrt{(x_1 - x_2)^2}$$

$$d = \sqrt{(78 - 63)^2} = 15$$

$$d = 15$$

Weight (Kgs)	Distance for 63 Kgs	Like Pizza?
78	15	Yes
54	9	No
69	6	Yes
73	10	Yes
59	4	No
48	15	No
82	19	No
65	2	Yes

Sort the table based on the distance.

Weight (Kgs)	Distance for 63 Kgs	Like Pizza?
65	2	Yes
59	4	No
69	6	Yes
54	9	No
73	10	Yes
78	15	Yes
48	15	No
82	19	No

Now, given that k = 3.

So pick top 3 rows of the sorted table.

Weight (Kgs)	Distance for 63 Kgs	Like Pizza?
65	2	Yes
59	4	No
69	6	Yes

Q. 14 Explain k-medoids. **(6 Marks)**

Ans. :

- The medoid of a cluster is defined as the object (or the data point) in the cluster whose average dissimilarity to all the other objects in the cluster is minimal, that is, it is the most centrally located point in the cluster. The k-medoids algorithm is another approach for clustering similar to the k-means algorithm.
- k-medoids clustering is a partitioning method commonly used in domains that require robustness to outlier data, arbitrary distance metrics, or ones for which the mean or median does not have a clear definition. It is similar to k-means, and the goal of both methods is to divide a set of measurements or observations into k subsets or clusters so that the subsets minimise the sum of distances between a measurement and a center of the measurement's cluster. In the k-means algorithm, the center of the subset is the mean of measurements in the subset, often called a centroid. In the k-medoids algorithm, the center of the subset is a member of the subset, called a medoid.

- The k-medoids algorithm returns medoids which are the actual data points in the data set. This allows you to use the algorithm in situations where the mean of the data does not exist within the data set. This is the main difference between k-medoids and k-means where the centroids returned by k-means may not be within the data set. Hence k-medoids is useful for clustering categorical data where a mean is impossible to define or interpret.
- Both the k-means and k-medoids algorithms are partitional (breaking the dataset up into groups) and attempt to minimise the distance between points labelled to be in a cluster and a point designated as the center of that cluster. In contrast to the k-means algorithm, k-medoids chooses actual input data points as centres (called medoids or exemplars), and thereby allows for greater interpretability of the cluster centres than in k-means, where the center of a cluster is not necessarily one of the input data points (it is the average between the points in the cluster).

Q. 15 Compare k-means and k-medoids. **(4 Marks)**

Ans. :

Comparison Attribute	k-means	k-medoids
Center point	Average between the cluster points	Actual input data points
Works on	Minimising Euclidean distance	Minimising dissimilarity
Cluster centres	Less prominent	More prominent
Robustness to outliers and noise	Low	High
Need mean or median	Yes	No

Q. 16 Write a short note on dendrogram. **(4 Marks)**

Ans. :

- **Definition** : A dendrogram is a diagram representing a tree or hierarchy.
- It is a branching diagram representing a hierarchy of categories based on degree of similarity or number of shared characteristics. For example, the Fig. 5.7 illustrates a simple dendrogram for six observations.

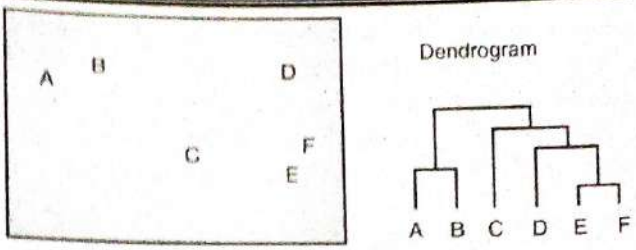


Fig. 5.7

- The key to interpreting a dendrogram is to focus on the height at which any two objects are joined together. In the given example, you can see that E and F are most similar, as the height of the link that joins them together is the smallest. The next two most similar objects are A and B.
- The height of the dendrogram indicates the order in which the clusters were joined. A more informative dendrogram can be created where the heights reflect the distance between the clusters as shown in the Fig. 5.7(a).

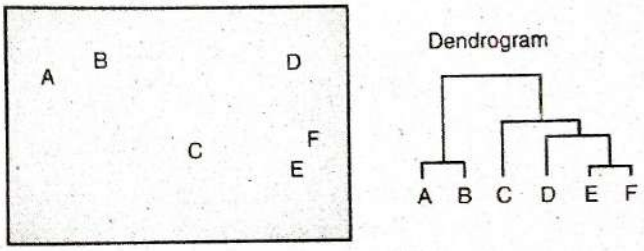


Fig. 5.7(a)

- In this case, the dendrogram shows that the big difference between clusters is between the cluster of A and B versus that of C, D, E, and F.
- It is important to appreciate that the dendrogram is a summary of the distance matrix, and, as occurs with most summaries, information is lost.
- For example, the dendrogram suggests that C and D are much closer to each other than is C to B, but the original data (shown in the scatter plot), shows that this is not true. The consequence of the information loss is that the dendrograms are most accurate at the bottom, showing which items are very similar.

Q. 17 Explain Agglomerative Hierarchical Clustering. **(6 Marks)**

Ans. : Agglomerative Hierarchical Clustering, commonly referred to as AGNES (AGglomerative NESting), works in a bottom-up manner. That is, each observation is initially considered as a single-element cluster (leaf). At each step of the algorithm, the two clusters that are the most

similar are combined into a new bigger cluster (nodes). This procedure is iterated until all points are a member of just one single big cluster (root). The result is a tree which can be displayed using a dendrogram. Consider that the data points are plotted as shown in Fig. 5.8.

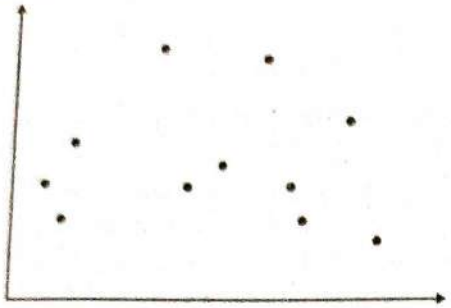


Fig. 5.8

In first iteration, individual point clusters are made as Fig. 5.8(a).

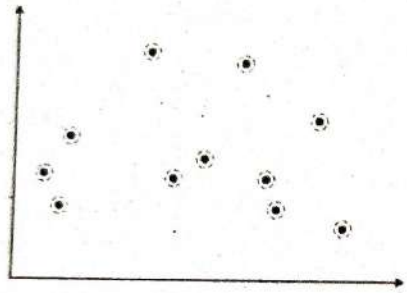


Fig. 5.8(a)

In the next few iterations, clusters closer to each other start merging.

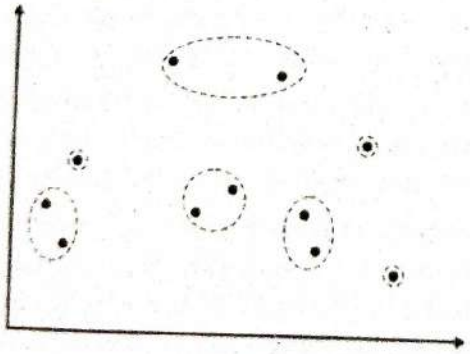


Fig. 5.8(b)

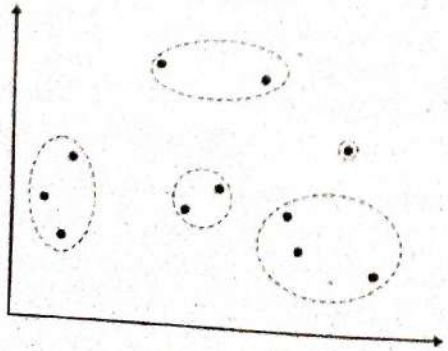


Fig. 5.8(c)

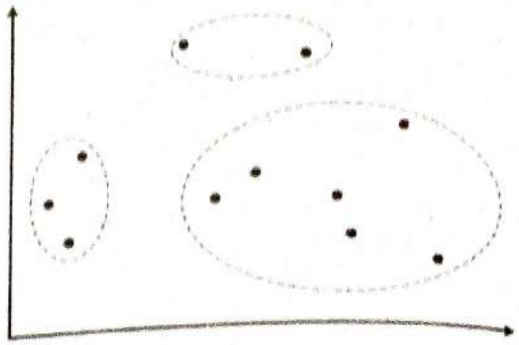


Fig. 5.8(d)

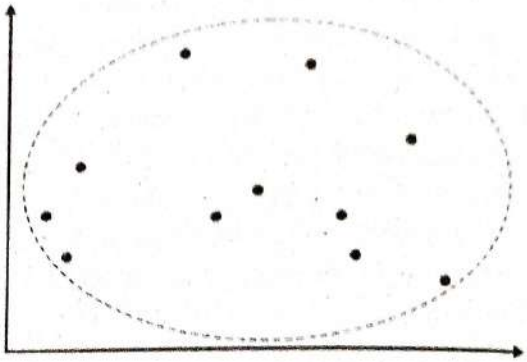


Fig. 5.8(e)

Q. 18 Explain Divisive Hierarchical Clustering. **(6 Marks)**

Ans. :

Divisive Hierarchical Clustering, commonly referred to as DIANA (DIvise ANAlysis), works in a top-down manner. DIANA is like the reverse of AGNES. It begins with the root, in which all observations are included in a single cluster. At each step of the algorithm, the current cluster is split into two clusters that are considered most heterogeneous. The process is iterated until all observations are in their own cluster. The result is a tree which can be displayed using a dendrogram. Consider that the data points are plotted as shown in Fig. 5.9

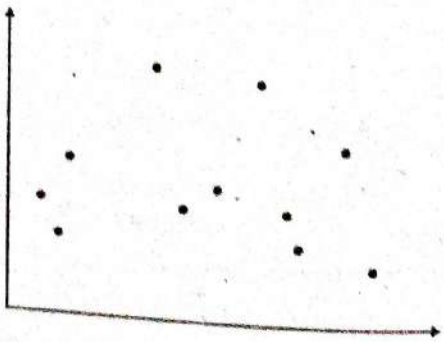


Fig. 5.9

In first iteration, all points are considered as part of one big cluster.

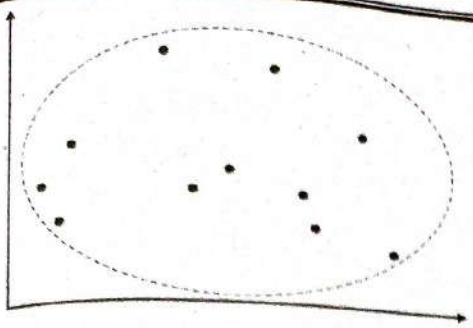


Fig. 5.9(a)

In the next few iterations, clusters are further broken up into smaller clusters until all points are in their own cluster.

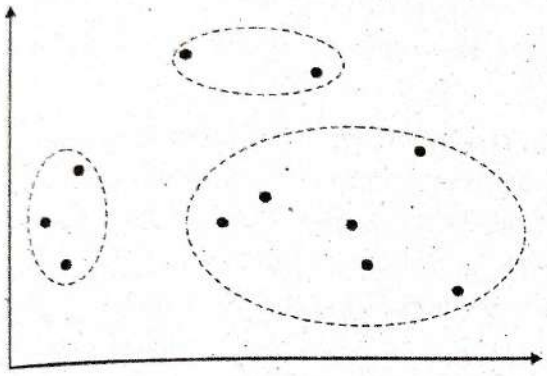


Fig. 5.9(b)

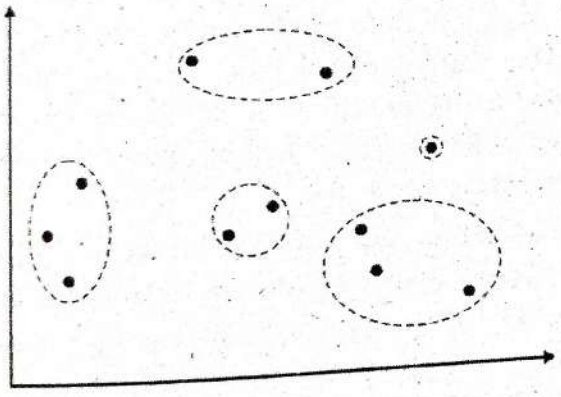


Fig. 5.9(c)

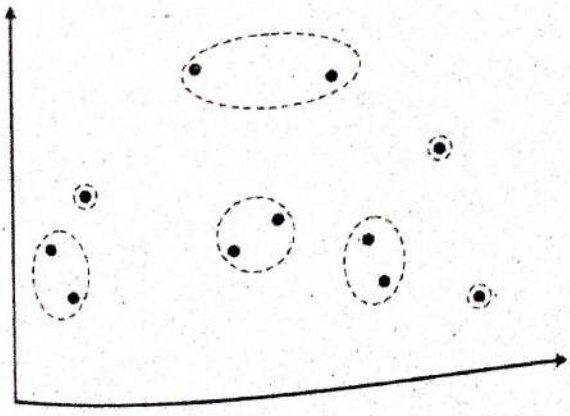


Fig. 5.9(d)

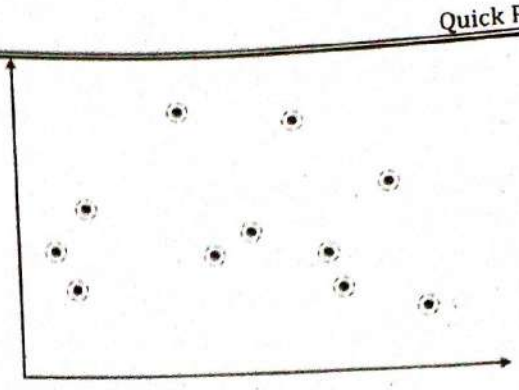


Fig. 5.9(e)

Q. 19 List a few Agglomeration (Linkage) Methods. **(4 Marks)**

Ans. :

The most common methods are as following.

1. **Maximum or complete linkage clustering**

   This method computes all pairwise dissimilarities between the elements in cluster 1 and the elements in cluster 2 and considers the largest value of these dissimilarities as the distance between the two clusters. It tends to produce more compact clusters.

2. **Minimum or single linkage clustering**

   This method computes all pairwise dissimilarities between the elements in cluster 1 and the elements in cluster 2 and considers the smallest of these dissimilarities as a linkage criterion. It tends to produce long, "loose" clusters.

3. **Mean or average linkage clustering**

   This method computes all pairwise dissimilarities between the elements in cluster 1 and the elements in cluster 2 and considers the average of these dissimilarities as the distance between the two clusters. The compactness of the clusters it creates can vary.

4. **Centroid linkage clustering**

   This method computes the dissimilarity between the centroid for cluster 1 and the centroid for cluster 2.

5. **Ward's minimum variance method**

   This method minimises the total within cluster variance. At each step, the pair of clusters with the smallest between-cluster distance are merged. It tends to produce more compact clusters.

Q. 20 Explain the terms epsilon, neighbourhood, density, min_sample with respect to DBSCAN. **(4 Marks)**

Ans. :

1. **Epsilon (eps) denoted by ε :** It is a measure of neighbourhood. Simply speaking, it helps you to identify how large the neighbourhood is. For example, if you draw a circle with a centre point C, then ε just denotes the radius of the circle. So, for the point C, circle is the neighbourhood, and ε is the radius of the circle.

The Fig. 5.10 illustrates the concept of eps.

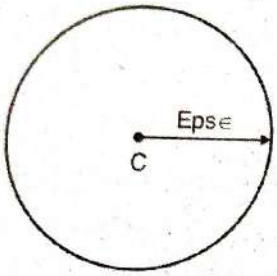


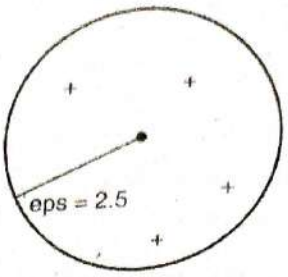
Fig. 5.10

Eps specifies how close points should be to each other to be considered a part of a cluster (same neighbourhood). It means that if the distance between two points is lower or equal to this value (eps), these points are considered to be neighbours.

Eps is crucial to choose appropriately for the data set and distance function and usually cannot be left at the default value. It controls the local neighbourhood of the points. When chosen too small, most data will not be clustered at all. If Eps is chosen to be too large, then it causes close clusters to be merged into one cluster, and eventually the entire data set to be returned as a single cluster.

2. **min_sample (also called as minPts) :** min_sample (or minPts) is the measure of minimum number of points you want to consider for identifying a neighbourhood. For example, if you require at least three points and you have 5 points around an area, then that area could be called as a neighbourhood. Another way to look at it is that min_sample tells you how many points you minimally require to consider an area dense. Min_sample primarily controls how tolerant the algorithm is towards noise (on noisy and large data sets it may be desirable to increase this parameter).

The Fig. 5.10(a) illustrates the concept of min_sample.



Let the minPoints

So, Density = 5

i.e. density > minPoints

Hence, it's a "Dense Region"

Fig. 5.10(a)

Q. 21 Draw a diagram to illustrate and explain core, border, and noise points with respect to DBSCAN. **(4 Marks)**

Ans. :

4. **Core point :** A point is considered to be a core point if at least a specified number (min_sample) of neighbouring points fall within the specified radius ε.

5. **Border point :** This is a point that has at least one core point at a particular distance but is not surrounded by enough min_sample points. In other words, if the point under observation does not have sufficient neighbours (data points) of its own, but it has at least one core point in its neighbourhood, then that point represents the border point of the cluster. Border points belong to the same cluster of their nearest core point.

6. **Noise points :** All other points that are neither core nor border points are considered as noise points.

The Fig. 5.11 illustrates core, border, and noise points.

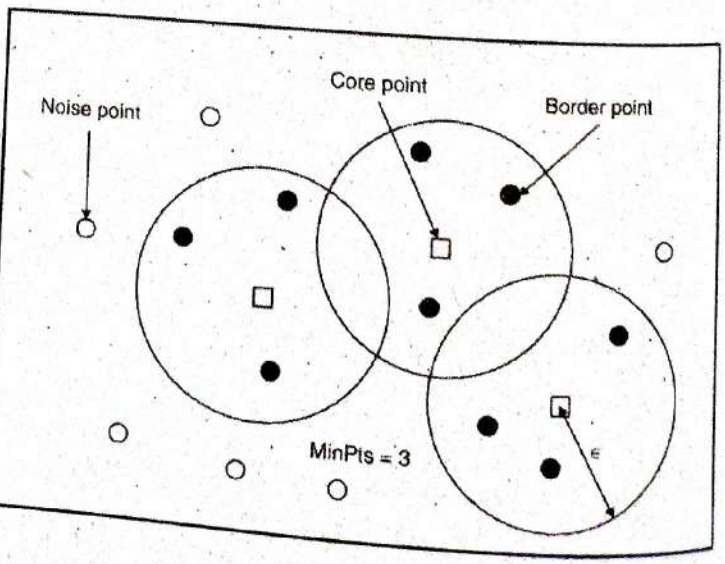


Fig. 5.11

Q. 22 What are the steps of DBSCAN algorithm. (4 Marks)

Ans. :

The steps in DBSCAN algorithm.

1. For each point in the dataset, find out the distance to every other point in the dataset.
2. All points that fall within the neighbourhood (within a radius of ε) are considered as neighbours.
3. If the minimum point threshold (min_sample) is reached, then the points are grouped together as a cluster or else marked as noise.
4. This process is repeated until all data points are categorised into clusters or as noise.

Q. 23 List advantages and disadvantages of DBSCAN. (4 Marks)

Ans. :

The advantages of DBSCAN are as following.

1. One of the main advantages of using DBSCAN is that it does not assume that the clusters have a spherical shape as in k-means algorithm.
2. It does not necessarily assign each point to a cluster but is capable of removing noise points.
3. You do not need to know the number of desired clusters (k) in advance.
4. It is highly efficient as once a point has been assigned to a cluster in DBSCAN, it does not change cluster membership.

DBSCAN has following challenges.

1. If your dataset has multiple densities or varying densities, DBSCAN tends to fail.
2. It is extremely sensitive to the value of eps and min_sample.
3. It may not work well with very high dimensionality data.

Q. 24 Write a short note on Outlier Analysis. (4 Marks)

Ans. :

- **Definition :** An outlier is a statistical observation that is markedly different in value from the others of the sample.
- Outlier detection (also known as anomaly detection) is the process of finding data objects with behaviours that are very different from expectation. Such objects are called outliers or anomalies. Outlier detection is important in many applications in addition to fraud detection such as medical care, public safety and security, industry damage detection, image processing, sensor/video network surveillance, and intrusion detection. Outlier detection and clustering analysis are two highly related tasks. Clustering finds the majority patterns in a data set and organizes the data accordingly, whereas outlier detection tries to capture those exceptional cases that deviate substantially from the majority patterns. Outlier detection and clustering analysis serve different purposes.
- Assume that a given statistical process is used to generate a set of data objects. An outlier is a data object that deviates significantly from the rest of the objects, as if it were generated by a different mechanism. For simplicity, you may refer to data objects that are not outliers as "normal" or expected data. Similarly, you may refer to outliers as "abnormal" data.
- For example, in the Fig. 5.12, the objects in the region R, are outliers.

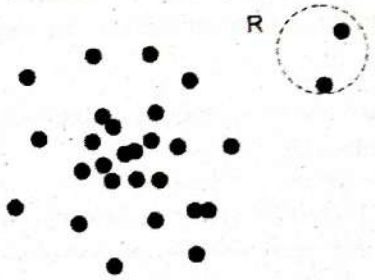


Fig. 5.12

- One thing to note here is that outliers are different from noisy data. Noise is a random error or variance in a measured variable. In general, noise is not interesting in data analysis, including outlier detection. For example, in credit card fraud detection, a customer's purchase behaviour can be modelled as a random variable. A customer may generate some "noise transactions" that "random errors" or "variance," such as by buying a bigger lunch one day, or having one more cup of coffee than usual. Such transactions should not be treated as outliers; otherwise, the credit card company would incur heavy costs from verifying that many transactions. The company may also lose customers by bothering them with multiple false alarms. As in many other data analysis and data mining tasks, noise should be removed before outlier detection.

- Outliers are interesting because they are suspected of not being generated by the same mechanisms as the rest of the data. Therefore, in outlier detection, it is important to justify why the outliers detected are generated by some other mechanisms. This is often achieved by making various assumptions on the rest of the data and showing that the outliers detected violate those assumptions significantly.
- Outlier detection is also related to novelty detection in evolving data sets. For example, by monitoring a social media web site where new content is incoming, novelty detection may identify new topics and trends in a timely manner. Novel topics may initially appear as outliers. To this extent, outlier detection and novelty detection share some similarity in modelling and detection methods. However, a critical difference between the two is that in novelty detection, once new topics are confirmed, they are usually incorporated into the model of normal behaviour so that follow-up instances are not treated as outliers anymore.

Q. 25 Explain a few methods of outlier detection. **(6 Marks)**

Ans. :

- There are several methods for outlier detection such as the following.

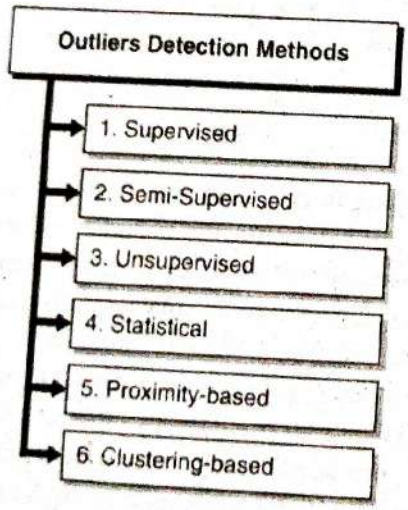


Fig. 5.13

1. Supervised Method for Outlier Detection

Supervised methods model data normality and abnormality. Domain experts examine and label a sample of the underlying data. Outlier detection can then be modelled as a classification problem. The task is to learn a classifier that can recognise outliers. The sample is used for training and testing. In some applications, the experts may label just the normal objects, and any other objects not matching the model of normal objects are reported as outliers. Other methods model the outliers and treat objects not matching the model of outliers as normal.

2. Semi-Supervised Method for Outlier Detection

- In many applications, although obtaining some labelled examples is feasible, the number of such labelled examples is often small. You may encounter cases where only a small set of the normal and/or outlier objects are labelled, but most of the data are unlabelled. Semi-supervised outlier detection methods were developed to tackle such scenarios.
- Semi-supervised outlier detection methods can be regarded as applications of semi-supervised learning methods. For example, when some labelled normal objects are available, you can use them, together with unlabelled objects that are close by, to train a model for normal objects. The model of normal objects then can be used to detect outliers those objects not fitting the model of normal objects are classified as outliers.
- If only some labelled outliers are available, semi-supervised outlier detection is trickier. A small number of labelled outliers are unlikely to represent all the possible outliers. Therefore, building a model for outliers based on only a few labelled outliers is unlikely to be effective. To improve the quality of outlier detection, you can get help from models for normal objects learned from unsupervised methods.

3. Unsupervised Method for Outlier Detection

- In some application scenarios, objects labelled as "normal" or "outlier" are not available. Thus, an unsupervised learning method has to be used.
- Unsupervised outlier detection methods make an implicit assumption: The normal objects are somewhat "clustered." In other words, an unsupervised outlier detection method expects that normal objects follow a pattern far more frequently than outliers. Normal objects do not have to fall into one group sharing high similarity. Instead, they can form multiple groups, where each group has distinct features. However, an outlier is expected to occur far away in feature space from any of those groups of normal objects.
- Many clustering methods can be adapted to act as unsupervised outlier detection methods.

4. Statistical Method for Outlier Detection

- Statistical methods (also known as model-based methods) make assumptions of data normality. They assume that normal data objects are generated by a statistical model, and that data not following the model are outliers. The effectiveness of statistical methods highly depends on whether the assumptions made for the statistical model hold true for the given data. There are many kinds of statistical models. For example, the statistic models used in the methods may be parametric or non-parametric.

5. Proximity-based Method for Outlier Detection

- Proximity-based methods assume that an object is an outlier if the nearest neighbours of the object are far away in feature space, that is, the proximity of the object to its neighbours significantly deviates from the proximity of most of the other objects to their neighbours in the same data set.
- The effectiveness of proximity-based methods relies heavily on the proximity (or distance) measure used. In some applications, such measures cannot be easily obtained. Moreover, proximity-based methods often have difficulty in detecting a group of outliers if the outliers are close to one another.

6. Clustering-based Method for Outlier Detection

- Clustering-based methods assume that the normal data objects belong to large and dense clusters, whereas outliers belong to small or sparse clusters, or do not belong to any clusters. Clustering is an expensive data mining operation. A straightforward adaptation of a clustering method for outlier detection can be very costly, and thus does not scale up well for large data sets.

Q. 26 Explain isolation forest. (6 Marks)

Ans. :

- Isolation Forest (iForest) is an unsupervised learning approach that detects anomalies by isolating instances without relying on any distance or density measure. It exploits the following two quantitative properties of anomalies.
  1. They are the minority consisting of few instances.
  2. They have attribute-values that are very different from those of normal instances.
- In other words, anomalies are "few and different" which make them more susceptible to a mechanism called isolation. Isolation can be implemented by any means that separates instances.
- iForest uses a binary tree structure called isolation tree (iTree) which can be constructed effectively to isolate instances. Because of the susceptibility to isolation, anomalies are more likely to be isolated closer to the root of an iTree whereas normal points are more likely to be isolated at the deeper end of an iTree. This forms the basis of iForest method to detect anomalies.

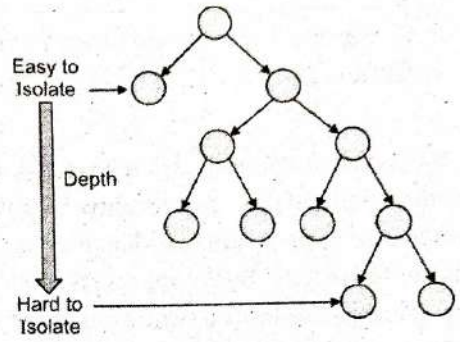


Fig. 5.14

- iForest builds an ensemble of iTrees for a given data set. Anomalies are instances which have short average path lengths on the iTrees. There are two training parameters and one evaluation parameter in this method. The training parameters are the number of trees to build, and subsampling size and the evaluation parameter is the tree height limit during evaluation.
- In randomly generated binary trees where instances are recursively partitioned, these trees produce noticeable shorter paths for anomalies. Hence, when a forest of random trees collectively produce shorter path lengths for some particular points, they are highly likely to be anomalies.

Q. 27 Explain isolation forest. (6 Marks)

Ans. : iForest has the following characteristics.

1. Isolation trees enable iForest to exploit subsampling to a great extent. Since iForest does not need to isolate all of normal instances, it can frequently ignore the big majority of the training sample. As a consequence, iForest works very well when the sampling size is kept small, a property that is in contrast with the great majority of existing methods, where large sampling size is usually desirable.

2. iForest utilises no distance or density measures to detect anomalies. This eliminates a major computational cost of distance calculation in all distance-based and density-based methods.
3. iForest has a linear time complexity with a small constant and a minimal memory requirement. It is an algorithm with constant training time and space complexities.
4. iForest has the capacity to scale up to handle extremely large data size and high-dimensional problems with a large number of irrelevant attributes.

Q. 28 With an example, explain Local Outlier Factor method for outlier analysis. **(6 Marks)**

Ans. :

- The Local Outlier Factor (LOF) is based on a concept of a local density, where locality is given by k nearest neighbours, whose distance is used to estimate the density. By comparing the local density of an object to the local densities of its neighbours, you can identify regions of similar density, and points that have a substantially lower density than their neighbours. These are considered to be outliers.
- LOF algorithm is an unsupervised anomaly detection method which computes the local density deviation of a given data point with respect to its neighbours. Outliers are the samples that have a substantially lower density than their neighbours. The following example shows how to use LOF for outlier detection.

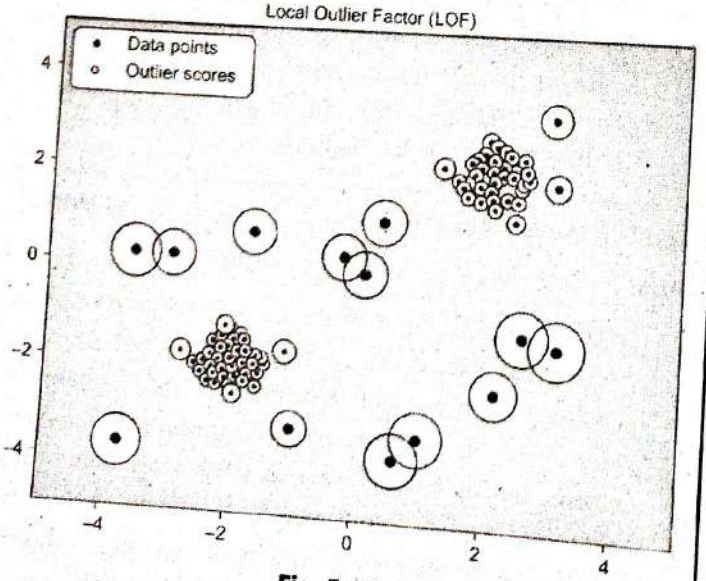


Fig. 5.15

- Suppose that there is a distance, k – distance (A), from the object A to the k^{th} nearest neighbour. The set of nearest neighbours includes all objects at this distance. You denote the set of k nearest neighbours as $N_k(A)$.
- For example, in the Fig. 5.15(a), Point A is of interest.

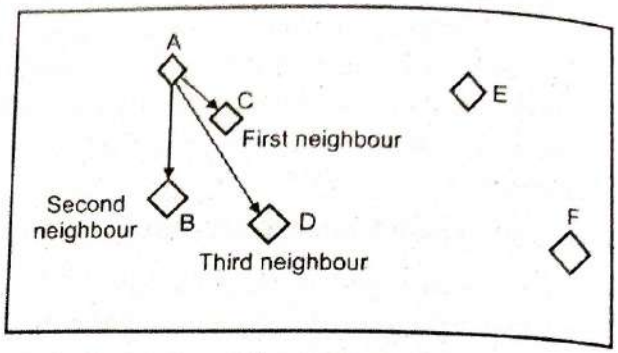


Fig. 5.15(a)

  - C is the closest point to A
  - B is the second closest point to A
  - D is the third closest point to A
- If k = 3, then the distance from A to D is the k-distance.
- This k-distance is used to determine the reachability distance as following. The reachability distance from the point of interest to any point is the maximum number amongst the k-distance and the actual distance between the data points.

$$\text{reachability} - \text{distance}_k(A, B) = \max(\text{k} - \text{distance}(B)), \text{dist}(A, B)$$

Q. 29 Explain extrinsic method of measuring clustering quality. **(6 Marks)**

Ans. :

- When the ground truth is available, you can compare it with a clustering to assess the clustering. Thus, the core task in extrinsic methods is to assign a score, $Q(C, C_g)$, to a clustering C given the ground truth C_g.
- Whether an extrinsic method is effective largely depends on the measure, Q, it uses. In general, a measure Q on clustering quality is effective if it satisfies the following four essential criteria.

1. Cluster homogeneity : This requires that the purer the clusters in a clustering are, the better the clustering. Suppose that ground truth says that the objects in a data set, D, can belong to categories

L1, ..., Ln. Consider clustering, C1, wherein a cluster $C \in C1$ contains objects from two categories Li, Lj $(1 \le i < j \le n)$. Also consider clustering C2, which is identical to C1 except that C2 is split into two clusters containing the objects in Li and Lj, respectively. A clustering quality measure, Q, respecting cluster homogeneity should give a higher score to C2 than C1, that is, Q(C2, Cg) > Q(C2, Cg) as C2 is better clustered (having homogenous items instead of mix) than C1.

2. Cluster completeness : This is the complement of cluster homogeneity. Cluster completeness requires that for a clustering, if any two objects belong to the same category according to ground truth, then they should be assigned to the same cluster. Cluster completeness requires that a clustering should assign objects belonging to the same category (according to ground truth) to the same cluster.

Consider clustering C1, which contains clusters A1 and A2, of which the members belong to the same category according to ground truth. Let clustering C2 be identical to C1 except that A1 and A2 are merged into one cluster in C2. Then, a clustering quality measure, Q, respecting cluster completeness should give a higher score to C2, that is, Q(C2, Cg) > Q(C1, Cg) as C2 has one complete cluster (instead of two) containing both A1 and A2.

3. Rag bag : In many practical scenarios, there is often a "rag bag" category containing objects that cannot be merged with other objects. Such a category is often called "miscellaneous," "other," and so on. The rag bag criterion states that putting a heterogeneous object into a pure cluster should be penalised more than putting it into a rag bag.

Consider a clustering C1 and a cluster $C \in C1$ such that all objects in C except for one, denoted by o, belong to the same category according to ground truth. Consider a clustering C2 identical to C1 except that o is assigned to a cluster $C' \neq C$ in C2 such that C' contains objects from various categories according to ground truth, and thus is noisy. In other words, C' in C2 is a rag bag. Then, a clustering quality measure, Q, respecting the rag bag criterion should give a higher score to C2, that is, Q(C2, Cg) > Q(C1, Cg) as it correctly puts the object o in a separate noise category instead of mixing it with a "good" cluster.

4. Small cluster preservation : If a small category is split into small pieces in a clustering, those small pieces may likely become noise and thus the small category cannot be discovered from the clustering. The small cluster preservation criterion states that splitting a small category into pieces is more harmful than splitting a large category into pieces.

Consider an extreme case. Let D be a data set of n + 2 objects such that, according to ground truth, n objects, denoted by $o_1, ..., o_n$, belong to one category and the other two objects, denoted by o_{n+1}, o_{n+2} belong to another category.

Suppose clustering C_1 has three clusters,

$$A_1 = \{o_1, ..., o_n\}$$
$$A_2 = \{o_{n+1}\}$$
$$A_3 = \{o_{n+2}\}$$

- Let clustering C_2 have three clusters, too, namely

$$A_1 = \{o_1, ..., o_{n-1}\}$$
$$A_2 = \{o_n\}$$
$$A_3 = \{o_{n+1}, o_{n+2}\}$$

- In other words, C_1 splits the small category and C_1 splits the big category. A clustering quality measure, Q, preserving small clusters should give a higher score to C_2, that is, $Q(C_2, C_g) > Q(C_1, C_g)$.

Q. 30 Explain silhouette coefficient. **(6 Marks)**

Ans. :

- The silhouette coefficient is an intrinsic measure of cluster quality. For a data set, D, of n objects, suppose D is partitioned into k clusters, $C_1, ..., C_k$. For each object $o \in D$, you calculate a(o) as the average distance between o and all other objects in the cluster to which o belongs. Similarly, b(o) is the minimum average distance from o to all clusters to which o does not belong. Formally, suppose $o \in C_i$ $(1 \le i \le k)$; then

$$a(o) = \frac{\Sigma o' \in C_i, o \neq o' \, dist(o, o')}{|C_i| - 1}$$

$$b(o) = \min_{C_i : 1 \le j \le k, i = j} \frac{\Sigma o' \in C_j \, dist(o, o')}{|C_j|}$$

- The silhouette coefficient of o is then defined as

$$s(o) = \frac{b(o) - a(o)}{\max\{a(o), b(o)\}}$$

- The value of the silhouette coefficient is between −1 and 1.
  - 1 means clusters are well apart from each other and clearly distinguished.
  - 0 means clusters are indifferent, or you can say that the distance between clusters is not significant
  - −1 means clusters are assigned incorrectly
- The value of a(o) reflects the compactness of the cluster to which o belongs. The smaller the value, the more compact the cluster. The value of b(o) captures the degree to which o is separated from other clusters. The larger b(o) is, the more separated o is from other clusters. Therefore, when the silhouette coefficient value of o approaches 1, the cluster containing o is compact and o is far away from other clusters, which is the preferable case. However, when the silhouette coefficient value is negative, i.e., $b(o) < a(o)$, then this means that, in expectation, o is closer to the objects in another cluster than to the objects in the same cluster as o. In many cases, this is a bad situation and should be avoided.
- To measure a cluster's fitness within a clustering, you can compute the average silhouette coefficient value of all objects in the cluster. To measure the quality of a clustering, you can use the average silhouette coefficient value of all objects in the data set. The silhouette coefficient and other intrinsic measures can also be used in the elbow method to heuristically derive the number of clusters in a data set by replacing the sum of within-cluster variances.

❑❑❑

Chapter 6 : Introduction to Neural Networks

Q. 1 Write a short note on synapses. **(4 Marks)**

Ans. :

- Neurons send signals using action potentials. An action potential is a shift in the neuron's electric potential caused by the flow of ions in and out of the neural membrane. Signals pass from an axon to a dendrite through a junction called synapses. Synapses can connect an axon to another axon or a dendrite to another dendrite. Action potentials can trigger both chemical and electrical synapses.
- A neuron could typically have around 10,000 synapses. So, given that there are 100 billion neurons, and each neuron could have 10,000 synapses, there are typically 1,000 trillion (1 quadrillion) synapses within the brain!
- One thing to note here is that there is a small gap between the end of an axon and the adjacent dendrite of the neighbouring neuron. This gap is called a synapse.
- A nervous stimulus is an electric impulse. It is transmitted across a synaptic gap by means of electrochemical process. The synaptic gap has an important role to play in the activities of the nervous system. It scales (multiplies) the input signal by a weight.
- If the input signal is S, and the synaptic weight is W, then the stimulus that finally reaches the soma due to input S is the product $S \times W$. The significance of the weight W provided by the synaptic gap lies in the fact that this weight, together with other synaptic weights, embody the knowledge stored in the network of neurons.
- This is in contrast with digital computers where the knowledge is stored as a program or block of data in the memory. The weight factors provided by the synaptic gaps are modified over time and experience. This phenomenon, perhaps, accounts for development of skills through practice, or loss of memory due to infrequent recall of stored information.

Q. 2 What is Hebb's rule? **(4 Marks)**

Ans. :

- Neutral networks evolve based on repeated experiences over time. This is often cited as Hebb's rule. Hebb's rule says that when the axon of a cell *A* is close enough to excite a cell *B* and takes part on its activation in a repetitive and persistent way, some type of growth process or metabolic change takes place in one or both cells. Such frequent joint activation increases the efficiency of cell *A* in the activation of *B*.

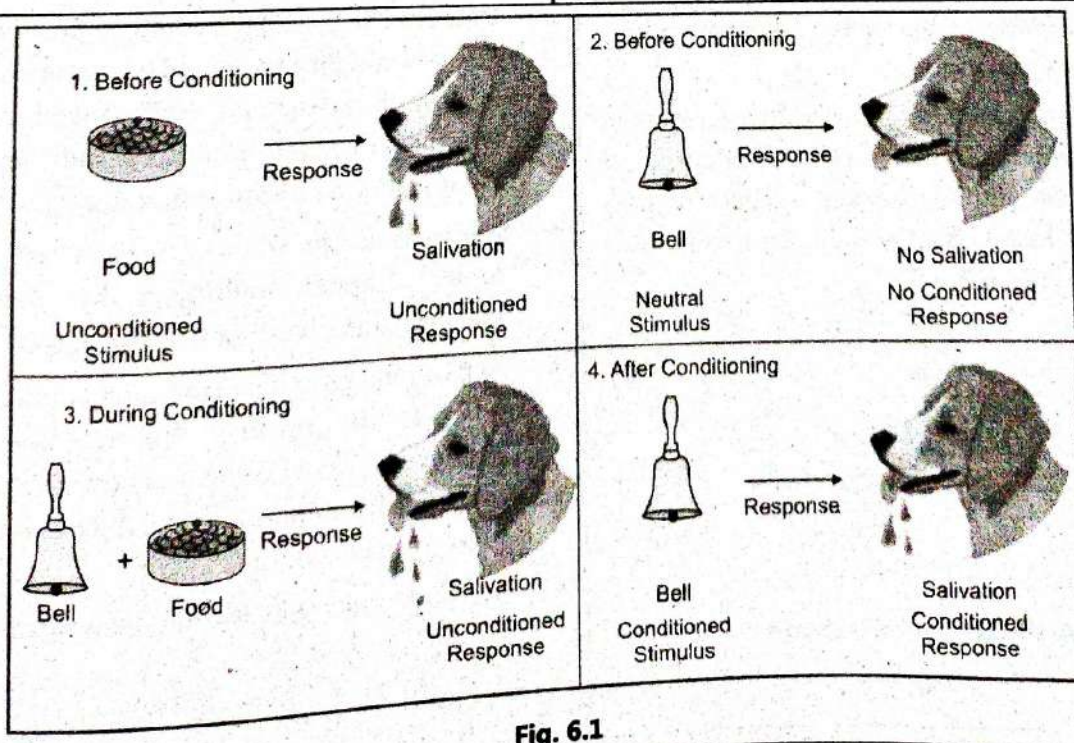


Fig. 6.1

- Neurons that fire together wire together". The changes in the strength of synaptic connections are proportional to the correlation in the firing of the two connecting neurons. So, if two neurons consistently fire simultaneously, then any connection between them will change in strength, becoming stronger. However, if the two neurons never fire simultaneously, the connection between them will die away. The basic idea is that if two neurons both respond to something, then they should be connected in some way. It is important to note that the neurons must be previously connected, sufficiently close to one another, so that the synapse can be reinforced.
- There are other names for this behaviour of jointly firing neurons such as long-term potentiation and neural plasticity.

Q. 2 Write a short note on Artificial Neural Network. **(4 Marks)**

Ans. :

- Artificial Neural Networks (ANNs) are inspired by biological neural networks. ANNs try to mimic the computations structure (not biological structure) and working of biological neurons and try to establish learning and response models that would work on provided inputs and produce the appropriate output.
- **Definition :** Artificial Neural Network is a computing system made up of a number of simple, highly interconnected processing elements, which process information by their dynamic state response to external input.
- If these functions are diagrammatically represented (without involving biological structure or complexities) to outline the key aspects of how neurons work, then it could possibly look something like as shown in Fig. 6.2.

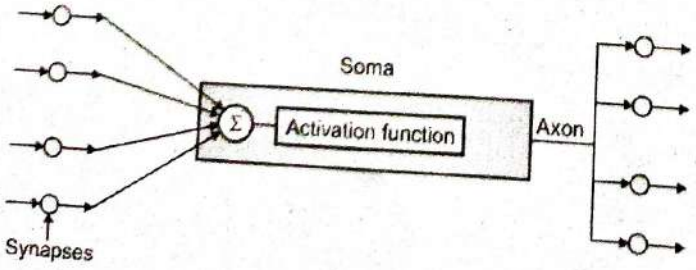


Fig. 6.2 : Artificial Neural Networks

- This forms the basis of building an ANN, taking an inspiration from biological neuron. An ANN is a group of interconnected artificial neurons, interacting with one another in a collaborative manner.
- In such a system, excitation (input signals) is applied to the input of the network and the input is multiplied by appropriate weights as the input signal passes through the network. Based on the accumulated weighted signals, the activation function decides if the artificial neuron would produce output (get activated) or not.

Q. 3 Write a short note on Artificial Neural Network. **(4 Marks)**

Ans. :

- The mathematical model extracts only the bare essentials required to accurately represent a neuron and removing all of the other unnecessary details. It is one of the several models used today for modelling an artificial neuron.
- The model is represented as shown in Fig. 6.3.

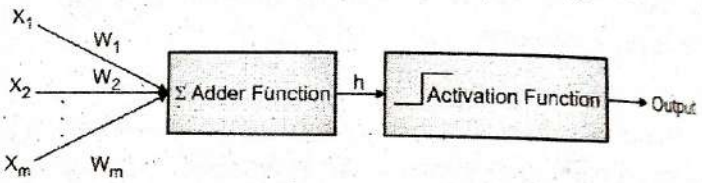


Fig. 6.3 : McCulloch-Pitts (MCP) Neuron Model

- $X_1 \ldots X_m$ are inputs.
- $W_1 \ldots W_m$ are respective weights for those inputs. Note here that weights could be positive or negative. Positive weights are called excitatory and they make it more likely that the neuron would fire. Negative weights are called inhibitory and they make it less likely that the neuron would fire.
- Adder function sums (accumulates) the weighted inputs (inputs multiplied by their respective weights) and gives out *h* as the output
- Activation function (threshold function) decides whether the artificial neuron fires ('spikes' or produces output) given the current sum of weighted inputs. When a neuron fires it gives output as 1, else it gives an output of 0.
- So, mathematically, it could be written as

$$h = \sum_{i=1}^{m} W_i X_i$$

Suppose the threshold above which the neuron fires is denoted by θ.

Then,

If $h > \theta$, then the neuron fires and output is 1.

If $h \leq \theta$, then the neuron does not fire, and output is 0.

Q. 4 The following inputs and weights are fed to McCulloch-Pitts Neuron Model. Assuming that threshold is 0.3, does the neuron fire? **(6 Marks)**

Inputs	Weights
1	1
0	− 0.5
0.5	− 1

Ans. : Given that inputs are

- $X_1 = 1$
- $X_2 = 0$
- $X_3 = 0.5$

Given that weights are

- $W_1 = 1$
- $W_2 = -0.5$
- $W_3 = -1$

Total aggregated and weighted input to the neuron is given as

$$h = \sum_{i=1}^{m} W_i X_i$$

$$h = W_1 X_1 + W_2 X_2 + W_3 X_3$$

$$h = 1 \times 1 + -0.5 \times 0 + -1 \times 0.5$$

$$h = 1 + 0 - 0.5$$

$$h = 0.5$$

Given that threshold θ is 0.3.

$$h\,(0.5) > \theta\,(0.3)$$

Hence, the neuron would fire and produce an output of 1.

Q. 5 Determine weights and threshold for the given data using McCulloch-Pitts neuron model. **(6 Marks)**

X_1	X_2	D
0	0	0
0	1	0
1	0	1
1	1	0

Ans. :

You need to determine weights such that the neuron produces the output as given for the corresponding inputs.

X_1	X_2	W_1	W_2	D
0	0			0
0	1			0
1	0			1
1	1			0

This is as good as solving linear equations.

From, McCulloch-Pitts Neuron Model, you know that the neuron fires when the sum of weighted inputs is greater than the threshold.

So, $W_1X_1 + W_2X_2 >$ Threshold to produce an output (which is 1)

Let's assume that $W_1 = 1$ and $W_2 = -1$

For inputs $X_1 = 0$ and $X_2 = 0$

$$h_1 = W_1X_1 + W_2X_2$$

$$= 1 \times 0 + -1 \times 0 = 0$$

For inputs $X_1 = 0$ and $X_2 = 1$

$$h_2 = W_1X_1 + W_2X_2$$

$$= 1 \times 0 + -1 \times 1 = -1$$

For inputs $X_1 = 1$ and $X_2 = 0$

$$h_3 = W_1X_1 + W_2X_2$$

$$= 1 \times 1 + -1 \times 0 = 1$$

For inputs $X_1 = 1$ and $X_2 = 1$

$$h_4 = W_1X_1 + W_2X_2$$

$$= 1 \times 1 + -1 \times 1 = 0$$

So, only h_3 should produce output based on the given table. Hence, you can choose threshold = 0.5 and weights as $W_1 = 1$ and $W_2 = -1$, h_1, h_2, and h_4 are all less than the threshold value of 0.5 and hence the neuron would not fire.

X_1	X_2	W_1	W_2	Weighted Sum	D(Threshold = 0.5). Output = 1, if weighted sum > threshold?
0	0	1	−1	0	0 (0 < 0.5)
0	1	1	−1	−1	0 (− 1< 0.5)
1	0	1	−1	1	1 (1 > 0.5)
1	1	1	−1	0	0 (0 < 0.5)

Q. 6 Implement logical AND using McCulloch-Pitts neuron model. Also, draw the neuron.

Ans. : As you know, the truth table for logical AND is as following.

X_1	X_2	Y
0	0	0
0	1	0
1	0	0
1	1	1

Assume $W_1 = 0.5$ and $W_2 = 0.5$ and threshold = 0.9.

X_1	X_2	W_1	W_2	Weighted Sum	Y(Threshold = 0.9) Output = 1, if weighted sum > threshold?
0	0	0.5	0.5	0	0 (0 < 0.9)
0	1	0.5	0.5	0.5	0 (0.5< 0.9)
1	0	0.5	0.5	0.5	0 (0.5 < 0.9)
1	1	0.5	0.5	1	1 (1 > 0.9)

The MCP neuron to implement AND could be implemented as following.

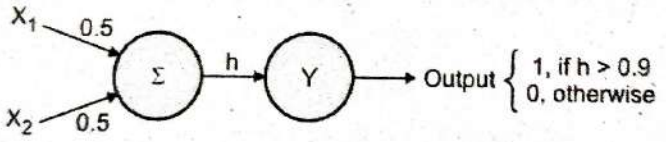


Fig. 6.4

Q. 7 Build a McCulloch-Pitts neuron model to help the bird to eat objects as given in the following table.

Object	Purple?	Round?	Eat?
Blueberry	Yes	Yes	Yes
Golf Ball	No	Yes	No
Flower	Yes	No	No
Leaf	No	No	No
Eggplant	Yes	Yes	Yes
Figs	Yes	Yes	Yes
Rice	No	No	No
Insects	No	Yes	No

Ans. :

The bird is likely to eat an object that is both purple and round. This is very similar to logical AND function. You can simplify the above table as following.

Purple? X_1	Round? X_2	Eat? Y
1	1	1
0	1	0
1	0	0
0	0	0

Assume $W_1 = 0.5$ and $W_2 = 0.5$ and threshold = 0.9.

X_1	X_2	W_1	W_2	Weighted Sum	Y(Threshold = 0.9). Output = 1, if weighted sum > threshold?
0	0	0.5	0.5	0	0 (0 < 0.9)
0	1	0.5	0.5	0.5	0 (0.5 < 0.9)
1	0	0.5	0.5	0.5	0 (0.5 < 0.9)
1	1	0.5	0.5	1	1 (1 > 0.9)

The MCP neuron to help the bird could be implemented as shown in Fig. 6.5.

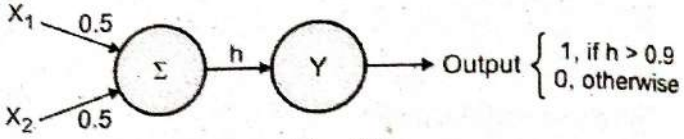


Fig. 6.5

Q. 8 Explain Perceptron. **(6 Marks)**

Ans. :

- One neuron is not that useful. It does not do very much, except fire or not fire based on received inputs and weights (very much like a binary switch). If you feed in the same set of inputs over and over again, the output of the neuron never varies. It either fires or does not fire.
- A major property of a biological neuron is that it learns and responds based on its learning. So, to make artificial neurons more useful and to mimic biological neurons, artificial neurons need to work together as a set of neurons to form an artificial neural network resembling a biological neural network.
- McCulloch and Pitts neuron model only considers the inputs, the weights, and the threshold for generating an output. It ignores the point that learning happens between the neurons in the way that they are connected together. So, in order to make a neuron learn, you need to think how you should change the weights and thresholds of the neurons (connected together in the neural network) so that the neural network learns and gets the right answer more often.

- Perceptron (more precisely perceptron network) was developed in 1958 by the scientist Frank Rosenblatt, inspired by earlier work by Warren McCulloch and Walter Pitts. It is the most basic form of a neural network.
- Note here that modern neural networks use other types of artificial neuron models today. However, it is important for you to understand the concept behind perceptron to understand other advanced artificial neuron models.
- **Definition** : The Perceptron is a collection of McCulloch and Pitts neurons put together with a set of inputs and corresponding weights.
- Fig. 6.6 outlines a perceptron network.

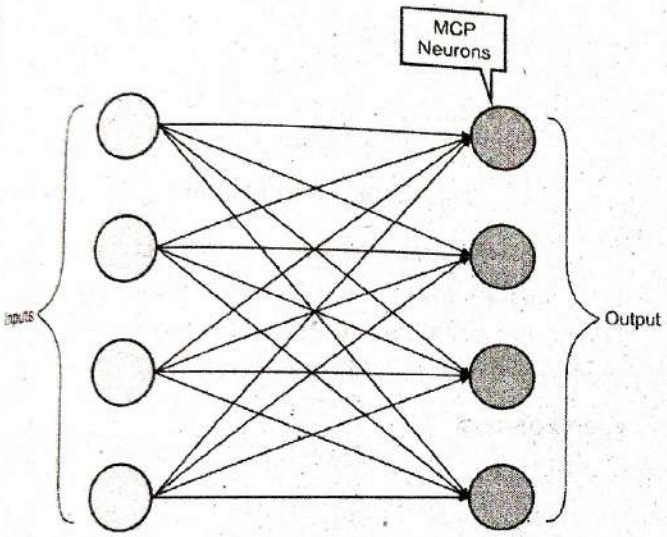


Fig. 6.6 : Perceptron network

- The Perceptron network consists of a set of input nodes (left) connected to McCulloch and Pitts neurons (right) using weighted connections. Note here that input nodes are not neurons. They are just drawn to make the diagram visually symmetric.
- The neurons in the Perceptron are completely independent of each other. Each neuron independently works to aggregate all the weighted inputs and then use the activation function to decide whether to fire. The neurons might get the same input but with different weights attached to them. So, all the neurons receive all the inputs but could be with different weights.
- Also, note here that unlike the diagram of a perceptron network shown here, you may have higher or a smaller number of neurons compared with the number of inputs. So, you could have 4 inputs fed to 3 neurons or 5 inputs fed to 11 neurons. Number of inputs need not be same as number of neurons. The number of inputs is determined by the data. Also, the number of outputs is determined by the target value that you wish to get out of supervised learning.
- Mathematically, in general, in a Perceptron network there will be m inputs and n neurons. In McCulloch and Pitts neuron, the weights were labelled as W_i, where i could range from 1 to m. In Perceptron network, each neuron can get its own specific weight. So, the weights are denoted as W_{ij} where index i denotes the input node number and index j represents the specific neuron to which that weight and input needs to be fed. For example, $W32$ is the weight that connects input node 3 to neuron 2.

Q. 9 With a diagram, explain multilayer perceptron. **(6 Marks)**

Ans. :

- Each layer takes inputs from the previous layer. That way, the inputs to further layers become refined and adjusted and you could build a complex decision making (or classification) system.
- Fig. 6.7 shows a multilayer perceptron network.
- In this multilayer perception network, layer 1 processes the first set of inputs. The outputs from various neurons in layer 1 become inputs for layer 2 neurons. Generally, the outputs from the previous layer are provided weights as well as they are fed to the subsequent layers. So, layer 2 would get weighted inputs from layer 1. Similarly, layer 3 would get weighted inputs from layer 2. Finally, layer 3 neuron could produce the desired output.

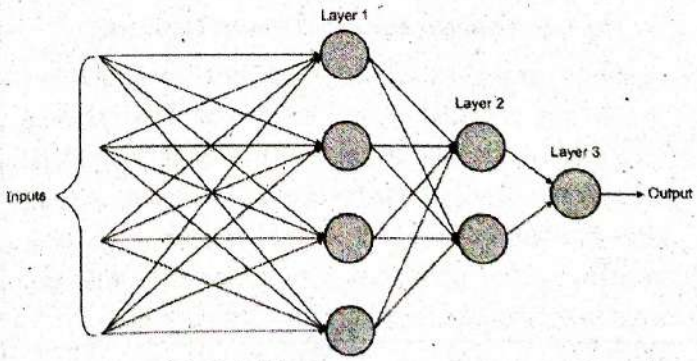


Fig. 6.7 : Multilayer Perceptron Network

- Each perceptron makes a decision by weighing up the results from the previous layer of decision-making. In this way a perceptron in the second layer can make a decision at a more complex and more abstract level than perceptron in the first layer.

Perceptrons at third layer can carry out even more complex decisions. In this way, a many-layer network of perceptrons can engage in sophisticated decision making.

Q. 10 Explain shallow and deep neural networks. (4 Marks)

Ans. :

- Based on the number of layers that a neural network has, it can be either termed as shallow neural network (less than 3 layers) or deep neural network (3 or more than 3 layers).
- The input and output layers are visible and other intermediate layers are considered "hidden".
- More the number of layers, the more complex is the neural network. Deep neural networks are used for complex decision making such as handwriting and speech recognition.
- Fig. 6.8 provides a visual representation of shallow and deep neural network.

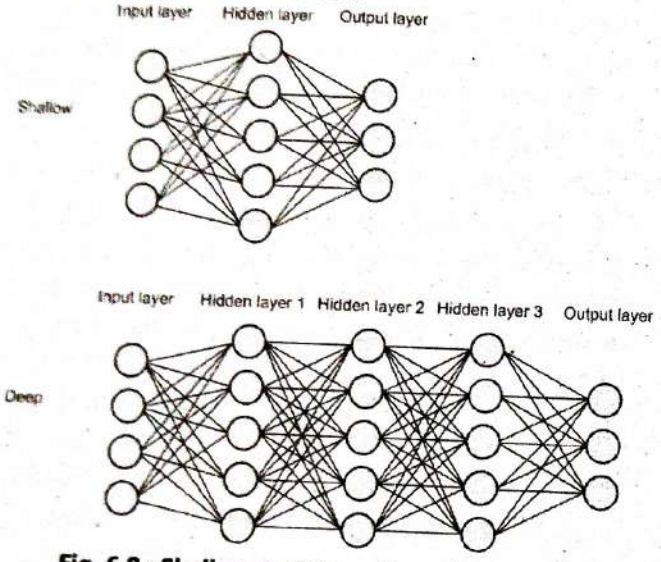


Fig. 6.8 : Shallow and Deep Neural Network

- Deep learning is a subset of machine learning that's based on artificial neural networks. The learning process is deep because the structure of artificial neural networks consists of multiple input, output, and hidden layers. Each layer contains units that transform the input data into information that the next layer can use for a certain predictive task.

Q. 11 Explain a few common neural network architectures. (6 Marks)

Ans. :

- There are several ways in which a neural network could be created. These ways are referred as neural network architecture. Some of the common neural network architectures.

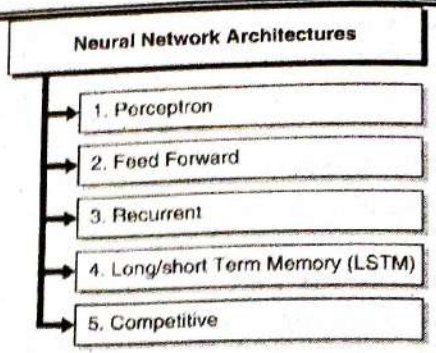


Fig. 6.9(a) : Neural Network Architecture

1. Perceptron

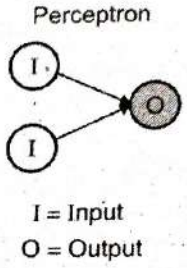


Fig. 6.9(b) : Perceptron

It is the simplest possible neural network. It can have various inputs but does not have any hidden layers between the input and the output. It is commonly used for classification.

2. Feed Forward

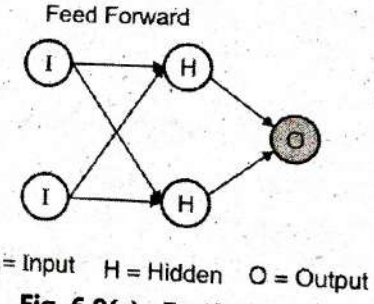


Fig. 6.9(c) : Feed Forward

- In a feed-forward neural network, all the perceptrons are arranged in layers. The hidden layers are not connected with the outside world. There could be one or more hidden layers (so a feed forward network could be shallow or deep as you learnt earlier). Also, every perceptron in one layer is connected with each node in the next layer. Therefore, all the nodes are fully connected.
- In a feed-forward neural network, neurons in the same layer are not interconnected. Hence, the neurons in the same layer work independently of each other. In such a network, algorithms such as backpropagation are commonly used to minimise errors when choosing the weights. Such a neural

network is commonly used for pattern recognition, speech recognition, and character recognition.

3. Recurrent Neural Network

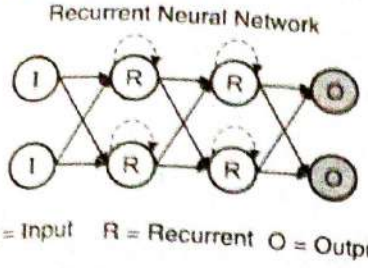


Fig. 6.9(d) : Recurrent Neural Network

- Recurrent neural networks (RNNs) are a variation to feed-forward networks. In this type, each neuron in hidden layers receives an input with a specific delay in time. This type of neural network is used when there is a need to access previous information in current iteration.
- For example, when you are trying to predict the next word in a sentence, you need to know the previously used words first. The neural network computations consider the prior information fed to the neurons.
- However, it is slower when compared with the regular feed forward neural network. It is commonly used for language translation, speech to text conversion, and robotic control.

Simple RNN

- A simple RNN is a fully-connected RNN where the output is to be fed back to input. The Fig. 6.9(e) illustrates a simple RNN.

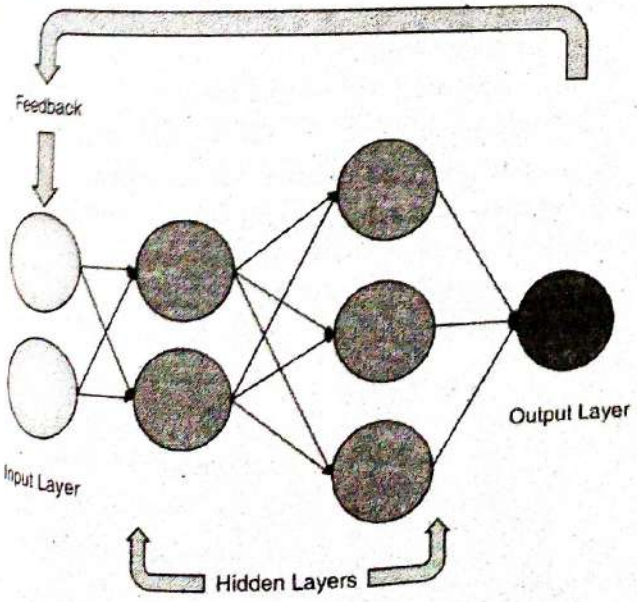


Fig. 6.9(e) : Simple RNN

- It is the most basic form of RNN.

Deep RNN

- An RNN is deep with respect to time. But what if it is deep with respect to space as well, as in a feed-forward network? This is the fundamental notion that has inspired researchers to explore Deep Recurrent Neural Networks, or Deep RNNs.
- In a typical deep RNN, the looping operation is expanded to multiple hidden units. The Fig. 6.9(f) illustrates such an RNN.

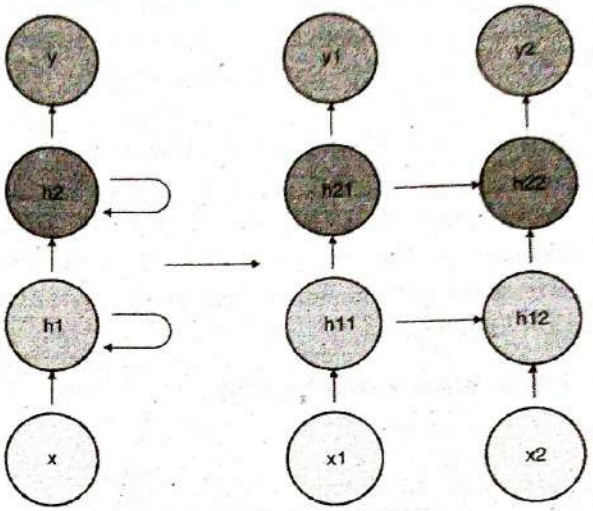


Fig. 6.9(f) : Deep RNN

- An RNN can also be made deep by introducing depth to a hidden unit as follows.

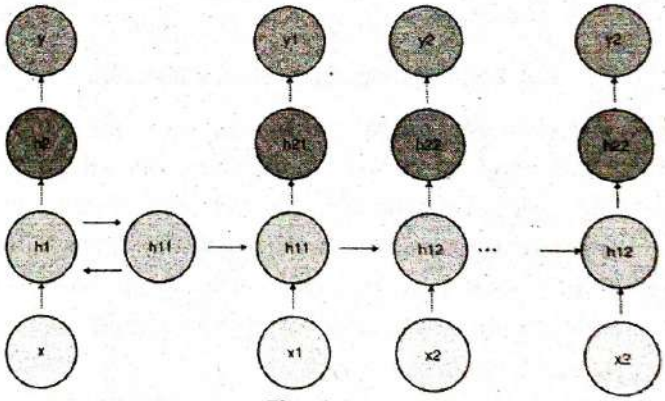


Fig. 6.9(g)

- This model increases the distance traversed by a variable from time t to time t+1.

4. Long / Short Term Memory (LSTM) Neural Network

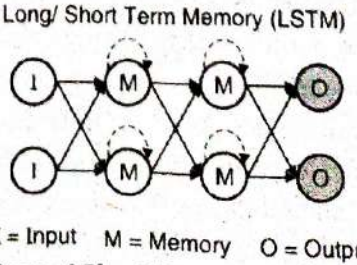


Fig. 6.9(h) : Long / Short Term Memory (LSTM) Neural Network

- LSTM networks treat neurons as a memory cell. They can process data with larger memory gaps. RNNs cannot remember data from a long time ago but LSTMs can. For example, an RNN can predict the next word from what has been written just a few seconds ago.
- But, if you were to predict language spoken in a country, you would need to "remember" which all languages are spoken in that country and which amongst those languages is relevant in the given prediction context. For example, consider that you need to predict the language spoken in the sentence "I grew up in Tamil Nadu. I am fluent in _____". What do you predict? Possibly, a set of languages spoken around that state such as Tamil, Kannada, Malayalam, etc. So, you need LSTM in such a scenario that "remembers" the context for a longer period of time.

5. Competitive Neural Network

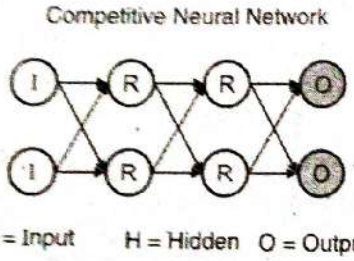


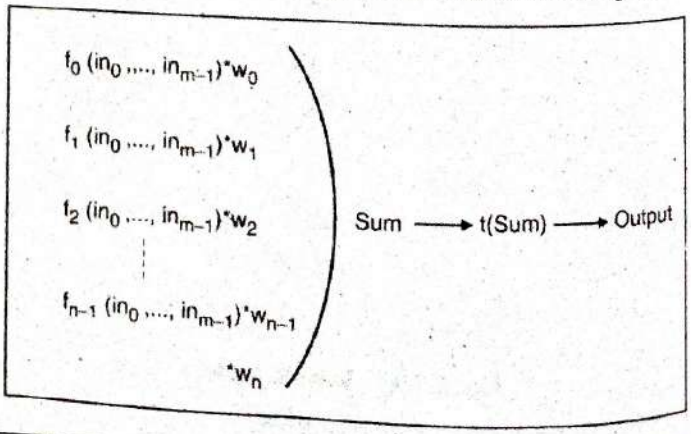
Fig. 6.9(i) : Competitive Neural Network

- Competitive neural networks are structurally similar to feed forward neural networks except that the output neurons are connected amongst themselves usually through negative weights. For a given input pattern, the output neurons of a competitive neural network tend to compete among themselves for that input.
- The output neuron that has the highest activation for the given input pattern is declared the winner. If a neuron wins the competition, its weight is updated according to the competitive learning rule.
- The output neurons compete for the right to respond to a subset of the input data. Competitive learning works by increasing the specialisation of each neuron in the network. Individual neurons of the network learn to specialise on groups of similar patterns and become 'feature detectors' for different classes of input patterns.
- Competitive neural network is commonly used in unsupervised learning for clustering data. So, suppose that the data you want to cluster has got 3 features using which you can decide which cluster the particular data point belongs to. In such a scenario, each neuron could specialise in detecting one of the three features. So, if that particular neuron fires, then it means that the particular feature is present.

Q. 12 Write a short note on Functional Link Artificial Neural Network (FLANN).

Ans. :

- A simple transformation of the input data for a feed forward network often dramatically speeds training. The input layer becomes much larger, sometimes impractically so. However, the number of hidden neurons can usually be greatly decreased or eliminated entirely.
- The basic strategy of the functional link artificial neural network is to generate additional, synthetic (artificial) inputs for a standard feedforward network by applying functions to the original, raw inputs. This is illustrated for a single functional link neuron as following.
- These functions may be applied to individual inputs. They may also be functions of more than one input, generating interaction components. The hope is that with the input information presented to the network in such a variety of forms, no hidden layer will be required to decode the complex patterns in the raw data. To train such a network, the regression technique already described can supply excellent initial estimates of the weights connecting the function outputs to the output neurons. The conjugate gradient algorithm then touches up these weights so as to minimise the error of the outputs.



There are two choices or models.

(1) The first model involves products of the inputs with each other. This is called outer-product or tensor model. The lowest-order version of this model forms products of all pairs of inputs. For example, suppose there are three inputs, (x_0, x_1, x_2). Then, the feedforward network would be presented with six inputs :

$\{x_0, x_1, x_2, x_0, x_1, x_0, x_2, x_1, x_2\}$

The Fig. 6.10 illustrates this.

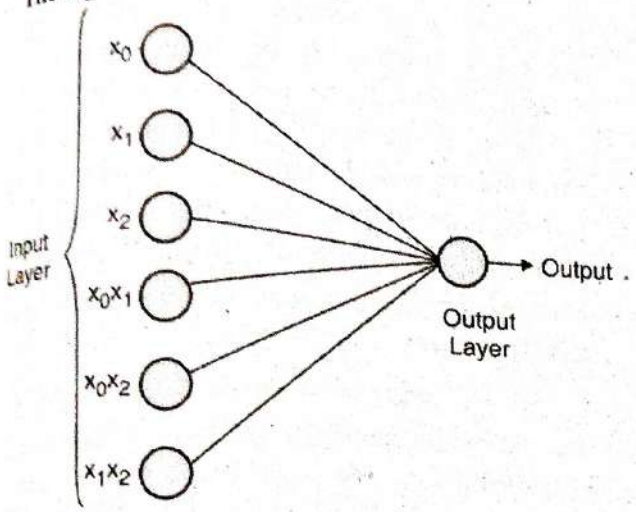


Fig. 6.10

Higher-order versions of this model are formed by multiplying in triples or more. However, it is obvious that the number of terms generated this way rapidly increases beyond the limits of practicality. In fact, even the lowest-order model using only pairs becomes unmanageable for any but very small input vectors. If there are n raw inputs, then there will $n + n * \frac{n-1}{2}$ function values to present to the actual network. It is interesting to note, though, that the infamous XOR problem can be solved without a hidden layer by using $\{x_0, x_1, x_0, x_1\}$ as the input to the network.

(2) The second choice of model does not involve any interaction between inputs. It simply applies one or more univariate functions to each input. These functions may or may not be the same for all inputs. This is typically called the functional expansion model. For example, in polynomial regression you do this when you take an input variable x and generate additional inputs such as x^2, x^3, etc.

In fact, the only difference between polynomial regression and a functional link neural network using polynomial functions is that regression treats the weighted sum of polynomial functions as the desired output. The functional link neural network applies a nonlinear activation function, such as the logistic function to the weighted sum of polynomials to produce the output. This makes the network approach more robust than straight regression in many practical problems involving noisy data, though it does unfortunately mean that iterative techniques like the conjugate gradient algorithm must be used to find the optimum weights.

Q. 13 Write a short note on activation functions. **(4 Marks)**

Ans. :

- **Definition :** Activation function decides whether the artificial neuron would fire or not for a given set of inputs.
- Fig. 6.11 outlines what an activation function is at a high-level.

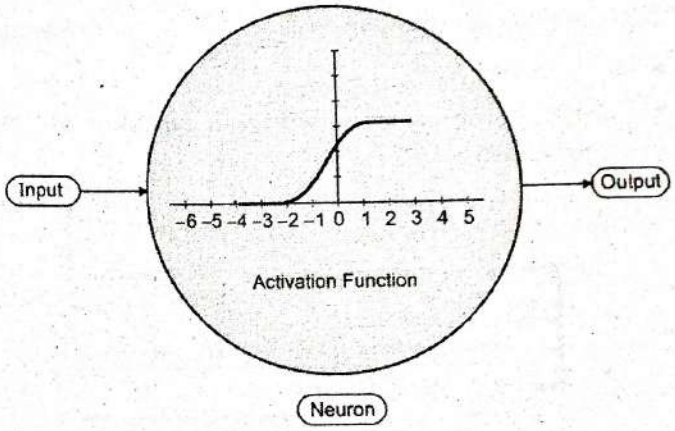


Fig. 6.11 : Activation Functions

- The activation function is a mathematical "gate" in between the input feeding the current neuron and its output going to the next layer. It can be as simple as a step function that turns the neuron output on and off, depending on a rule or threshold. It can also be a transformation that maps the input signals into output signals that are needed for the neural network to function.
- Activation function is a crucial component of any neural network. It determines the output of a neural network model, its accuracy, and also the computational efficiency of training a model. Activation functions also play a major role in deciding the speed at which a neural network can generate output when provided with inputs.
- Activation functions are mathematical equations that work on provided input values. Each neuron in

the network has a corresponding activation function. Different neuron layers may have different activation functions based on their roles and the desired behaviour. It is also desired that activation functions must be computationally efficient because they are calculated across thousands or even millions of neurons for each data sample.

Q. 14 Describe the various types of commonly used activation functions. **(6 Marks)**

OR Explain binary step function. **(6 Marks)**

OR Explain Sigmoid function. **(6 Marks)**

OR Explain logistic function. **(6 Marks)**

OR Explain S-curve and why is it used. **(6 Marks)**

OR Explain TanH function. **(6 Marks)**

OR Explain Hyperbolic Tangent function. **(6 Marks)**

OR Explain ReLU function. **(6 Marks)**

Ans. :

- Some of the common activation functions are as shown in Fig. 6.12.

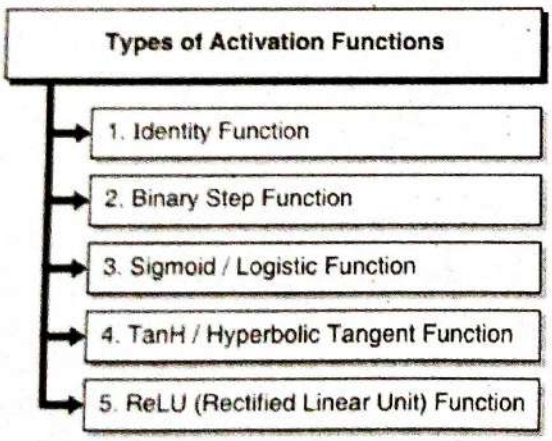


Fig. 6.12(a) : Types of Activation Functions

1. Identity Function

- The simplest activation function is the identity function that passes on the incoming signal as the outgoing signal without any change or processing. It just replicates what is provided as input.
- Mathematically, the identity activation function g (x) is defined as

$$g(x) = x$$

- The Fig. 6.12(b) shows the identity function graphically.

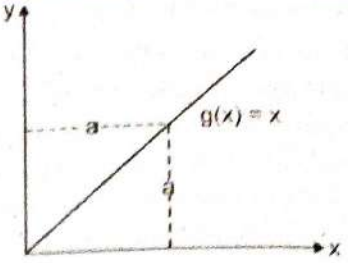


Fig. 6.12(b) : Identity Function Graph

2. Binary Step Function

- A binary step function is a threshold-based activation function. If the input value is above certain threshold, the neuron is activated, and it produces an output of 1. If the input value is below certain threshold, the neuron is not activated, and it produces an output of 0. This is the activation function we have used so far in our discussions.
- Mathematically, the binary step activation function g (x) is defined as

$$g(x) = 1 \text{ when } x > 0, \text{ otherwise } 0$$

- The Fig. 6.12(c) shows the binary step function graphically.

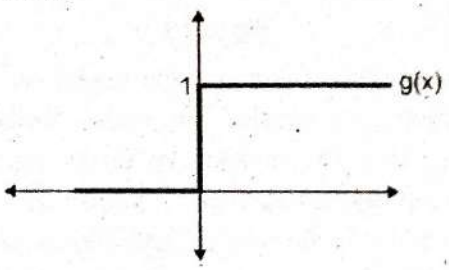


Fig. 6.12(c)

- If you have a specific threshold value, then the step function is mathematically represented as

$$g(x) = 1 \text{ when } x > \theta \text{ (threshold), otherwise } 0$$

- The Fig. 6.12(d) shows the binary step function, having a thresholdθ, graphically.

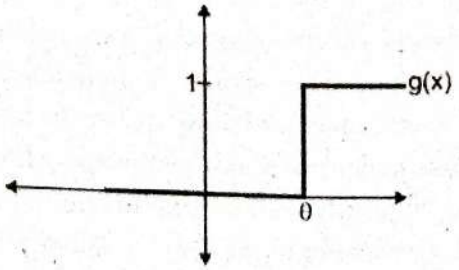


Fig. 6.12(d) : Binary Step Function

- One common problem with a binary step function is that it does not produce different valued outputs (such as 0.1, 0.2, 0.5, etc.). Hence, it cannot be used

for classifying the inputs into one of the several categories. It can only produce two outputs, 0 or 1, and hence can only classify an input into one output category.

Another problem with step function is that it is not continuous and hence cannot be used to compute mathematical differentiation. Quite often, ANN training algorithm requires that the activation function be continuous and mathematical differentiation can be computed.

Sigmoid / Logistic Function

Sigmoid function takes an input and may produce any output between 0 and 1. It is easy to work with and has all the desired properties of an activation function such as the following.

- It is non-linear
- It is continuously differentiable
- It is monotonic
- It has a fixed output range

It is also called as logistic function. Sigmoid functions are one of the most widely used activation functions today.

Mathematically, sigmoid function S (x) is defined as

$$S(x) = \frac{1}{1 + e^{-x}}$$

The Fig. 6.12(e) shows sigmoid function graphically. It is also called as S-curve.

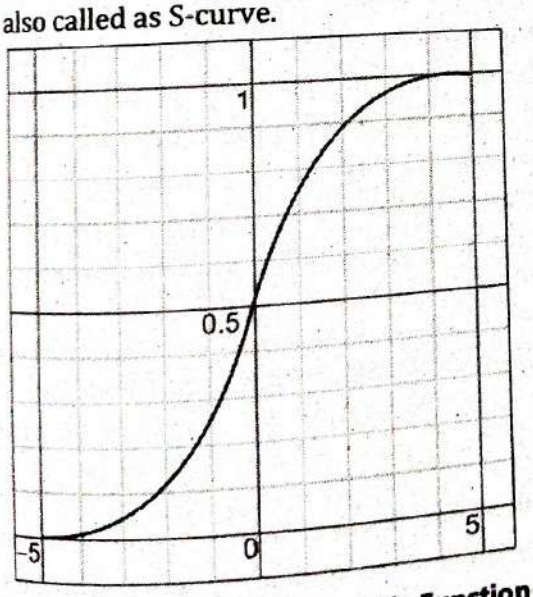


Fig. 6.12(e) : Sigmoid / Logistic Function

From the plot, sigmoid function is non-linear. Hence, unlike step functions, it can give a range of values. Also, any small change in the values of x causes values of y to change significantly. So, it can also work as a step function where required.

- One drawback of sigmoid function is that towards the end of the curve, y values change very less as compared to changes in the x value.
- The gradient (slope) at that region (towards the ends of the curve)becomes increasingly smaller and insignificant. This is also called as "vanishing gradients" problem. When you reach this stage, the neural network model does not learn anything significant thereafter. However, despite this, note that the sigmoid function is most commonly used activation function.

4. TanH / Hyperbolic Tangent Function

- TanH or Hyperbolic Tangent is another commonly used activation function. It is very similar to sigmoid function. It takes an input and may produce any output between -1 and 1. It is easy to work with and has all the desired properties of an activation function such as the following.
  - It is non-linear
  - It is continuously differentiable
  - It is monotonic
  - It has a fixed output range
  - Its output is zero-centred (making it easier to model inputs that have strongly negative, neutral, and strongly positive values)
- Mathematically, TanH function is defined as

$$\text{TanH}(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

- The Fig. 6.12(f) shows TanH function graphically. As you notice, it looks very similar to sigmoid curve.

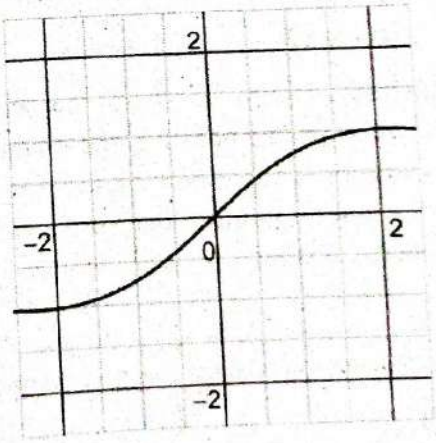


Fig. 6.12(f) : TanH Function

- Similar to sigmoid function, TanH function also has "vanishing gradient" problem where towards the end of the TanH curve, y values change very less as compared to changes in the x value. The gradient

(slope) at that region (towards the ends of the curve) becomes increasingly smaller and insignificant.

5. **ReLU (Rectified Linear Unit) Function**

- ReLU is one of the latest activation functions. It is computationally more efficient and allows the neural network to respond very quickly. It also has the following characteristics.
  - It is non-linear
  - It is continuously differentiable (except at 0)
  - It supports back propagation
  - It does not have a fixed output range
  - It is not zero-centred
  - It does not have vanishing gradient problem
- Mathematically, ReLU function is defined as

$$R(x) = \begin{cases} 0 \text{ for } x \leq 0 \\ x \text{ for } x > 0 \end{cases} = \max(0, x)$$

- The Fig. 6.12(g) shows ReLU function graphically.

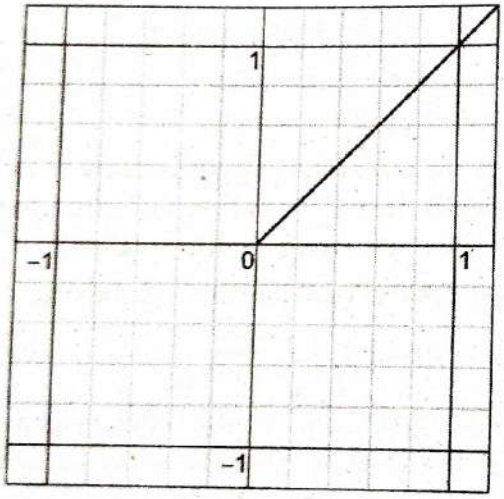


Fig. 6.12(g) : ReLU Function

- One drawback of ReLU function is that when inputs approach zero, or are negative, the gradient of the function becomes zero. Consequently, the neural network cannot perform backpropagation and cannot learn further. It is called Dying ReLU problem.

Q. 15 Explain the learning process involved in implementing an ANN. **(6 Marks)**

Ans. : The learning process of an ANN, at a high-level, could be thought of comprising of the following four steps where you need to make decision about how you are going to implement an ANN.

1. **The number of layers in the network :** As you know, a neural network could be shallow or deep. A shallow network could have just one layer of neurons whereas a deep neural network could have multi-layers. You need to decide what kind of neural network is suitable for your requirement.
2. **The direction of signal flow :** You have also learnt about various NN architectures where signal could just flow in one direction (feed forward) or could also back propagate (such as Recurrent Neural Network). So, you need to choose how you want the signal to flow.
3. **The number of nodes in each layer :** In the case of a multi-layer network, the number of nodes in each layer can be varied. However, the number of nodes or neurons in the input layer is equal to the number of features (fields) of the input data set. Similarly, the number of output nodes will depend on possible outcomes, e.g. number of classes in the case of supervised learning. So, you need to appropriately choose the number of nodes in each of the hidden layers. A larger number of nodes in the hidden layers helps in improving the performance. However, too many nodes may result in overfitting as well as an increased computational expense.
4. **The value of weights attached with each interconnection between neurons :** Deciding the value of weights attached with each interconnection between neurons, so that a specific learning problem can be solved correctly, is quite a difficult problem by itself. You need to adjust the weights until the margin of error is under the desired threshold. It may not be possible to be 100% accurate. But, for example, if your threshold is 95%, you could adjust weights until the results are 95% accurate.

Q. 16 Write a short note on Radial Basis Function. **(4 Marks)**

Ans. :

- Radial Basis Function (RBF) is one of the popular kernel tricks used to classify non-linear data using Support Vector Machines (SVM).
- Mathematically it is denoted as following.

$$K(x, y) = \exp\left(-\frac{(x-y)^2}{2\sigma^2}\right)$$

Often times $\frac{1}{2\sigma^2}$ is represented as γ. So, RBF could be re-written as

$$K(x, y) = e^{-\gamma(x-y)^2}$$

The way it works is that it finds the influence of the datapoint to be classified with the datapoints that are already classified and assigns the higher-dimensional relationship based on the influence. Higher the influence of the already classified datapoint on the datapoint to be classified, higher the chances of the new datapoint to be classified as the one that influenced the most.

So, assume that the point x is known, and the point y is to be classified. Then, RBF calculates the distance between those two points in the infinite dimension as $(x-y)^2$ γ is the scaling factor that you may choose as required to make the distance numbers more significant.

Q. 17 Calculate the radial distance of point A having value of 6 with respect to point B and C having value of 8 and 12 respectively. **(4 Marks)**

Ans. :

Assume $\gamma = 1$

Distance of point A with respect to

$$B = e^{-\gamma(x-y)^2}$$

$$= e^{-1(6-8)^2} = e^{-4} = 0.018$$

Distance of point A with respect to

$$C = e^{-\gamma}(x-y)^2$$

$$= e^{-1(6-12)^2} = e^{-36} = 0$$

So, point B has higher influence over point A than point C.

Q. 18 Demonstrate how you could use RBF over XOR function to draw a linear SVM. **(6 Marks)**

Ans. :

The truth table for XOR function is as following.

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

If you plot these points, you get the layout as shown in Fig. 6.13(a).

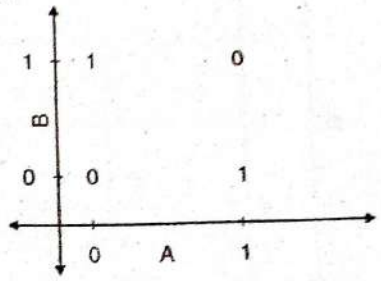


Fig. 6.13(a)

No matter how hard you try, you would not be able to separate out these points as in the case of a linear SVM. Let's see how RBF helps.

Note that there are two classes here y = {0, 1} and four training samples, two for each class.

{0, 1} and {1, 0} produce 1 (+ ve class) whereas {0, 0} and {1, 1} produce 0 (- ve class). Assume $\gamma = 1$.

Let's choose elements {0, 1} and {1, 0} as the basis for RBF.

X (Input)	Basis = {0, 1}	Basis = {1, 0}
{0, 0}	$e^{-\gamma(x-y)^2} = e^{-1(\{0,0\}-\{0,1\})^2} = e^{-1(0-0\,0-1)^2} = e^{-1}$ = 0.37	$e^{-\gamma(x-y)^2} = e^{-1(\{0,0\}-\{1,0\})^2} = e^{-1(0-1\,0-0)^2} = e^{-1}$ = 0.37
{0, 1}	$e^{-\gamma(x-y)^2} = e^{-1(\{0,1\}-\{0,1\})^2} = e^{-1(0-0\,1-1)^2} = e^{0} = 1$	$e^{-\gamma(x-y)^2} = e^{-1(\{0,1\}-\{1,0\})^2} = e^{-1(0-1\,1-0)^2} = e^{-4}$ = 0.018
{1, 0}	$e^{-\gamma(x-y)^2} = e^{-1(\{1,0\}-\{0,1\})^2} = e^{-1(1-0\,0-1)^2} = e^{-4}$ = 0.018	$e^{-\gamma(x-y)^2} = e^{-1(\{1,0\}-\{1,0\})^2} = e^{-1(1-1\,0-0)^2} = e^{0} = 1$
{1, 1}	$e^{-\gamma(x-y)^2} = e^{-1(\{1,1\}-\{0,1\})^2} = e^{-1(1-0\,1-1)^2} = e^{-1}$ = 0.37	$e^{-\gamma(x-y)^2} = e^{-1(\{1,1\}-\{1,0\})^2} = e^{-1(1-1\,1-0)^2} = e^{-1}$ = 0.37

So, RBF does the following transformations to the respective points.

Old Point	New Point
{0, 0}	{0.37, 0.37}
{0, 1}	{1, 0.018}
{1, 0}	{0.018, 1}
{1, 1}	{0.37, 0.37}

If you plot these points, you see that they are now clearly separatable as in the case of a linear SVM.

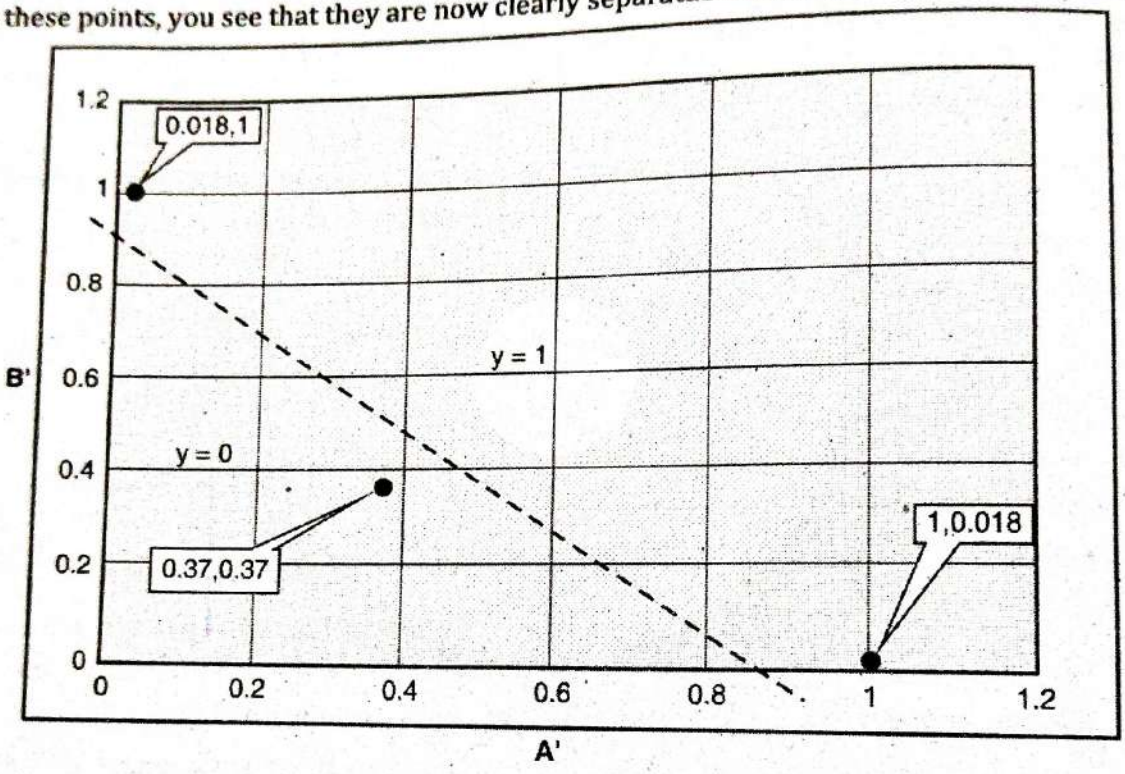


Fig. 6.13(b)

Points {1, 0.018} and {0.018, 1} give output y = 1.

Points {0.37, 0.37) and {0.37, 0.37) give output y = 0.

Hence, as you see, RBF transformed the XOR points to make them linearly separable.