SUBJECT CODE : 410242

Choice Based Credit System

SAVITRIBAI PHULE PUNE UNIVERSITY - 2019 SYLLABUS

B.E. (Computer) Semester - VII

MACHINE LEARNING

(For END SEM Exam - 70 Marks)

Iresh A. Dhotre

M.E. (Information Technology)

Ex-Faculty, Sinhgad College of Engineering,

Pune.

FEATURES

- ☑ Written by Popular Authors of Text Books of Technical Publications
- ☑ Covers Entire Syllabus
- ☑ Question - Answer Format
- ☑ Exact Answers and Solutions
- ☑ Solved Model Question Paper (As Per 2019 Pattern)

SOLVED SPPU QUESTION PAPERS

- March - 2019
- June - 2022

DECODE®

A Guide For Engineering Students

DECODE®

A Guide For Engineering Students

MACHINE LEARNING

(For END SEM Exam - 70 Marks)

SUBJECT CODE : **410242**

B.E. (Computer Engineering) Semester - VII

© Copyright with Technical Publications
All publishing rights (printed and ebook version) reserved with Technical Publications.
No part of this book should be reproduced in any form, Electronic, Mechanical, Photocopy or any information storage and retrieval system without prior permission in writing, from Technical Publications, Pune.

Published by :

TECHNICAL® PUBLICATIONS
SINCE 1993 An Up-Thrust for Knowledge

Amit Residency, Office No.1, 412, Shaniwar Peth,
Pune - 411030, M.S. INDIA Ph.: +91-020-24495496/97
Email : info@technicalpublications.in
Website : www.technicalpublications.in

Printer :
Yogiraj Printers & Binders, Sr.No. 10/1A,Ghule Industrial Estate, Nanded Village Road,
Tal. - Haveli, Dist. - Pune - 411041.

ISBN 978-93-5585-241-0

9 789355 852410

SPPU 19

SYLLABUS

Machine Learning - (410242)

Credit	Examination Scheme :
03	End-Sem (Paper) : 70 Marks

Unit III Supervised Learning : Regression

Bias, Variance, Generalization, Underfitting, Overfitting, Linear regression, Regression : Lasso regression, Ridge regression, Gradient descent algorithm. Evaluation Metrics : MAE, RMSE, R2 **(Chapter - 3)**

Unit IV Supervised Learning : Classification

Classification : K-nearest neighbour, Support vector machine. Ensemble Learning : Bagging, Boosting, Random Forest, Adaboost.

Binary-vs-Multiclass Classification, Balanced and Imbalanced Multiclass Classification Problems, Variants of Multiclass Classification: One-vs-One and One-vs-All

Evaluation Metrics and Score : Accuracy, Precision, Recall, Fscore, Cross-validation, Micro-Average Precision and Recall, Micro-Average F-score, Macro-Average Precision and Recall, Macro-Average F-score. **(Chapter - 4)**

Unit V Unsupervised Learning

K-Means, K-medoids, Hierarchical, and Density-based Clustering, Spectral Clustering. Outlier analysis: introduction of isolation factor, local outlier factor.

Evaluation metrics and score : elbow method, extrinsic and intrinsic methods **(Chapter - 5)**

Unit VI Introduction To Neural Networks

Artificial Neural Networks : Single Layer Neural Network, Multilayer Perceptron, Back Propagation Learning, Functional Link Artificial Neural Network, and Radial Basis Function Network, Activation functions,

Introduction to Recurrent Neural Networks and Convolutional Neural Networks **(Chapter - 6)**

TABLE OF CONTENTS

Unit III

Chapter - 3 Supervised Learning : Regression (3 - 1) to (3 - 26)

3.1 Bias and Variance ..... 3 - 1

3.2 Underfitting and Overfitting ..... 3 - 5

3.3 Linear Regression ..... 3 - 9

3.4 Regression : Lasso Regression, Ridge Regression ..... 3 - 17

3.5 Gradient Descent Algorithm ..... 3 - 21

3.6 Evaluation Metrics : MAE, RMSE, R2 ..... 3 - 23

Unit IV

Chapter - 4 Supervised Learning : Classification (4 - 1) to (4 - 34)

4.1 Classification : K-Nearest Neighbour ..... 4 - 1

4.2 Support Vector Machine ..... 4 - 6

4.3 Ensemble Learning : Bagging, Boosting, Random Forest, Adaboost ..... 4 - 12

4.4 Binary-vs-Multiclass Classification ..... 4 - 19

4.5 Variants of Multiclass Classification : One-vs-One and One-vs-All ..... 4 - 23

4.6 Evaluation Metrics and Score ..... 4 - 25

4.7 Cross-Validation ..... 4 - 30

4.8 Micro-Average ..... 4 - 32

4.9 Macro-Average ..... 4 - 34

Unit V

Chapter - 5 Unsupervised Learning (5 - 1) to (5 - 32)

5.1 Introduction to Clustering ..... 5 - 1

5.2 K-Means and K-Medoids ..... 5 - 8

5.3 Hierarchical Clustering ..... 5 - 17

5.4 Density-Based Clustering ..... 5 - 22

5.5 Outlier Analysis ..... 5 - 24

5.6 Evaluation Metrics and Score ..... 5 - 26

Unit VI

Chapter - 6 Introduction to Neural Networks (6 - 1) to (6 - 28)

6.1 Artificial Neural Networks ..... 6 - 1

6.2 Multilayer Perceptron ..... 6 - 8

6.3 Back Propagation Learning ..... 6 - 10

6.4 Functional Link Artificial Neural Network ..... 6 - 13

6.5 Radial Basis Function Network ..... 6 - 15

6.6 Activation Functions ..... 6 - 17

6.7 Introduction to Recurrent Neural Networks ..... 6 - 21

6.8 Convolutional Neural Networks ..... 6 - 24

Solved Model Question Paper (M - 1) to (M - 3)

Unit III

3 Supervised Learning : Regression

3.1 : Bias and Variance

Q.1 What is bias in machine learning ?

Ans. : • Bias is a phenomenon that skews the result of an algorithm in favor or against an idea.

- Bias is considered a systematic error that occurs in the machine learning model itself due to incorrect assumptions in the ML process.
- Bias is the difference between the average prediction of our model and the correct value which we are trying to predict. Fig. Q.1.1 shows bias.

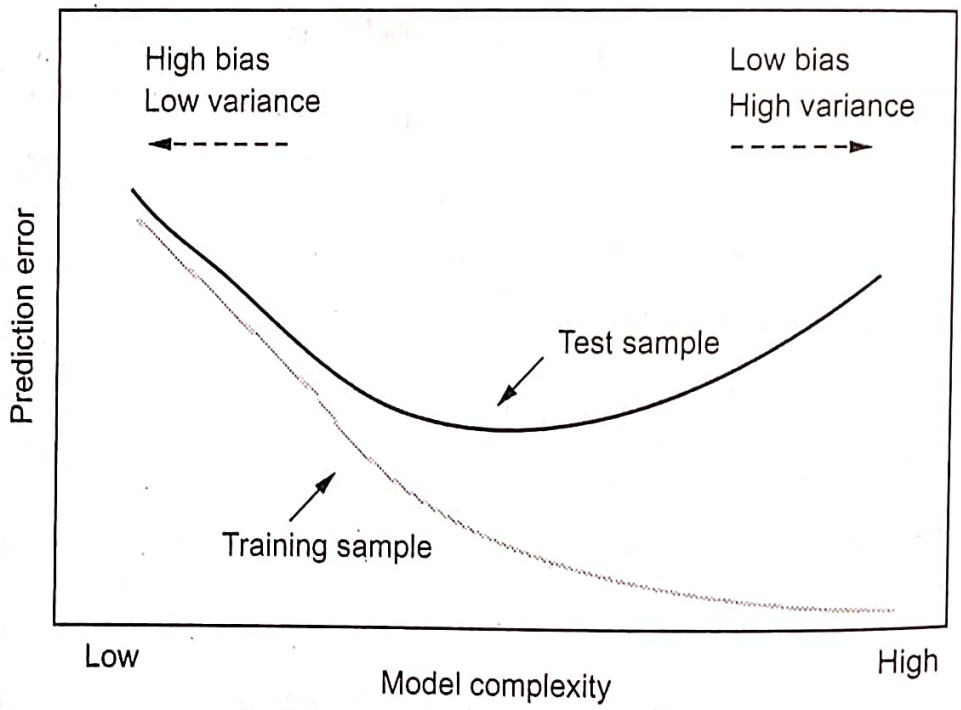


Fig. Q.1.1

- Bias and variance are components of reducible error. Reducing errors requires selecting models that have appropriate complexity and flexibility, as well as suitable training data.
- Low bias : A low bias model will make fewer assumptions about the form of the target function.
- High bias : A model with a high bias makes more assumptions and the model becomes unable to capture the important features of our dataset. A high bias model also cannot perform well on new data.
- Examples of machine learning algorithms with low bias are decision trees, k-nearest neighbours and support vector machines.
- An algorithm with high bias is linear regression, linear discriminant analysis and logistic regression.

Q.2 How to reduce the high bias ?

Ans. : • If the average predicted values are far off from the actual values then the bias is high. High bias causes algorithm to miss relevant relationship between input and output variable.

- When a model has a high bias then it implies that the model is too simple and does not capture the complexity of data thus underfitting the data.
- Low variance means there is a small variation in the prediction of the target function with changes in the training data set. At the same time, High variance shows a large variation in the prediction of the target function with changes in the training dataset.
- High bias can be identified when we have high training error and validation error or test error is same as training error.
- Following methods are used to reduce high bias :
  1. Increase the input features as the model is underfitted.
  2. Decrease the regularization term.
  3. Use more complex models, such as including some polynomial features.

Q.3 Define variance. Explain low and high variance ? How to reduce high variance ?

Ans. : • Variance indicates how much the estimate of the target function will alter if different training data were used. In other words, variance describes how much a random variable differs from its expected value.

- Variance is based on a single training set. Variance measures the inconsistency of different predictions using different training sets, it's not a measure of overall accuracy.
- Low variance means there is a small variation in the prediction of the target function with changes in the training data set. High variance shows a large variation in the prediction of the target function with changes in the training dataset.
- Variance comes from highly complex models with a large number of features.
  1. Models with high bias will have low variance.
  2. Models with high variance will have a low bias.
- Following methods are used to reduce high variance :
  1. Reduce the input features or number of parameters as a model is overfitted.
  2. Do not use a much complex model.
  3. Increase the training data.
  4. Increase the regularization term.

Q.4 Explain bias-variance trade off.

Ans. : • In the experimental practice we observe an important phenomenon called the bias variance dilemma.

- In supervised learning, the class value assigned by the learning model built based on the training data may differ from the actual class value. This error in learning can be of two types, errors due to 'bias' and error due to 'variance'.
  1. Low-bias, low-variance : The combination of low bias and low variance shows an ideal machine learning model. However, it is not possible practically.

2. Low-bias, high-variance : With low bias and high variance, model predictions are inconsistent and accurate on average. This case occurs when the model learns with a large number of parameters and hence leads to an overfitting
3. High-bias, low-variance : With high bias and low variance, predictions are consistent but inaccurate on average. This case occurs when a model does not learn well with the training dataset or uses few numbers of the parameter. It leads to underfitting problems in the model.
4. High-bias, high-variance : With high bias and high variance, predictions are inconsistent and also inaccurate on average.

Q.5 What is difference between bias and variance ?

Ans. :

Sr. No.	Bias	Variance
1.	Bias is the difference between the average prediction and the correct value.	Variance is the amount that the prediction will change if different training data sets were used.
2.	The model is incapable of locating patterns in the dataset that it was trained on and it produces inaccurate results for both seen and unseen data.	The model recognizes the majority of the dataset's patterns and can even learn from the noise or data that isn't vital to its operation.
3.	Low bias models : k-nearest neighbors, decision trees and support vector machines.	Low variance models : Linear regression and logistic regression.
4.	High bias models : Linear regression and logistic regression.	High variance models : k-Nearest neighbors, decision trees and support vector machines.

3.2 : Underfitting and Overfitting

Q.6 What is overfitting and underfitting in machine learning model ? Explain with example. ☞ [SPPU : June-22, Marks 6]

Ans. • Fig. Q.6.1 shows underfitting and overfitting.

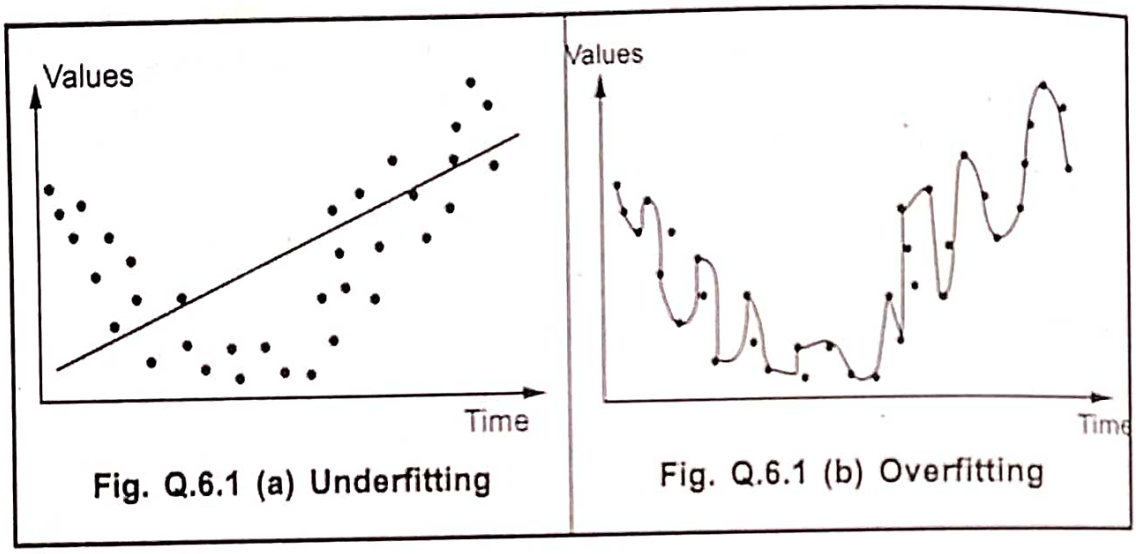


Fig. Q.6.1 (a) Underfitting

Fig. Q.6.1 (b) Overfitting

- **Underfitting** occurs when the model is unable to match the input data to the target data. This happens when the model is not complex enough underfitting and overfitting to match all the available data and performs poorly with the training dataset.
- These kind of models are very simple to capture the complex patterns in data like linear and logistic regression.

Underfitting examples :

1. The learning time may be prohibitively large and the learning stage was prematurely terminated.
2. The learner did not use a sufficient number of iterations.
3. The learner tries to fit a straight line to a training set whose examples exhibit a quadratic nature.

- **Overfitting** relates to instances where the model tries to match non-existent data. This occurs when dealing with highly complex models where the model will match almost all the given data points and perform well in training datasets. However, the model would not be

able to generalize the data point in the test data set to predict the outcome accurately.

- These models have low bias and high variance. These models are very complex like decision trees which are prone to overfitting.
- Reasons for overfitting are noisy data, training data set is too small and large number of features.

Q.7 How to avoid overfitting and underfitting model ?

Ans. : 1. Following methods are used to avoid overfitting :

- Cross validation
- Training with more data
- Removing features
- Early stopping the training
- Regularization
- Ensembling

2. Following methods are used to avoid underfitting :

- By growing the education time of the model.
- By increasing the wide variety of functions.

Q.8 How do we know if we are underfitting or overfitting ?

Ans. :

1. If by increasing capacity we decrease generalization error, then we are underfitting, otherwise we are overfitting.
2. If the error in representing the training set is relatively large and the generalization error is large, then underfitting.
3. If the error in representing the training set is relatively small and the generalization error is large, then overfitting;
4. There are many features but relatively small training set.

Q.9 Explain the Fig. Q.9.1 (a), (b) and (c).

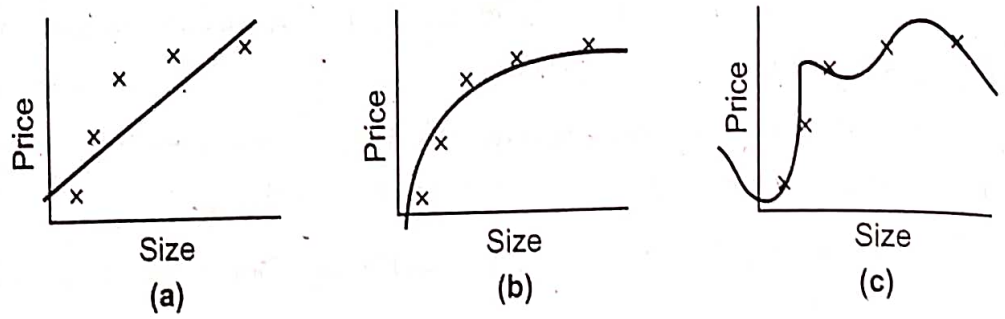


Fig. Q.9.1

Ans. : • Given Fig. Q.9.1 is related to overfitting and underfitting.

Underfitting (High bias and low variance) :

- A statistical model or a machine learning algorithm is said to have underfitting when it cannot capture the underlying trend of the data.
- It usually happens when we have less data to build an accurate model and also when we try to build a linear model with a non-linear data.

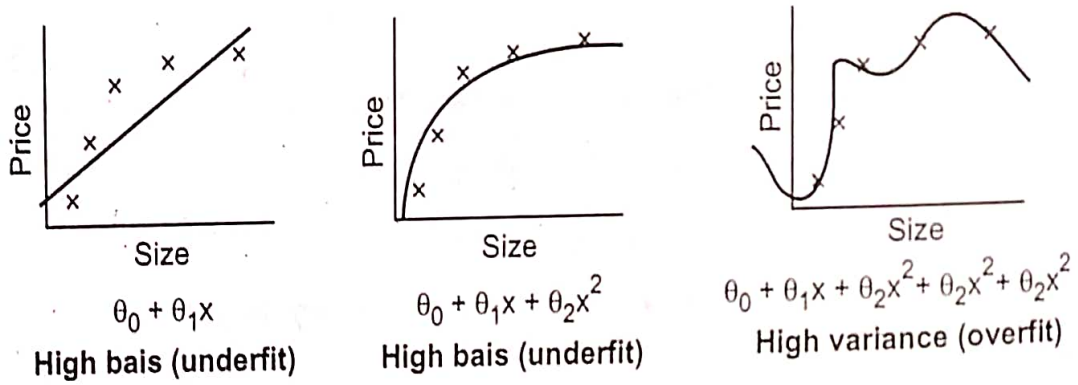


Fig. Q.9.2

- In such cases the rules of the machine learning model are too easy and flexible to be applied on such minimal data and therefore the model will probably make a lot of wrong predictions.
- Underfitting can be avoided by using more data and also reducing the features by feature selection.

Overfitting (High variance and low bias) :

- A statistical model is said to be overfitted, when we train it with a lot of data.

- When a model gets trained with so much of data, it starts learning from the noise and inaccurate data entries in our data set.
- Then the model does not categorize the data correctly, because of too many details and noise.
- The causes of overfitting are the non-parametric and non-linear methods because these types of machine learning algorithms have more freedom in building the model based on the dataset and therefore they can really build unrealistic models.
- A solution to avoid overfitting is using a linear algorithm if we have linear data or using the parameters like the maximal depth if we are using decision trees.

Q.10 Explain difference between overfitting and underfitting.

Ans. :

Sr. No.	Overfitting	Underfitting
1.	Very low training error.	High training error.
2.	It is too complex.	Model is too simple.
3.	Support high variance and low bias.	Support low variance and high bias.
4.	Smaller quantity of features.	Larger quantity of feature.
5.	Performs more regularization.	Performs less regularization.
6.	Training error much lower than the test error.	Training error close to test error.

Q.11 What is goodness of fit ?

Ans. : • The goodness of fit of a model explains how well it matches a set of observations. Usually, the goodness of fit indicators summarizes the disparity between observed values and the model's anticipated values.

- In machine learning algorithm, a good fit is when both the training data error and the test data are minimal. As the algorithm learns, the mistake in the training data for the modal is decreasing over time and so is the error on the test dataset.
- If we train for too long, the training dataset performance may continue to decline due to the model being overfitting and learning the irrelevant detail and noise in the training dataset. At the same time, the test set error begins to rise again as the ability of the model to generalize decreases.
- The errors in the test dataset start increasing, so the point, just before the raising of errors, is the good point, and we can stop here for achieving a good model.

3.3 : Linear Regression

Q.12 Define and explain regression with its model.

Ans. : • Regression finds correlations between dependent and independent variables. If the desired output consists of one or more continuous variable, then the task is called as regression.

- Therefore, regression algorithms help predict continuous variables such as house prices, market trends, weather patterns, oil and gas prices etc.
- Fig. Q.12.1 shows regression.

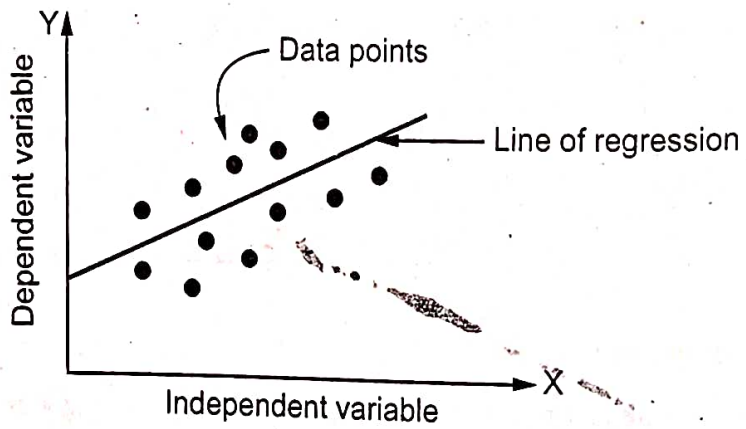


Fig. Q.12.1 Regression

- When the targets in a dataset are real numbers, the machine learning task is known as regression and each sample in the dataset has a real-valued output or target.
- Regression analysis is a set of statistical methods used for the estimation of relationships between a dependent variable and one or more independent variables. It can be utilized to assess the strength of the relationship between variables and for modelling the future relationship between them.
- The two basic types of regression are linear regression and multiple linear regression.

Q.13 Explain univariate regression.

Ans. : • Univariate data is the type of data in which the result depends only on one variable. If there is only one input variable then we call it 'Single Variable Linear Regression' or 'Univariate Linear Regression.

- The function that we are trying to develop looks like this :

$$h_\theta(x) = \theta_0 + \theta_1 x$$

$$y = mx + b$$

- That is because linear regression is essentially the algorithm for finding the line of best fit for a set of data.
- The algorithm finds the values for θ_0 and θ_1 that best fit the inputs and outputs given to the algorithm. This is called univariate linear regression because the ?? parameters only go up to 1.
- The univariate linear regression algorithm is much simpler than the one for multivariate.

Q.14 When is it suitable to use linear regression over classification ?

Ans. : • Linear regression is a statistical method that allows us to summarize and study relationships between two continuous (quantitative) variables.

- The objective of a linear regression model is to find a relationship between the input variables and a target variable.

1. One variable, denoted x, is regarded as the predictor, explanatory, or independent variable.
2. The other variable, denoted y, is regarded as the response, outcome, or dependent variable.

- Regression models predict a continuous variable, such as the sales made on a day or predict temperature of a city. Let's imagine that we fit a line with the training points that we have. If we want to add another data point, but to fit it, we need to change existing model.
- This will happen with each data point that we add to the model; hence, linear regression isn't good for classification models.
- Regression estimates are used to explain the relationship between one dependent variable and one or more independent variables. Classification predicts categorical labels (classes), prediction models continuous - valued functions. Classification is considered to be supervised learning.
- Classifies data based on the training set and the values in a classifying attribute and uses it in classifying new data. Prediction means models continuous - valued functions, i.e. predicts unknown or missing values.

Q.15 Why do we need to regularize in regression ? Explain.

Ans. : • Regression model fails to generalize on unseen data. This could happen when the model tries to accommodate all kinds of changes in the data including those belonging to both the actual pattern and also the noise.

- As a result, the model ends up becoming a complex model having significantly high variance due to overfitting, thereby impacting the model performance (accuracy, precision, recall, etc.) on unseen data.
- Regularization needed for reducing overfitting in the regression model.
- Regularization techniques are used to calibrate the coefficients of the determination of multi-linear regression models in order to minimize the adjusted loss function.

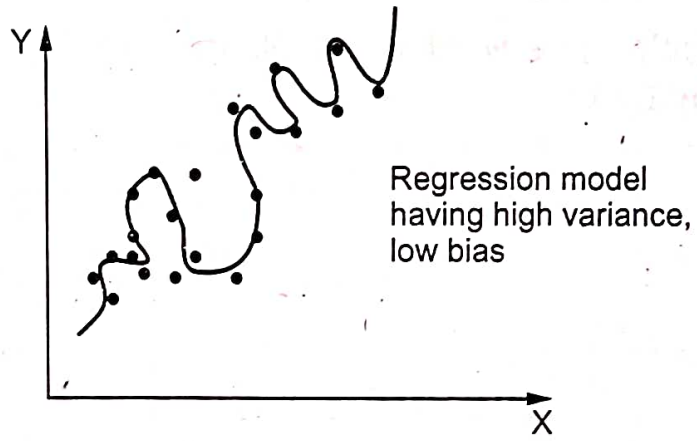


Fig. Q.15.1 Regression model before regularization

Fig. Q.15.2 Regression model after regularization

- Regularization methods provide a means to control our regression coefficients, which can reduce the variance and decrease our of sample error.
- The goal is to reduce the variance while making sure that the model does not become biased (underfitting). After applying the regularization technique, the above model could be obtained.

Q.16 Consider following data for 5 students.

Each X_i (i = 1 to 5) represents the score of ith student in standard X and corresponding Y_i (i = 1 to 5) represents the score of ith student in standard XII.

i) What linear regression equation best predicts standard XIIth score ?

ii) Find regression line that fits best for given sample data.

iii) How to interpret regression equation ?

iv) If a student's score is 80 in std X, then what is his expected score in XII standard ?

Student	Score in X standard (X_i)	Score in XII standard (Y_i)
1	95	85
2	85	95
3	80	70
4	70	65
5	60	70

Ans. : • The mean of the x values denoted by $\overline{X}$

- The mean of the y values denoted by $\overline{Y}$
- The standard deviation of the x values (denoted S_X)
- The standard deviation of the y values (denoted S_Y)

X_i	Y_i	X^2	XY	Y^2	$(X_i - \overline{X})$	$(Y_i - \overline{Y})$
95	85	9025	8075	7225	17	8
85	95	7225	8075	9025	7	18
80	70	6400	5600	4900	2	– 7
70	65	4900	4550	4225	– 8	– 12
60	70	3600	4200	4900	– 18	– 7
$\Sigma X = 390$	$\Sigma Y = 385$	$\Sigma X^2 = 31150$	$\Sigma XY = 30500$	$\Sigma Y^2 = 30275$		

Find : $\overline{X}, \overline{Y}, S_X, S_Y$

$$\overline{X} = \frac{\sum X_i}{n}, \quad \overline{Y} = \frac{\sum Y_i}{n}, \quad S_X = \sqrt{\frac{\sum (X_i - \overline{X})^2}{n-1}}, \quad S_Y = \sqrt{\frac{\sum (Y_i - \overline{Y})^2}{n-1}}$$

$$\overline{X} = 390/5 = 78, \quad \overline{Y} = 385/5 = 77,$$

$$S_X = \sqrt{(390-78)^2/4} = \sqrt{24336} = 156$$

$$S_Y = \sqrt{(385-77)^2/4} = \sqrt{23716} = 154$$

We also need to compute the squares of the deviation scores :

X_i	Y_i	$(X_i - \overline{X})^2$	$(Y_i - \overline{Y})^2$	$(X_i - \overline{X}) \times (Y_i - \overline{Y})$
95	85	289	64	136
85	95	49	324	136
80	70	4	49	– 14
70	65	64	144	96
60	70	324	49	126
$\Sigma X = 390$	$\Sigma Y = 385$	730	630	$\Sigma(X_i - \overline{X}) \times (Y_i - \overline{Y}) = 480$

The regression equation is a linear equation of the form : $\hat{Y} = \beta_0 + \beta_1 X$

First, we solve for the regression coefficient (β_1) :

$$\beta_1 = \sum[(X_i - \overline{X}) \times (Y_i - \overline{Y})] / \sum [(X_i - \overline{X})^2]$$

$$= 480 / 730 = 0.657$$

Once we know the value of the regression coefficient (β_1), we can solve for the regression slope (β_0) :

$$\beta_0 = \overline{Y} - \beta_1 \overline{X}$$

$$\beta_0 = 385 - 0.657 * 390 = 128.77$$

Therefore, the regression equation is $\hat{Y} = 128.77 + 0.657\ X$

Q.17 Consider following data :

i) Find values of β_0 *and* β_1 w.r.t. linear regression model which best fits given data.

ii) Interpret and explain equation of regression line.

iii) If new person rates "Bahubali -Part I" as 3 then predict the rating of same person for "Bahubali -Part II".

Person	X_i = Rating for movie "Bahubali Part - I" by i^{th} person	Y_i = Rating for movie "Bahubali Part - II" by i^{th} person
1st	4	3
2nd	2	4
3rd	3	2
4th	5	5
5th	1	3
6th	3	1

Ans. :

X_i	Y_i	X^2	XY	Y^2	$(X_i - \overline{X})$	$(Y_i - \overline{Y})$
4	3	16	12	9	1	0
2	4	4	8	16	– 1	1
3	2	9	6	4	0	– 1
5	5	25	25	25	2	2
1	3	1	3	9	– 2	0
3	1	9	3	1	0	– 2
$\sum X = 18$	$\sum Y = 18$	$\sum X^2 = 64$	$\sum XY = 57$	$\sum Y^2 = 64$		

Find : $\overline{X}$, $\overline{Y}$, S_X, S_Y

$$\overline{X} = \frac{\sum X_i}{n}, \quad \overline{Y} = \frac{\sum Y_i}{n}, \quad S_X = \sqrt{\frac{\sum (X_i - \overline{X})^2}{n-1}}, \quad S_Y = \sqrt{\frac{\sum (Y_i - \overline{y})^2}{n-1}}$$

$$\overline{X} = 18/6 = 3, \quad \overline{Y} = 18/6 = 3,$$

$$S_X = \sqrt{(18-3)^2 / 5} = \sqrt{45}$$

$$S_Y = \sqrt{(18-3)^2 / 5} = \sqrt{45}$$

We also need to compute the squares of the deviation scores :

X_i	Y_i	$(X_i - \overline{X})^2$	$(Y_i - \overline{Y})^2$	$(X_i - \overline{X}) \times (Y_i - \overline{Y})$
4	3	1	0	0
2	4	1	1	– 1
3	2	0	1	0
5	5	4	4	4
1	3	4	0	0
3	1	0	4	0
$\Sigma X = 18$	$\Sigma Y = 18$	10	10	$\Sigma(X_i - \overline{X}) \times (Y_i - \overline{Y}) = 3$

The regression equation is a linear equation of the form: $\hat{Y} = \beta_0 + \beta_1 X$

First, we solve for the regression coefficient (β_1) :

$$\beta_1 = \sum [(X_i - \overline{X}) \times (Y_i - \overline{Y})] / \sum [(X_i - \overline{X})^2]$$

$$= 3/10 = 0.3$$

Once we know the value of the regression coefficient (β_1), we can solve for the regression slope (β_0):

$$\beta_0 = \overline{Y} - \beta_1 \overline{X}$$

$$\beta_0 = 18 + 0.3 * 18 = 12.6$$

Therefore, the regression equation is $\hat{y} = 12.6 + 0.3\ X$

Q.18 What do you mean by a linear regression ? Which applications are best modeled by linear regression ?

☞ **[SPPU : March-19, In Sem, Marks 5]**

Ans. : • Best applications of linear regression are as follows :

1. If a company's sales have increased steadily every month for the past few years, by conducting a linear analysis on the sales data with monthly sales, the company could forecast sales in future months.
2. Linear regression can also be used to analyze the marketing effectiveness, pricing and promotions on sales of a product.
3. Linear regressions can be used in business to evaluate trends and make estimates or forecasts.

4. Supposing two campaigns are run on TV and Radio in parallel, a linear regression can capture the isolated as well as the combined impact of running this ads together.

Also Refer Q.12.

3.4 : Regression : Lasso Regression, Ridge Regression

Q.19 What do you mean by logistic regression ? Explain with example. [SPPU : June-22, Marks 8]

Ans. : • Logistic regression is a form of regression analysis in which the outcome variable is binary or dichotomous. A statistical method used to model dichotomous or binary outcomes using predictor variables.

- **Logistic component :** Instead of modeling the outcome, Y, directly, the method models the log odds (Y) using the logistic function.
- **Regression component :** Methods used to quantify association between an outcome and predictor variables. It could be used to build predictive models as a function of predictors.
- **In simple** logistic regression, logistic regression with 1 predictor variable.

Logistic Regression :

$$\ln\left(\frac{P(Y)}{1-P(Y)}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_k X_k$$

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_k X_k + \varepsilon$$

- With logistic regression, the response variable is an indicator of some characteristic, that is, a 0/1 variable. Logistic regression is used to determine whether other measurements are related to the presence of some characteristic, for example, whether certain blood measures are predictive of having a disease.
- If analysis of covariance can be said to be a t test adjusted for other variables, then logistic regression can be thought of as a chi-square test for homogeneity of proportions adjusted for other variables. While the response variable in a logistic regression is a 0/1 variable, the logistic

regression equation, which is a linear equation, does not predict the 0/1 variable itself.

- Fig. Q.19.1 shows sigmoid curve for logistic regression.

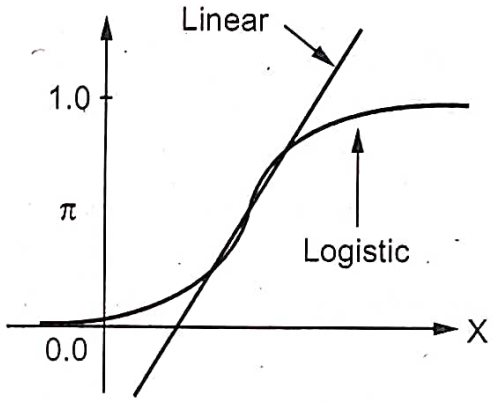


Fig. Q.19.1

- The linear and logistic probability models are :

Linear Regression :

$$p = a_0 + a_1X_1 + a_2X_2 + \ldots + a_kX_k$$

Logistic Regression :

$$\ln[p/(1-p)] = b_0 + b_1X_1 + b_2X_2 + \ldots + b_kX_k$$

- The linear model assumes that the probability p is a linear function of the regressors, while the logistic model assumes that the natural log of the odds $p/(1-p)$ is a linear function of the regressors.

- The major advantage of the linear model is its interpretability. In the linear model, if a 1 is 0.05, that means that a one-unit increase in X1 is associated with a 5 % point increase in the probability that Y is 1.

- The logistic model is less interpretable. In the logistic model, if b1 is 0.05, that means that a one - unit increase in X1 is associated with a 0.05 increase in the log odds that Y is 1. And what does that mean? I've never met anyone with any intuition for log odds.

Q.20 Explain in detail the ridge regression and the lasso regression.

☞ [SPPU : March-20, In Sem, Marks 10]

Ans. : • Ridge regression and the Lasso are two forms of regularized regression. These methods are seeking to improve the consequences of multicollinearity.

1. When variables are highly correlated, a large coefficient in one variable may be alleviated by a large coefficient in another variable, which is negatively correlated to the former.
2. Regularization imposes an upper threshold on the values taken by the coefficients, thereby producing a more parsimonious solution, and a set of coefficients with smaller variance.

Ridge

- Ridge estimation produces a biased estimator of the true parameter β.

$$
\begin{aligned}
E[\hat{\beta}^{ridge} | X] &= (X^T X + \lambda I)^{-1} X^T X \beta \\
&= (X^T X + \lambda I)^{-1} (X^T X + \lambda I - \lambda I)\beta \\
&= [I - \lambda (X^T X + \lambda I)^{-1}]\beta \\
&= \beta - \lambda (X^T X + \lambda I)^{-1} \beta
\end{aligned}
$$

- Ridge regression shrinks the regression coefficients by imposing a penalty on their size. The ridge coefficients minimize a penalized residual sum of squares.
- Ridge regression protects against the potentially high variance of gradients estimated in the short directions.

Lasso

- One significant problem of ridge regression is that the penalty term will never force any of the coefficients to be exactly zero. Thus, the final model will include all p predictors, which creates a challenge in model interpretation. A more modern machine learning alternative is the lasso.
- The lasso works in a similar way to ridge regression, except it uses a different penalty term that shrinks some of the coefficients exactly to zero.
- Lasso is a regularized regression machine learning technique that avoids over-fitting of training data and is useful for feature selection.

- The lasso is a shrinkage method like ridge, with subtle but important differences. The lasso estimate is defined by,

$$\hat{\beta}^{lasso} = \arg\min_{\beta} \sum_{i=0}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2$$

subject to $\sum_{j=1}^{p} |\beta_j| < t$

- Note the name "lasso" is actually an acronym for : **Least Absolute Selection and Shrinkage Operator.**
- The only difference from Ridge regression is that the regularization term is in absolute value. But this difference has a huge impact on the trade-off.
- Lasso method overcomes the disadvantage of Ridge regression by not only punishing high values of the coefficients β but actually setting them to zero if they are not relevant.

Q.21 Write short note on types of regression ?

☞ [SPPU : March-19, Marks 5]

Ans. : Types of regression are linear regression, logistic regression, ridge regression, lasso regression and polynomial regression. Also Refer Q.19 and Q.20

Q.22 What is difference between ridge regression and the lasso regression ?

Ans. :

Sr. No.	Ridge regression	Lasso regression
1.	Performs L2 regularization.	Performs L1 regularization.
2.	When many predictor variables are significant in the model and their coefficients are roughly equal then ridge regression tends to perform better because it keeps all of the predictors in the model.	In cases where only a small number of predictor variables are significant, lasso regression tends to perform better because it's able to shrink insignificant variables completely to zero and remove them from the model.

3.	Adds penalty equivalent to square of the magnitude of coefficients.	Adds penalty equivalent to absolute value of the magnitude of coefficients.
4.	The shrinkage factor is lambda times the sum of squares of regression coefficients.	The shrinkage factor is lambda times the sum of regression coefficients.

3.5 : Gradient Descent Algorithm

Q.23 Explain gradient descent algorithm. Also explain its limitation.

Ans. : • Gradient descent is a first-order optimization algorithm. To find a local minimum of a function using gradient descent, one takes steps proportional to the negative of the gradient of the function at the current point.

- Gradient descent is popular for very large-scale optimization problems because it is easy to implement, can handle black box functions, and each iteration is cheap.
- Given a differentiable scalar field f (x) and an initial guess x_1 , gradient descent iteratively moves the guess toward lower values of "f" by taking steps in the direction of the negative gradient – ∇ f (x).
- Locally, the negated gradient is the steepest descent direction, i.e., the direction that x would need to move in order to decrease "f" the fastest. The algorithm typically converges to a local minimum, but may rarely reach a saddle point, or not move at all if x_1 lies at a local maximum.
- The gradient will give the slope of the curve at that x and its direction will point to an increase in the function. So we change x in the opposite direction to lower the function value :

$$X_{k+1} = x_k - \lambda \nabla f(x_k)$$

- The $\lambda > 0$ is a small number that forces the algorithm to make small jumps

Limitations of gradient descent :

- Gradient descent is relatively slow close to the minimum : technically, its asymptotic rate of convergence is inferior to many other methods.
- For poorly conditioned convex problems, gradient descent increasingly 'zigzags' as the gradients point nearly orthogonally to the shortest direction to a minimum point

Q.24 Explain steepest descent method.

Ans. : • Steepest descent is also known as gradient method.

- This method is based on first order Taylor series approximation of objective function. This method is also called saddle point method. Fig. Q.24.1 shows steepest descent method.

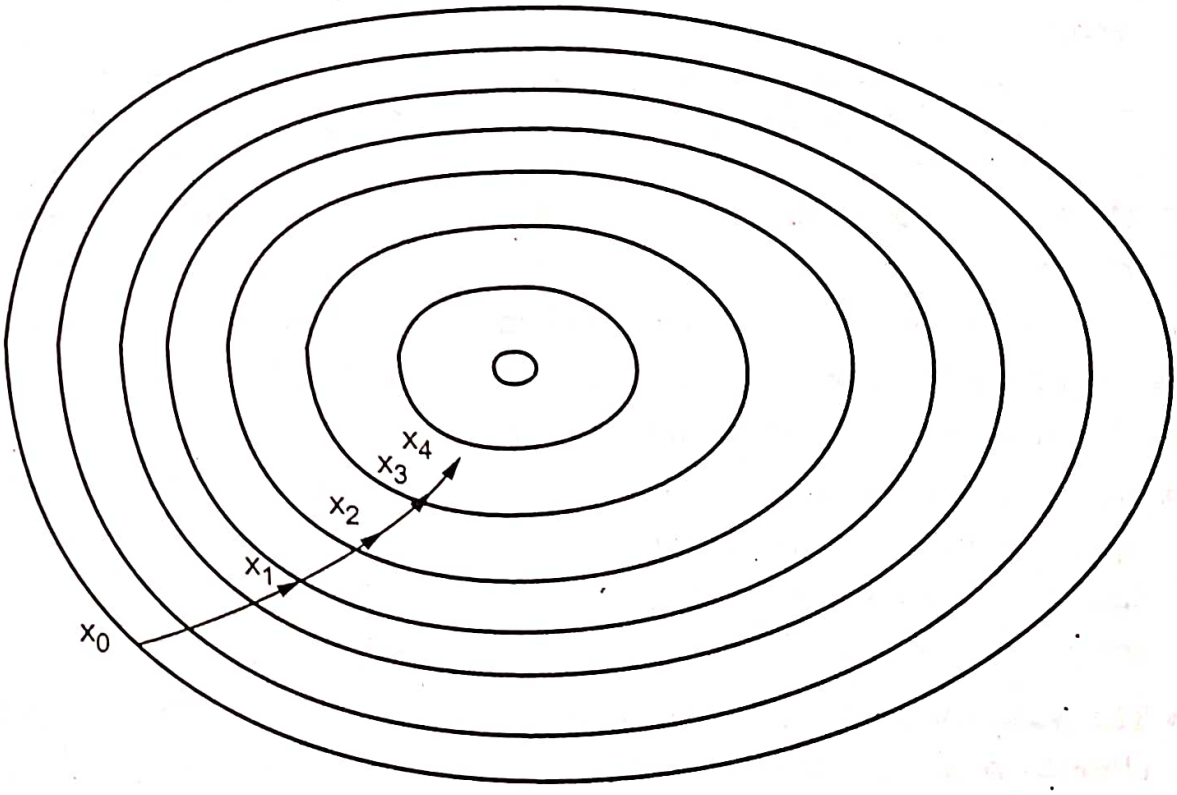


Fig. Q.24.1 Steepest descent method

- The steepest descent is the simplest of the gradient methods. The choice of direction is where f decreases most quickly, which is in the direction opposite to $\nabla f(x_i)$. The search starts at an arbitrary point x_0 and then go down the gradient, until reach close to the solution.

- The method of steepest descent is the discrete analogue of gradient descent, but the best move is computed using a local minimization rather than computing a gradient. It is typically able to converge in few steps but it is unable to escape local minima or plateaus in the objective function.
- The gradient is everywhere perpendicular to the contour lines. After each line minimization the new gradient is always orthogonal to the previous step direction. Consequently, the iterates tend to zig-zag down the valley in a very inefficient manner.
- The method of steepest descent is simple, easy to apply and each iteration is fast. It also very stable; if the minimum points exist, the method is guaranteed to locate them after at least an infinite number of iterations.

3.6 : Evaluation Metrics : MAE, RMSE, R2

Q.25 Define and explain Squared Error (SE) and Mean Squared Error (MSE) w.r.t regression.

Ans. : • The most common measurement of overall error is the sum of the squares of the errors, or SSE (sum of squared errors). The line with the smallest SSE is called the least-squares regression line.

- Mean Squared Error (MSE) is calculated by taking the average of the square of the difference between the original and predicted values of the data. It can also be called the quadratic cost function or sum of squared errors.
- The value of MSE is always positive or greater than zero. A value close to zero will represent better quality of the estimator/predictor. An MSE of zero (0) represents the fact that the predictor is a perfect predictor.

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{n}(\text{Actual values} - \text{Predicted values})^2$$

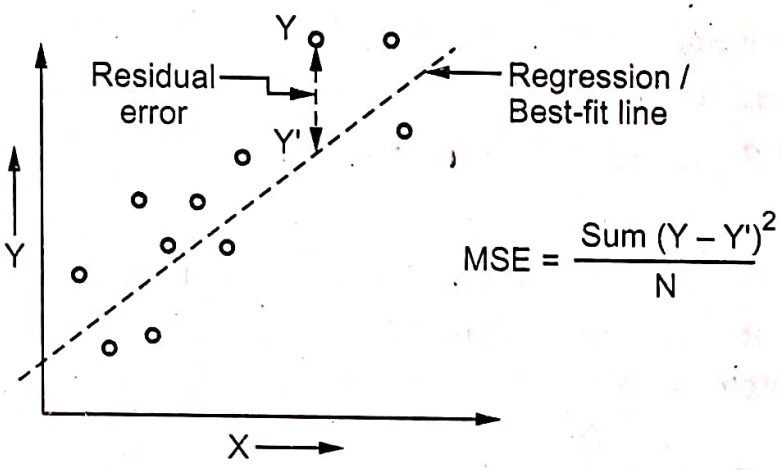


Fig. Q.25.1 Representation of MSE

- Here N is the total number of observations/rows in the dataset. The sigma symbol denotes that the difference between actual and predicted values taken on every i value ranging from 1 to n.
- Mean squared error is the most commonly used loss function for regression. MSE is sensitive towards outliers and given several examples with the same input feature values, the optimal prediction will be their mean target value. This should be compared with Mean Absolute Error, where the optimal prediction is the median. MSE is thus good to use if you believe that your target data, conditioned on the input, is normally distributed around a mean value, and when it's important to penalize outliers extra much.
- MSE incorporates both the variance and the bias of the predictor. MSE also gives more weight to larger differences. The bigger the error, the more it is penalized.
- Example : You want to predict future house prices. The price is a continuous value, and therefore we want to do regression. MSE can here be used as the loss function

Q.26 How the performance of a regression function is measured ?

Ans. : • Following are the performance metrics used for evaluating a regression model :

a) Mean Absolute Error (MAE)

b) Mean Squared Error (MSE)

c) Root Mean Squared Error (RMSE)

d) R-squared

e) Adjusted R-squared

1. MAE :

- MAE is the sum of absolute differences between our target and predicted variables. So it measures the average magnitude of errors in a set of predictions, without considering their directions.

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

2. Mean Squared Error (MSE)

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

3. Root Mean Square Error (RMSE)

- Root Mean Square Error (RMSE)is a standard way to measure the error of a model in predicting quantitative data.

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

4. R-squared

- R-squared is also known as the coefficient of determination. This metric gives an indication of how good a model fits a given dataset. It indicates how close the regression line is to the actual data values.

- The R squared value lies between 0 and 1 where 0 indicates that this model doesn't fit the given data and 1 indicates that the model fits perfectly to the dataset provided.

$$\text{R-squared} = 1 - \frac{\text{First sum of errors}}{\text{Second sum of errors}}$$

$$R^2 = 1 - \frac{SS_{RES}}{SS_{TOT}} = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}$$

5) Adjusted R-squared :

- The adjusted R-squared shows whether adding additional predictors improve a regression model or not.

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

Q.27 For a given data having 100 examples, if squared errors SE_1, SE_2, and SE_3 are 13.33, 3.33 and 4.00 respectively, calculate Mean Squared Error (MSE). State the formula for MSE.

Ans. :

$$\text{Mean Squared Error} = \frac{\text{Squared Error}_1 + \text{Squared Error}_2 + \ldots + \text{Squared Error}_N}{\text{Number of data samples}}$$

$$\text{Mean Squared Error} = \frac{13.33 + 3.33 + 4.00}{100} = 0.2066$$

END...

Unit IV

4 Supervised Learning : Classification

4.1 : Classification : K-Nearest Neighbour

Q.1 What are neighbors ? Why is it necessary to use nearest neighbor while classifying ?

Ans. : • To find a predefined number of training samples closest in distance to the new point, and predict the label from these.

- The number of samples can be a user-defined constant (k-nearest neighbor learning), or vary based on the local density of points (radius-based neighbor learning).
- The distance can, in general, be any metric measure : standard Euclidean distance is the most common choice.
- In the nearest neighbor algorithm, we classified a new data point by calculating its distance to all the existing data points, then assigning it the same label as the closest labeled data point.
- Despite its simplicity, nearest neighbors has been successful in a large number of classification and regression problems, including handwritten digits or satellite image scenes.
- Being a non-parametric method, it is often successful in classification situations where the decision boundary is very irregular.
- Neighbors-based classification is a type of instance-based learning or non-generalizing learning : it does not attempt to construct a general internal model, but simply stores instances of the training data.
- Classification is computed from a simple majority vote of the nearest neighbors of each point : a query point is assigned the data class which has the most representatives within the nearest neighbors of the point.

- The basic nearest neighbors classification uses uniform weights : that is, the value assigned to a query point is computed from a simple majority vote of the nearest neighbors.
- Under some circumstances, it is better to weight the neighbors such that nearer neighbors contribute more to the fit. This can be accomplished through the weights keyword.
- The default value, weights = 'uniform', assigns uniform weights to each neighbor. weights = 'distance' assigns weights proportional to the inverse of the distance from the query point.
- Alternatively, a user-defined function of the distance can be supplied which is used to compute the weights.

Q.2 Explain kNN algorithm with its advantages and disadvantages.

Ans. : • The k-nearest neighbor (kNN) is a classical classification method and requires no training effort, critically depends on the quality of the distance measures among examples.

- It is the simplest machine learning algorithms based on supervised learning technique. kNN algorithm assumes the similarity between the new data and available data and put the new data into the category that is most similar to the available categories.
- The k-nearest neighbour classification is one of the most popular distance-based algorithms. This classification is based on measuring the distances between the test sample and the training samples to determine the final classification output. The traditional k-NN classifier works naturally with numerical data. Fig. Q.2.1 shows kNN.
- kNN stores all available data and classifies a new point based on the similarity. This algorithm also used for regression as well as for Classification but mostly it is used for the classification problems.
- kNN is a non-parametric algorithm, which means it does not make any assumption on underlying data. It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.

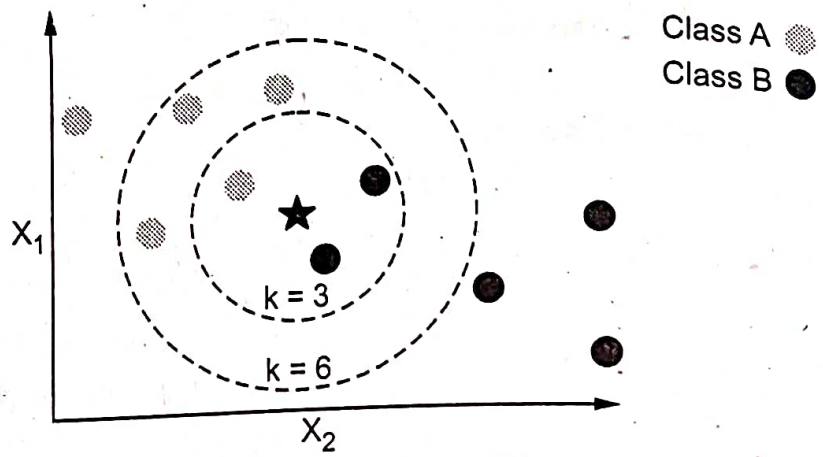


Fig. Q.2.1 kNN

- kNN, k is the number of nearest neighbors. The number of neighbors is the core deciding factor. k is generally an odd number if the number of classes is 2. When k = 1, then the algorithm is known as the nearest neighbor algorithm.
- k-NN algorithm gives user the flexibility to choose distance while building k-NN model.

  a) Euclidean distance b) Hamming distance

  c) Manhattan distance d) Minkowski distance

The performance of the kNN algorithm is influenced by three main factors :

1. The distance function or distance metric used to determine the nearest neighbors.
2. The decision rule used to derive a classification from the k-nearest neighbors.
3. The number of neighbors used to classify the new example.

Advantages :

1. The kNN algorithm is very easy to implement.
2. Nearly optimal in the large sample limit.
3. Uses local information, which can yield highly adaptive behavior.
4. Lends itself very easily to parallel implementations.

Disadvantages :

1. Large storage requirements.
2. Computationally intensive recall.
3. Highly susceptible to the curse of dimensionality.

Q.3 Let on a scale of 1 to 10 (where 1 is lowest and 10 is highest), a student is evaluated by internal examiner and external examiner and accordingly student result can be pass or fail. A sample data is collected for 4 students. If a new student is rated by internal and external examiner as 3 and 7 respectively (test instance) decide new student's result using KNN classifier.

Student No.	(x_{i1}) Rating by internal examiner	(x_{i2}) Rating by external examiner	(y) Result
S_1	7	7	Pass
S_2	7	4	Pass
S_3	3	4	Fail
S_4	1	4	Fail
Snew	3	7	?

Ans. :

x_i = Input vector of dimension 2 = $\{x_{i1}, x_{i2}\}$

where, x_{i1} = Rating by internal examiner

x_{i2} = Rating by external examiner

For i = 1, 2, 3, 4 (i.e. 4 number of sample instances)

a) y_i = Result of a student and $y_i \in$ {pass, fail}

b) $x_0 = (x_{01}, x_{02}) = (3, 7)$

Step 1 : Let K = 3 (because number of classes = 2 and K should be odd)

Step 2 : Calculation of Euclidean distance between x_0 and x_1, x_2, x_3, x_4 as

$$d_{i0} = \sqrt{\sum_{i=1}^{2} (x_{i1} - x_{01})^2}$$

where, d = Number of features in input vector = 2

$$d_{10} = \sqrt{\sum_{i=1}^{2}(x_{i1} - x_{01})^2}$$

$$= \sqrt{(x_{11} - x_{01})^2 + (x_{12} - x_{02})^2}$$

$$= \sqrt{(7-3)^2 + (7-7)^2} = 4$$

Similarly $d_{10} = 4$

$d_{20} = 5$

$d_{30} = 3$

$d_{40} = 3.60$

Step 3 : Arranging all above distances in non decreasing order.

$(d_{30}, d_{40}, d_{10}, d_{20}) = (3, 3.60, 4, 5)$

Step 4 : Select K = 3 distance from above as (d_{30}, d_{40}, d_{60})

(3, 3.60, 4, 5)

Step 5 : Decide instances corresponding to 3-nearest instances. For given test instance 3, nearest neighbors are 3rd student, 4th student, 1st student, i.e. (3, 4, fail), (1, 4, fail), (7, 7, pass).

Step 6 : Decide K_{pass} and K_{fail}

$K_{pass} = 1$

$K_{fail} = 2$

$K_{fail} > K_{pass}$

Step 7 : New student or test instance x_0 is classified to "fail" because K_{fail} is maximum.

4.2 : Support Vector Machine

Q.4 What do you mean by SVM ? Explain with example.

☞ [SPPU : May-22, Marks 8]

Ans. : • Support Vector Machines (SVMs) are a set of supervised learning methods which learn from the dataset and used for classification.

- Support vector machines are supervised machine learning algorithms, and they are used for classification and regression analysis. The SVM performs both linear classification and nonlinear classification.
- The nonlinear classification is performed using the kernel function. In nonlinear classification, the kernels are homogenous polynomial, complex polynomial, Gaussian radial basis function, and hyperbolic tangent function.
- SVM finds a hyperplane to separate the inputs into separate groups. There can be many hyperplanes that successfully divide the input vectors. To optimize the solution and find the optimum hyperplane, support vectors are used.
- The points closest to the hyperplane are known as support vectors. In SVM, the hyperplane having maximal distance from support vectors is chosen as the output hyperplane. The hyperplane and support vectors are shown in Fig. Q.4.1.

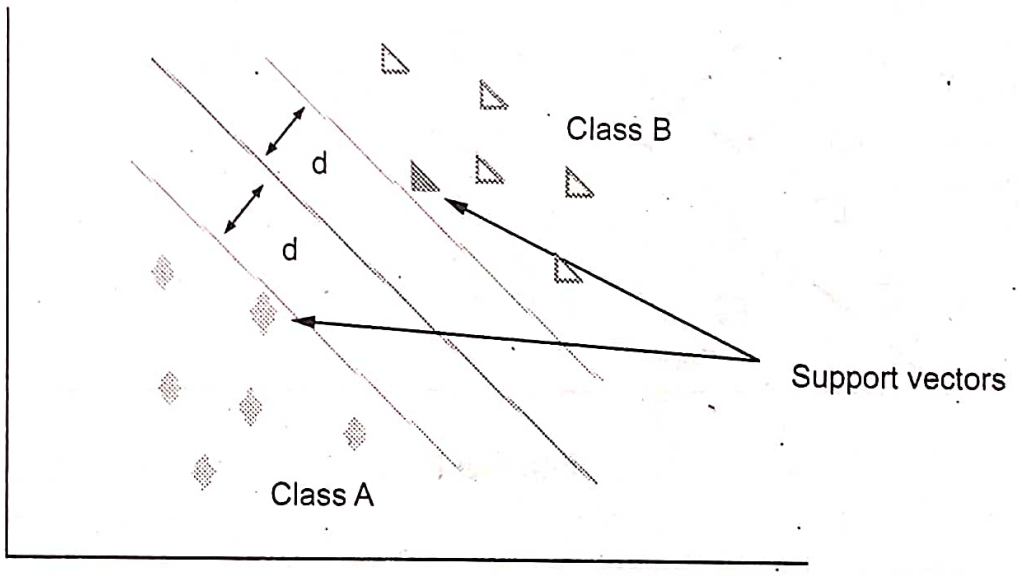


Fig. Q.4.1

- So as it's a far 2-d area so by using just using a straight line, we are able to without difficulty separate those instructions. But there may be multiple lines that may separate those lessons. Consider the beneath picture :

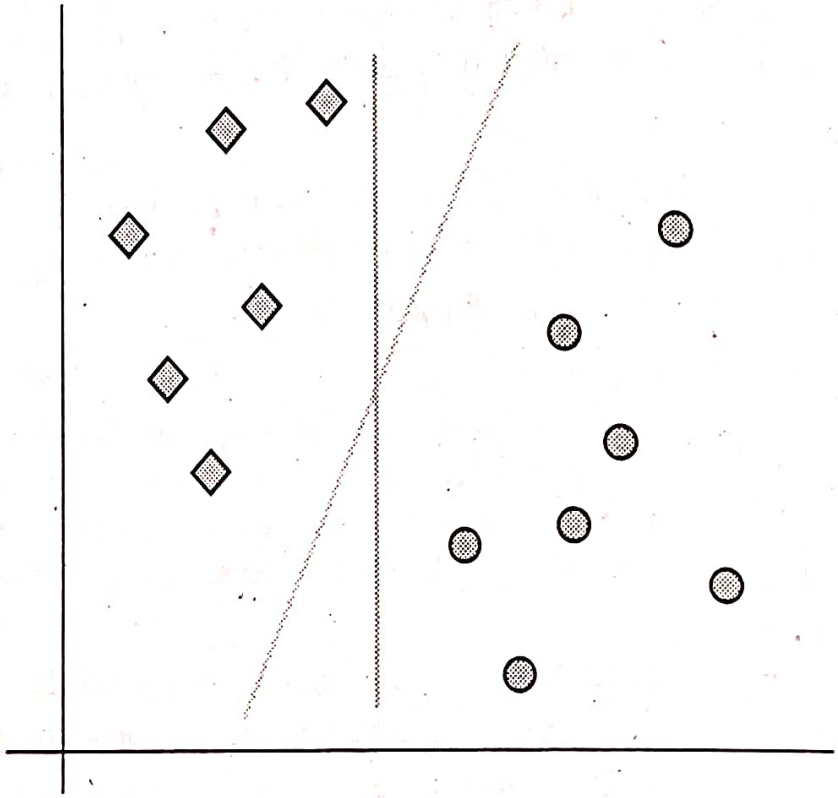


Fig. Q.5.2 Linear SVM understanding

- Hence, the SVM algorithm helps to discover the first-rate line or selection boundary; this exceptional boundary or region is known as a hyperplane. SVM algorithm unearths the nearest point of the traces from each of the lessons. These points are referred to as guide vectors. The distance between the vectors and the hyperplane is called the margin. And the purpose of SVM is to maximise this margin. The hyperplane with maximum margin is known as the most suitable hyperplane.

Q.6 Explain non linear SVM with examples. ☞ [SPPU : May-19, Marks 4]

Ans. : • Non-linear SVM : Non-linear SVM is used for non-linearly separated facts, because of this if a dataset can't be labelled through the usage of a directly line, then such facts is called as non-linear information and classifier used is known as non-linear SVM classifier.

- If information is linearly arranged, then we can separate it through using a directly line, however for non-linear information, we can not draw an unmarried directly line. Consider the beneath picture :

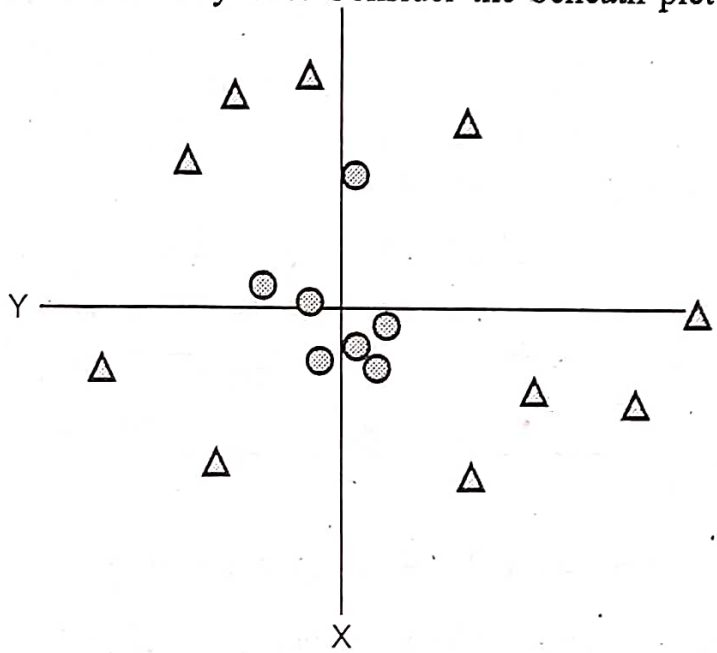


Fig. Q.6.1 Non linear SVM

- So to separate these data points, we need to feature one greater size. For linear information, we've used dimensions x and y, so for non-linear information, we will upload a 3rd dimension z. It can be calculated as :

$$z = x_2 + y_2$$

- By adding the third measurement, the sample area will become as below photograph :

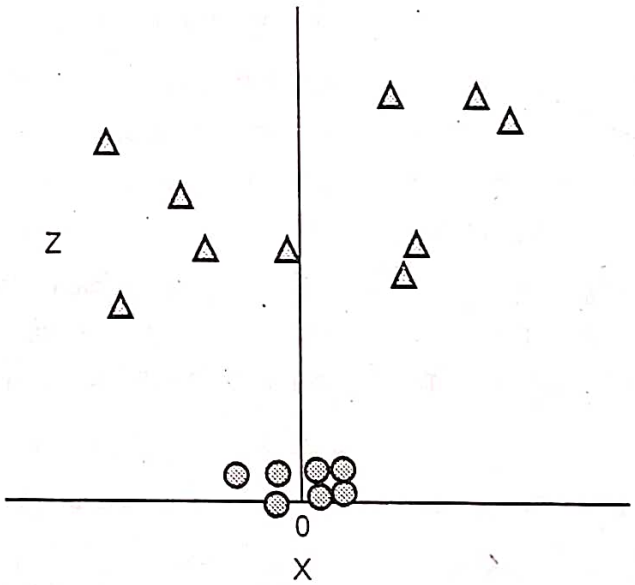


Fig. Q.6.2 Non linear SVM with third measurement

- So now, SVM will divide the datasets into instructions within the following way. Consider the under photo :

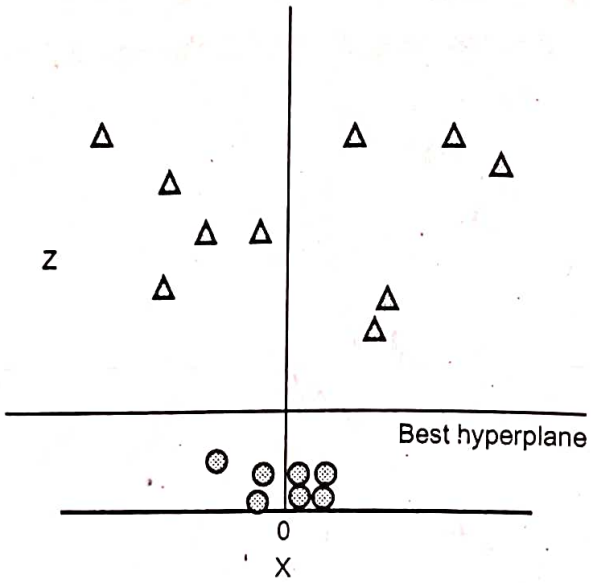


Fig. Q.6.3 Datasets representation

Q.7 Explain key properties of support vector machine.

Ans. :

1. Use a single hyperplane which subdivides the space into two half-spaces, one which is occupied by Class 1 and the other by Class 2
2. They maximize the margin of the decision boundary using quadratic optimization techniques which find the optimal hyperplane.
3. Ability to handle large feature spaces.
4. Overfitting can be controlled by soft margin approach
5. When used in practice, SVM approaches frequently map the examples to a higher dimensional space and find margin maximal hyperplanes in the mapped space, obtaining decision boundaries which are not hyperplanes in the original space.
6. The most popular versions of SVMs use non-linear kernel functions and map the attribute space into a higher dimensional space to facilitate finding "good" linear decision boundaries in the modified space.

Q.8 Explain applications and limitations of SVM.

Ans. : SVM applications

- SVM has been used successfully in many real-world problems,
  1. Text (and hypertext) categorization
  2. Image classification
  3. Bioinformatics (Protein classification, Cancer classification)
  4. Hand-written character recognition
  5. Determination of SPAM email.

Limitations of SVM

1. It is sensitive to noise.
2. The biggest limitation of SVM lies in the choice of the kernel.
3. Another limitation is speed and size.
4. The optimal design for multiclass SVM classifiers is also a research area.

4.3 : Ensemble Learning : Bagging, Boosting, Random Forest, Adaboost

Q.9 Explain ensemble learning.

Ans. : • The idea of ensemble learning is to employ multiple learners and combine their predictions. If we have a committee of M models with uncorrelated errors, simply by averaging them the average error of a model can be reduced by a factor of M.

- Unfortunately, the key assumption that the errors due to the individual models are uncorrelated is unrealistic; in practice, the errors are typically highly correlated, so the reduction in overall error is generally small.

- Ensemble modeling is the process of running two or more related but different analytical models and then synthesizing the results into a single score or spread in order to improve the accuracy of predictive analytics and data mining applications.

- Ensemble of classifiers is a set of classifiers whose individual decisions combined in some way to classify new examples.

- Ensemble methods combine several decision trees classifiers to produce better predictive performance than a single decision tree classifier. The main principle behind the ensemble model is that a group of weak learners come together to form a strong learner, thus increasing the accuracy of the model.
- There are two approaches for combining models : **voting and stacking.**
- In **voting**, no learning takes place at the meta level when combining classifiers by a voting scheme. Label that is most often assigned to a particular instance is chosen as the correct prediction when using voting.
- **Stacking** is concerned with combining multiple classifiers generated by different learning algorithms $L_1, \ldots, L_N$ on a single dataset S, which is composed by a feature vector $S_i = (x_i, t_i)$.
- The stacking process can be broken into two phases :
  1. Generate a set of base-level classifiers $C_1, \ldots, C_N$ Where $C_i = L_i(S)$
  2. Train a meta-level classifier to combine the outputs of the base-level classifiers
- Fig. Q.9.1 shows stacking frame.
- The training set for the meta-level classifier is generated through a leave-one-out cross validation process.

$$\forall i = 1, \ldots n \text{ and } \forall k = 1, \ldots, N : C_k^i$$

$$= L_k(S - s_i)$$

- The learned classifiers are then used to generate predictions for $s_i : \hat{y}_i^k = C_k^i(x_i)$
- The meta-level dataset consists of examples of the form $((\hat{y}_i^1, \ldots, \hat{y}_i^n), y_i)$, where the features are the predictions of the base-level classifiers and the class is the correct class of the example in hand.
- Why do ensemble methods work ?

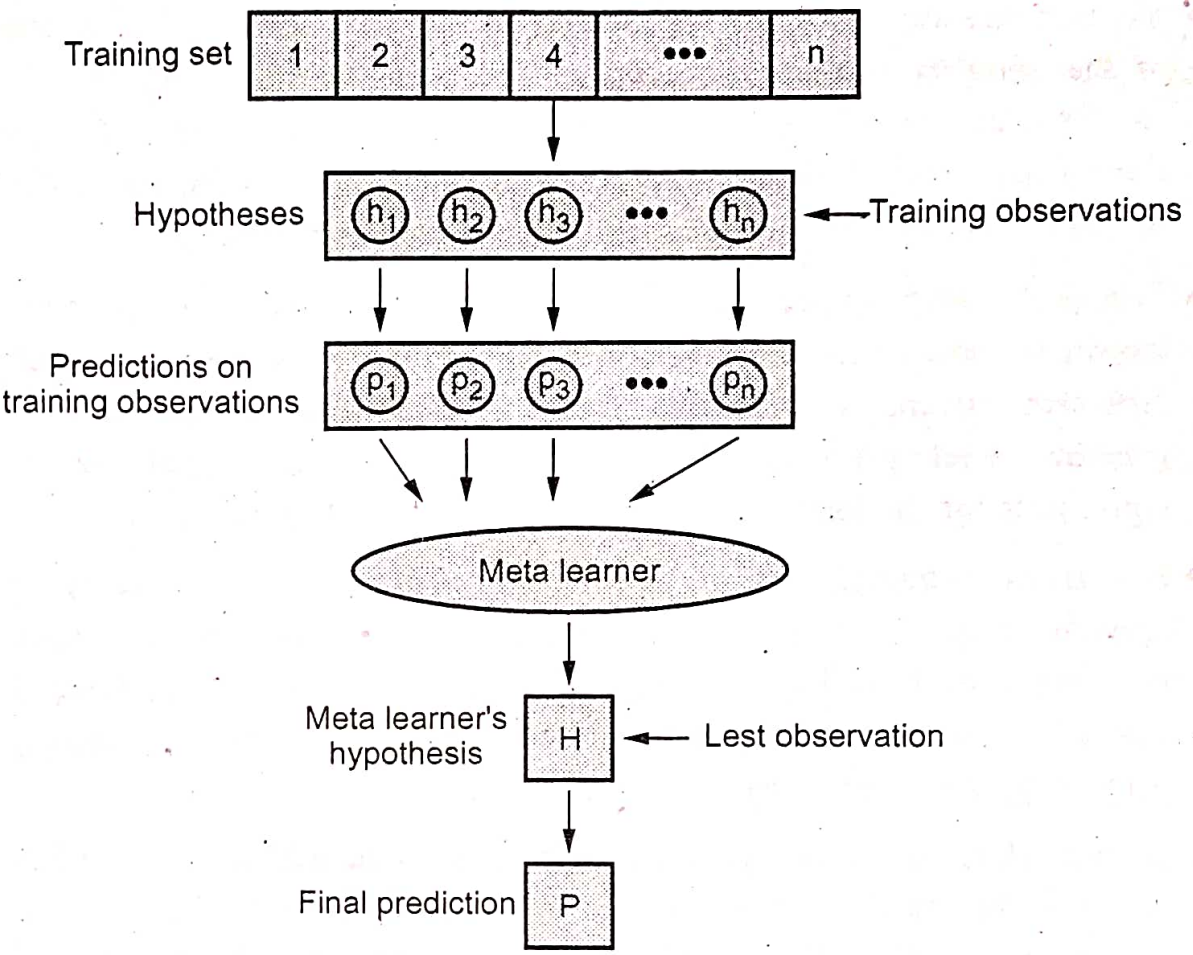


Fig. Q.9.1 Stacking frame

- Based on one of two basic observations :
  1. Variance reduction : If the training sets are completely independent, it will always helps to average an ensemble because this will reduce variance without affecting bias (e.g. bagging) and reduce sensitivity to individual data points.
  2. Bias reduction : For simple models, average of models has much greater capacity than single model. Averaging models can reduce bias substantially by increasing capacity and control variance by citting one component at a time.

Q.10 What is bagging ? Explain bagging steps. List its advantages and disadvantages.

Ans. : • Bagging is also called Bootstrap aggregating. Bagging and boosting are meta-algorithms that pool decisions from multiple classifiers. It creates ensembles by repeatedly randomly resampling the training data.

- Bagging was the first effective method of ensemble learning and is one of the simplest methods of arching. The meta-algorithm, which is a special case of the model averaging, was originally designed for classification and is usually applied to decision tree models, but it can be used with any type of model for classification or regression.
- Ensemble classifiers such as bagging, boosting and model averaging are known to have improved accuracy and robustness over a single model. Although unsupervised models, such as clustering, do not directly generate label prediction for each individual, they provide useful constraints for the joint prediction of a set of related objects.
- For given a training set of size n, create m samples of size n by drawing n examples from the original data, with replacement. Each *bootstrap sample* will on average contain 63.2 % of the unique training examples, the rest are replicates. It combines the m resulting models using simple majority vote.
- In particular, on each round, the base learner is trained on what is often called a "bootstrap replicate" of the original training set. Suppose the training set consists of n examples. Then a bootstrap replicate is a new training set that also consists of n examples, and which is formed by repeatedly selecting uniformly at random and with replacement n examples from the original training set. This means that the same example may appear multiple times in the bootstrap replicate, or it may appear not at all.
- It also decreases error by decreasing the variance in the results due to *unstable learners,* algorithms (like decision trees) whose output can change dramatically when the training data is slightly changed.
- **Pseudocode :**

1. Given training data $(x_1, y_1), \ldots, (x_m, y_m)$
2. For t = 1,..., T :
   a. Form bootstrap replicate dataset S_t by selecting m random examples from the training set with replacement.
   b. Let h_t be the result of training base learning algorithm on S_t.

3. Output combined classifier :
   $H(x) = \text{majority}\ (h_1(x), \ldots, h_T(x))$

Bagging steps :

1. Suppose there are N observations and M features in training data set. A sample from training data set is taken randomly with replacement.
2. A subset of M features is selected randomly and whichever feature gives the best split is used to split the node iteratively.
3. The tree is grown to the largest.
4. Above steps are repeated n times and prediction is given based on the aggregation of predictions from n number of trees.

Advantages of bagging :

1. Reduces over-fitting of the model.
2. Handles higher dimensionality data very well.
3. Maintains accuracy for missing data.

Disadvantages of bagging :

1. Since final prediction is based on the mean predictions from subset trees, it won't give precise values for the classification and regression model.

Q.11 Explain boosting steps. List advantages and disadvantages of boosting.

Ans. : Boosting steps :

1. Draw a random subset of training samples d_1 without replacement from the training set D to train a weak learner C_1.
2. Draw second random training subset d_2 without replacement from the training set and add 50 percent of the samples that were previously falsely classified/misclassified to train a weak learner C_1.
3. Find the training samples d_3 in the training set D on which C_1 and C_1 disagree to train a third weak learner C_3.
4. Combine all the weak learners via majority voting.

Advantages of boosting :

1. Supports different loss function.
2. Works well with interactions.

Disadvantages of boosting :

1. Prone to over-fitting.
2. Requires careful tuning of different hyper-parameters.

Q.12 What is random forest ? Explain working of random forest.

Ans. : • Ramdom forest is supervised learning algorithm. The "forest" it builds, is an ensemble of decision trees, usually trained with the "bagging" method. The general idea of the bagging method is that a combination of learning models increases the overall result.

- Random forest is a widely used classification and regression algorithm.
- Fig. Q.12.1 shows random forest algorithm.

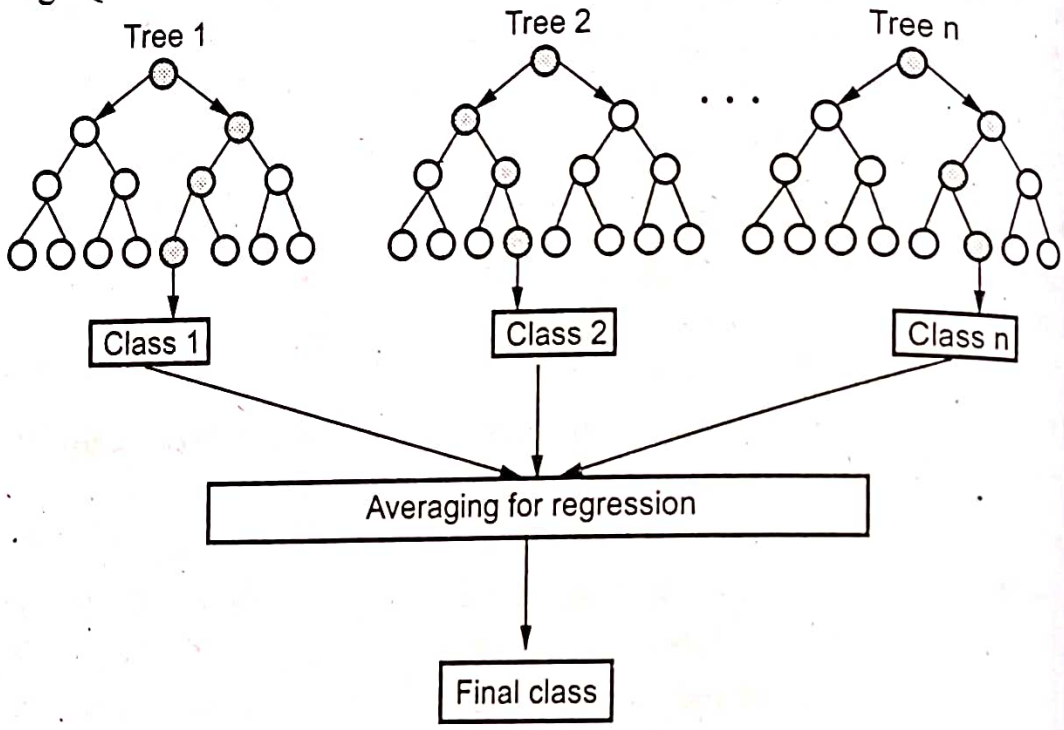


Fig. Q.12.1

- Random forest is a classifier that contains several decision trees on various subsets of a given dataset and takes the average to enhance the predicted accuracy of that dataset. Instead of relying on a single decision tree, the random forest collects the result from each tree and expects the final output based on the majority votes of predictions.
- Steps involved in random forest algorithm :

Step 1 : In random forest n number of random records are taken from the data set having k number of records.

Step 2 : Individual decision trees are constructed for each sample.

Step 3 : Each decision tree will generate an output.

Step 4 : Final output is considered based on majority voting or averaging for classification and regression respectively.

Q.13 Discuss application, advantages and disadvantages of random forest algorithm.

Ans. : • **Application of random forest :**

1. Banking : It is mainly used in the banking industry to identify loan risk.
2. Medicine : To identify illness trends and risks.
3. Land use : Random forest classifier is also used to classify places with similar land-use patterns.
4. Market trends : You can determine market trends using this algorithm.

• **Advantages :**

1. Random forest can be used to solve both classification as well as regression problems.
2. Random forest works well with both categorical and continuous variables.
3. Random forest can automatically handle missing values.
4. It is not affected by the dimensionality curse.

• **Disadvantages :**

1. Due to its complexities, training time is longer than for other models.
2. Not suitable for real-time predictions.
3. More trees slow down model.
4. Can't describe relationships within data.

Q.14 Write short note on :

i) Bagging ii) Boosting iii) Random forest. ☞ **[SPPU : June-22, Marks 8]**

Ans. : Refer Q.10, Q.11 and Q.12.

Q.15 Write short note on Adaboost.

Ans. : • AdaBoost stands for Adaptive Boosting. Adaboost is also an ensemble learning algorithm that is created using a bunch of what is

called a decision stump. AdaBoost uses an iterative approach to learn from the mistakes of weak classifiers and turn them into strong ones. AdaBoost is an iterative ensemble method. AdaBoost classifier builds a strong classifier by combining multiple poorly performing classifiers.

- AdaBoost is called adaptive because it uses multiple iterations to generate a single composite strong learner. During each round of training, a new weak learner is added to the ensemble and a weighting vector is adjusted to focus on examples that were misclassified in previous rounds.
- Steps for performing the AdaBoost algorithm :
  1. Initially, all observations are given equal weights.
  2. A model is built on a subset of data.
  3. Using this model, predictions are made on the whole dataset.
  4. Errors are calculated by comparing the predictions and actual values.
  5. While creating the next model, higher weights are given to the data points which were predicted incorrectly.
  6. Weights can be determined using the error value. For instance, the higher the error the more is the weight assigned to the observation.
  7. This process is repeated until the error function does not change, or the maximum limit of the number of estimators is reached.

4.4 : Binary-vs-Multiclass Classification

Q.16 What is binary classification ? Explain with example.

Ans. : • If the output is a binary value; (yes/no), (+/–), (0/1), then the learning problem is referred to as a binary classification problem. Example : learning to distinguish sport cars.

- Binary classification would generally fall into the domain of supervised learning since the training dataset is labeled. Discriminant function is used to represent a classifier.
- A well known example of a binary classification problem is predicting whether an email is spam or not spam.

- In binary classification, the objective is to design a rule that assigns objects to one of 2 classes, often referred to as positive and negative, on the basis of descriptive vectors of objects.
- **Example of classification :** The ABC bank wants to classify its customers based on whether they are expected to pay back their approved home loans. For this purpose, the history of past customers is used to train the classifier. The classifier provides rules, which identify potentially reliable future customers. The classification rule is as follows :

  a. ‡If age = "31...40"and income = low then credit-rating = bad

  b. ‡If age = "31...40"and income = high then credit-rating = very good

Q.17 Explain linear classification model.

Ans. : A classification algorithm (Classifier) that makes its classification based on a linear predictor function combining a set of weights with the feature vector.

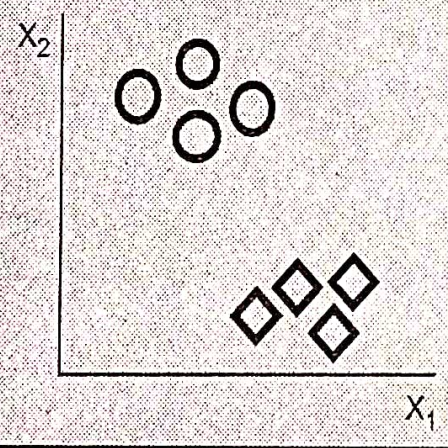
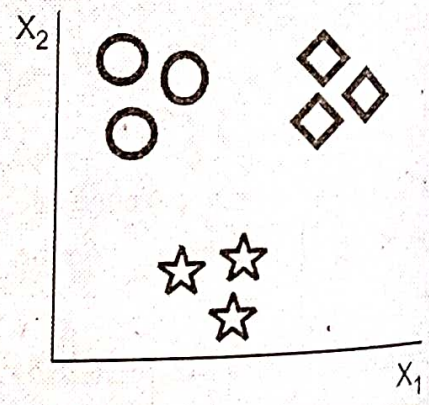
- A linear classifier does classification decision based on the value of a linear combination of the characteristics. Imagine that the linear classifier will merge into it's weights all the characteristics that define a particular class.
- Linear classifiers can represent a lot of things, but they can't represent everything. The classic example of what they can't represent is the XOR function.

Q.18 Define multiclass classification.

Ans. : Multiclass classification is a machine learning classification task that consists of more than two classes, or outputs. For example, using a model to identify animal types in images from an encyclopedia is a multiclass classification example because there are many different animal classifications that each image can be classified as.

Q.19 Explain difference between Binary-vs-Multiclass classifications.

Ans. :

Binary classification	Multiclass classification
Binary classification are those tasks where examples are assigned exactly one of two classes.	Multi-class classification is those tasks where examples are assigned exactly one of more than two classes
Binary classification can be used for spam detection and fraud detection.	Multiclass classification is often used in image recognition and document classification tasks.
Algorithm used in binary classifications are Logistic regression, kNN, Decision Trees, SVM.	Algorithm used in multi-class classification are Naive Bayes, Random forest and Gradient boosting.
Binary classification is a supervised machine learning algorithm that is used to predict one of two classes for an item.	Multiclass classification is a supervised machine learning algorithm used to predict one or more classes for an item.
	

Q.20 Explain balanced and imbalanced multiclass classification problems.

Ans. : Balanced classification :

- When the usage of a device gains knowledge of a set of rules, it's miles very crucial to teach the model on a dataset with nearly the same number of samples. This is referred to as a balanced elegance. We want

to have balanced training to train a version, however if the training isn't balanced, we need to use a class balancing method before using a device gaining knowledge of the set of rules.

Balanced classes

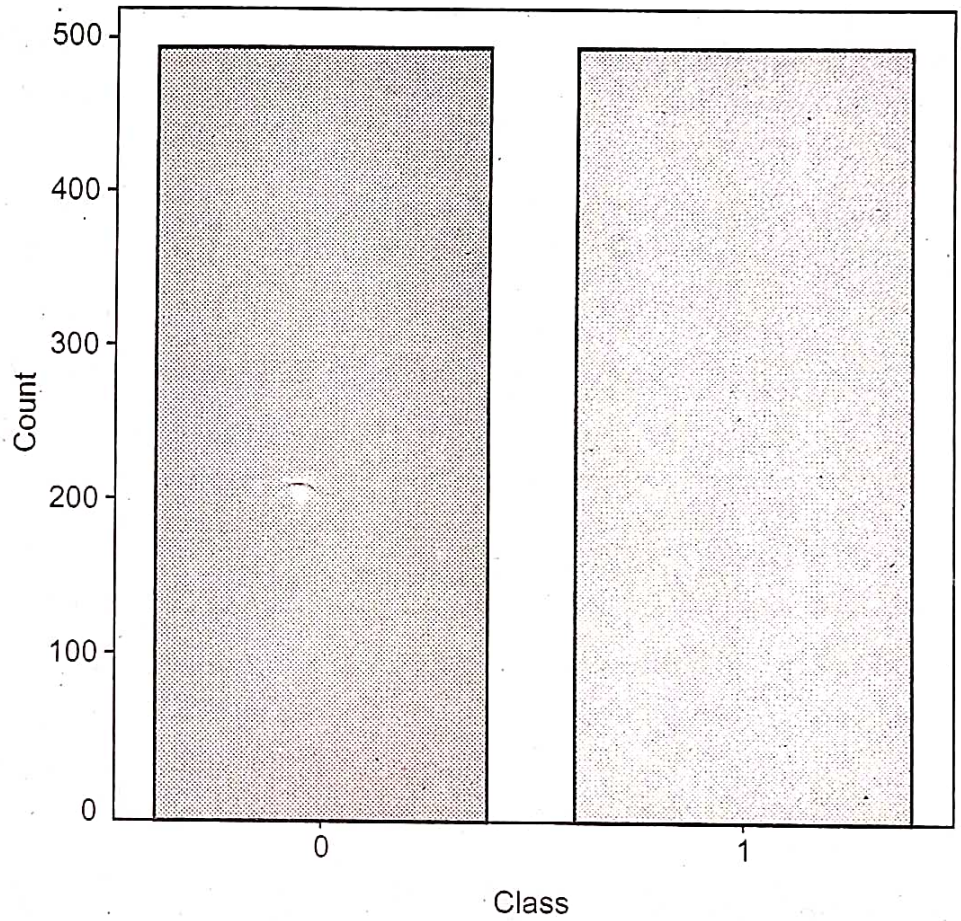


Fig. Q.20.1 Balanced classification

Imbalanced classification :

- In imbalanced type trouble is an instance of a type trouble in which the distribution of examples across the recognised classes is biassed or skewed. The distribution can vary from a moderate bias to an excessive imbalance where there is one instance in the minority class for loads, heaps or thousands and thousands of examples in the majority magnificence or instructions.
- Imbalanced classifications pose a mission for predictive modelling as most of the device mastering algorithms used for category were designed across the assumption of the same variety of examples for every magnificence. These outcomes in fashions that have negative

predictive overall performance, specially for the minority elegance. This is a problem due to the fact usually, the minority class is more important and therefore the problem is more sensitive to category errors for the minority class than most people's magnificence.

Imbalanced class distribution

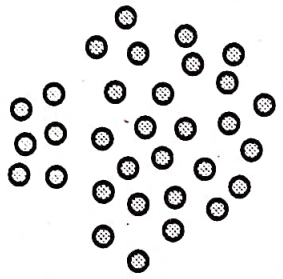


Fig. Q.20.2 Imbalanced classification

4.5 : Variants of Multiclass Classification : One-vs-One and One-vs-All

Q.21 Explain various multiclass classification techniques.

Ans. : 1. One Vs All (OVA) :

- For each class build a classifier for that class vs the rest. Build N different binary classifiers.
- For this approach, we require N = K binary classifiers, where the Kth classifier is trained with positive examples belonging to class K and negative examples belonging to the other K – 1 classes.
- When testing an unknown example, the classifier producing the maximum output is considered the winner, and this class label is assigned to that example.
- It is simple and provides performance that is comparable to other more complicated approaches when the binary classifier is tuned well.

2. All-Vs-All (AVA) :

- For each class build a classifier for those class vs the rest. Build N (N – 1) classifiers, one classifier to distinguish each pair of classes i and j .

- A binary classifier is built to discriminate between each pair of classes, while discarding the rest of the classes.
- When testing a new example, a voting is performed among the classifiers and the class with the maximum number of votes wins.

3. Calibration

- The decision function f of a classifier is said to be calibrated or well-calibrated if P (x is correctly classified | f(x) = s) ≃ s
- Informally f is a good estimate of the probability of classifying correctly a new datapoint x which would have output value x. Intuitively if the "raw" output of a classifier is g you can calibrate it by estimating the probability of x being well classified given that g(x)=y for all y values possible.

4. Error-Correcting Output-Coding (ECOC)

- Error correcting code approaches try to combine binary classifiers in a way that lets you exploit de-correlations and correct errors.
- This approach works by training N binary classifiers to distinguish between the K different classes. Each class is given a codeword of length N according to a binary matrix M . Each row of M corresponds to a certain class.
- The following table shows an example for K = 5 classes and N = 7 bit code words.

	f_1	f_2	f_3	f_4	f_5	f_6	f_7
Class 1	0	0	0	0	0	0	0
Class 2	0	1	1	0	0	1	1
Class 3	0	1	1	1	1	0	0
Class 4	1	0	1	1	0	1	0
Class 5	1	1	0	1	0	0	1

- Each class is given a row of the matrix. Each column is used to train a distinct binary classifier. When testing an unseen example, the output codeword from the N classifiers is compared to the given K code

words, and the one with the minimum hamming distance is considered the class label for that example.

4.6 : Evaluation Metrics and Score

Q.22 What is mean average precision ? Explain precision recall appropriateness.

Ans. : • **Mean Average Precision (MAP)** is also called average precision at seen relevant documents. It determine precision at each point when a new relevant document gets retrieved. Average of the precision value obtained for the top k documents, each time a relevant doc is retrieved.

- Avoids interpolation, use of fixed recall levels. MAP for query collection is arithmetic averaging. Average precision - recall curves are normally used to compare the performance of distinct IR algorithms.
- Use P = 0 for each relevant document that was not retrieved. Determine average for each query, then average over queries :

$$MAP = \frac{1}{N} \sum_{j=1}^{N} \frac{1}{Q_j} \sum_{i=1}^{Q_j} P(doc_i)$$

where Q_j = Number of relevant document for query j

N = Number of queries

$P(doc_i)$ = Precision at i^{th} relevant document

Precision - recall appropriateness :

- Precision and recall have been extensively used to evaluate the retrieval performance of IR algorithms. However, a more careful reflection reveals problems with these two measures :
- First, the proper estimation of maximum recall for a query requires detailed knowledge of all the documents in the collection.
- Second, in many situations the use of a single measure could be more appropriate.
- Third, recall and precision measure the effectiveness over a set of queries processed in batch mode.

- Fourth, for systems which require a weak ordering though, recall and precision might be inadequate.

Q.23 Write short note on F-measure.

Ans. : • The F measure is a measure of a test's accuracy and is defined as the weighted harmonic mean of the precision and recall of the test. The F - measure or F - score is one of the most commonly used "single number" measures in Information Retrieval, Natural Language Processing and Machine Learning.

- F-measure comes from Information Retrieval (IR) where Recall is the frequency with which relevant documents are retrieved or 'recalled' by a system, but it is known elsewhere as Sensitivity or True Positive Rate (TPR).
- Precision is the frequency with which retrieved documents or predictions are relevant or 'correct', and is properly a form of Accuracy, also known as Positive Predictive Value (PPV) or True Positive Accuracy (TPA). F is intended to combine these into a single measure of search 'effectiveness'.
- High precision and low accuracy is possible due to systematic bias. One of the problems with Recall, Precision, F - measure and Accuracy as used in Information Retrieval is that they are easily biased.
- The F-measure balances the precision and recall. The result is a value between 0.0 for the worst F-measure and 1.0 for a perfect F - measure.
- The formula for the standard F1 - score is the harmonic mean of the precision and recall. A perfect model has an F-score of 1.

$$\text{F - Measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Q.24 State formulae for calculating accuracy, true positive rate, true negative rate, false positive rate and false negative rate for binary classification tasks.

Ans. : The evaluation measure most used in practice is the accuracy rate. It evaluates the effectiveness of the classifier by its percentage of correct predictions.

Accuracy rate =

$$\frac{|\text{True negatives}| + |\text{True positives}|}{|\text{False negatives}| + \text{False positives}| + |\text{True negatives}| + |\text{True positives}|}$$

- True Positive Rate (TPR) is also called sensitivity, hit rate and recall.

$$\text{Sensitivity} = \frac{\text{Number of true positives}}{\text{Number of True positive} + \text{Number of False positive}}$$

- The true negative rate is also called specificity, which is the probability that an actual negative will test negative.

$$\text{Specificity} = \frac{|\text{True negative}|}{|\text{False positive}| + |\text{True positive}|}$$

- The **false negative rate** is also called the miss rate. It is the probability that a true positive will be missed by the test.

$$\text{Miss rate} = \frac{\text{False negative}}{\text{False negative} + \text{True positive}}$$

Summary :

Name	Formula
The Positive Rate (TP rate)	TP / (TP + FP)
True Negative Rate (TN rate)	TN / (TN + FN)
False Positive Rate (FP rate)	FP / (FP + TP)
False Negative Rate (FN rate)	FN / (FN + TP)

Q.25 What is a contingency table ? What does it represent ?

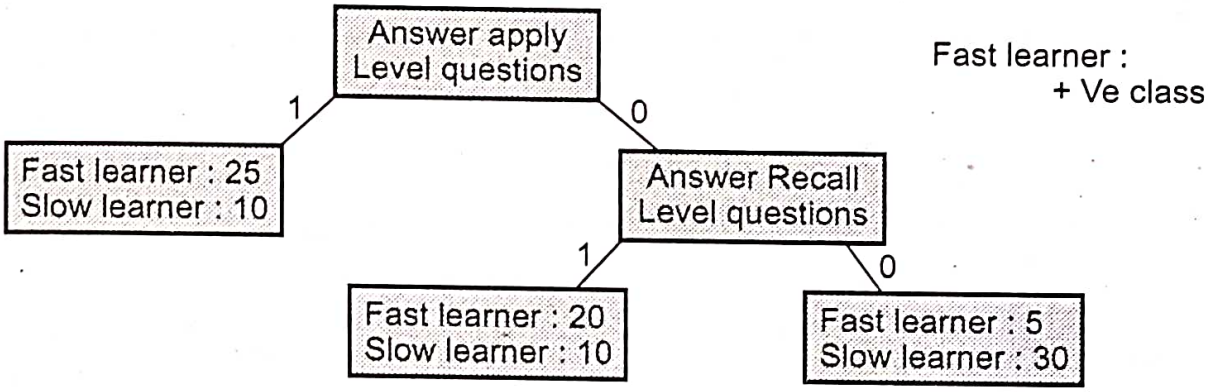
Ans. : • Contingency tables are used to analyze the relation between two or more categorical variables. A contingency table displays the frequency of specific combinations. They are used to summarize the relationships and not to evaluate the model.

• A contingency table corresponds to a probability mass function.

- A contingency table, sometimes called a two-way frequency table, is a tabular mechanism with at least two rows and two columns used to present categorical data in terms of frequency counts.
- An r X c contingency table shows the observed frequency of two variables, the observed frequencies of which are arranged into r rows and c columns. The intersection of a row and a column of a contingency table is called a cell.
- Contingency tables is represented as follows :

True class	Predicted class	
	Positive	Negative
Positive	True positive	False negative
Negative	False positive	True negative

Q.26 i) Find contingency table ii) Find recall iii) Precision iv) Negative recall v) False positive rate



Ans. : Contingency table

	Predicted			Total
Faster learner	25	20	5	50
Slow learner	10	10	30	50
Total	35	30	35	100

Actual

$$\text{Precision} = \frac{\text{True positive}}{\text{Actual results}} \quad \text{or} \quad \frac{\text{True positive}}{\text{True positive} + \text{False positive}}$$

$$\text{Recall} = \frac{\text{True positive}}{\text{Predicted results}} \quad \text{or} \quad \frac{\text{True positive}}{\text{True positive} + \text{False negative}}$$

Calculate precision and recall

Precision = 25/35 = 0.714

Recall = 25/30 = 0.833

False positive rate = (False positive) / (False positive + True negative)

= 10/(10 + 30) = 0.25

Q.27 Consider the following 3-class confusion matrix. Calculate precision and recall per class. Also calculate weighted average precision and recall for classifier.

	Predicted		
Actual	15	2	3
	7	15	8
	2	3	45

Ans. :

	Predicted			
Actual	15	2	3	20
	7	15	8	30
	2	3	45	50
	24	20	56	100

$$\text{Classifier Accuracy} = \frac{15+15+45}{100} = \frac{75}{100} = 0.75$$

Calculate per-class precision and recall :

First class $= \frac{15}{24} = 0.63$ and $\frac{15}{20} = 0.75$

Second class $= \frac{15}{20} = 0.75$ and $\frac{15}{30} = 0.50$

Third class $= \frac{45}{56} = 0.8$ and $\frac{45}{50} = 0.9$

4.7 : Cross-Validation

Q.28 What is cross - validation ? How it improves the accuracy of the outcome ?

Ans. : • Cross-validation is a technique for evaluating ML models by training several ML models on subsets of the available input data and evaluating them on the complementary subset of the data. Use cross-validation to detect overfitting, i.e., failing to generalize a pattern.

- In general, ML involves deriving models from data, with the aim of achieving some kind of desired behavior, e.g., prediction or classification.
- But this generic task is broken down into a number of special cases. When training is done, the data that was removed can be used to test the performance of the learned model on ``new" data. This is the basic idea for a whole class of model evaluation methods called **cross validation**.
- Types of cross validation methods are holdout, K-fold and leave-one-out.
- The holdout method is the simplest kind of cross validation. The data set is separated into two sets, called the training set and the testing set. The function approximate fits a function using the training set only.
- K-fold cross validation is one way to improve over the holdout method. The data set is divided into k subsets, and the holdout method is repeated k times. Each time, one of the k subsets is used as the test set and the other k-1 subsets are put together to form a training set. Then the average error across all k trials is computed.
- Leave-one-out cross validation is K-fold cross validation taken to its logical extreme, with K equal to N, the number of data points in the set. That means that N separate times, the function approximate is trained on all the data except for one point and a prediction is made for that point.

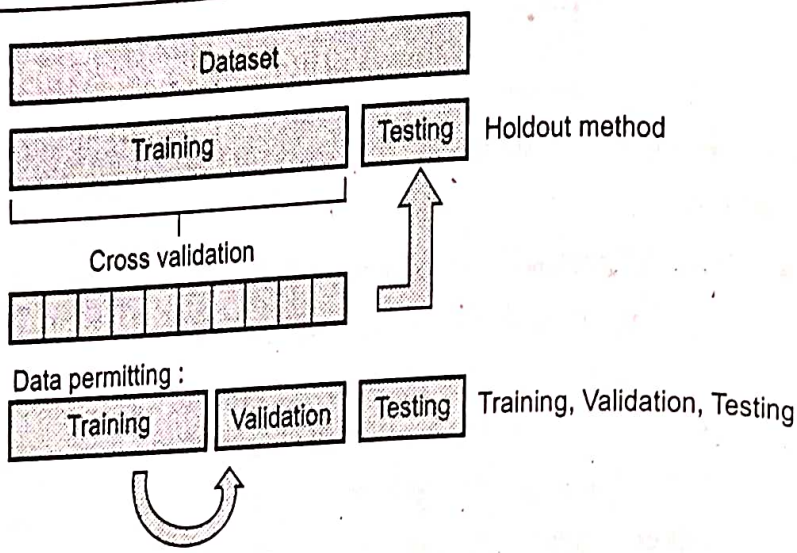


Fig. Q.28.1

Q.29 Explain with example K-fold cross validation.

Ans. : • K-fold CV is where a given data set is split into a K number of sections/folds where each fold is used as testing set at some point.

- Lets take the scenario of 5-Fold cross validation (K = 5). Here, the data set is split into 5 folds.
- In the first interaction, the first fold is used to test the model and the rest are used to train the model. In the second iteration, 2nd fold is used as the testing set while the rest serve as the training set. This process is repeated until each fold of the 5 folds has been used as the testing set.
- K-fold cross validation is performed as per the following steps :

1. Partition the original training data set into k equal subsets. Each subset is called a fold. Let the folds be named as $f_1, f_2, \ldots f_k$.
2. For i = 1 to i = k

- Keep the fold f_i as validation set and keep all the remaining k – 1 folds in the cross validation training set.

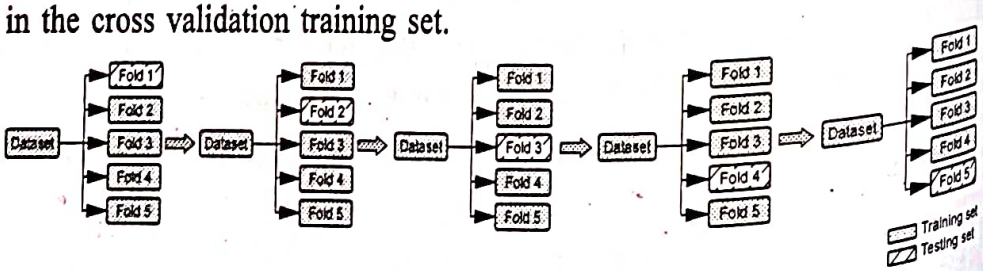


Fig. Q.29.1

3. Estimate the accuracy of your machine learning model by averaging the accuracies derived in all the k cases of cross validation.

- In the k-fold cross validation method, all the entries in the original training data set are used for both training as well as validation. Also, each entry is used for validation just once.
- The advantage of this method is that it matters less how the data gets divided. Every data point gets to be in a test set exactly once and gets to be in a training set $k-1$ times. The variance of the resulting estimating is reduced as k is increased.
- The disadvantage of this method is that the training algorithm has to be rerun scratch k times, which means it takes k times as much computation to make an evaluation. A variant of this method is to randomly divide the data into a test and training set k different times.
- The advantage of doing this is that you can independently choose how large each test set is and how many trials you average over.

4.8 : Micro-Average

Q.30 Write short note on micro-average methods.

Ans. : • In micro-average technique, you sum up the individual proper positives, fake positives and false negatives of the gadget for exceptional units and then apply them to get the statistics. For example, for a set of information, the system's

True advantageous (TP1) = 12

False nice (FP1) = 9

False terrible (FN1) = three

- Then precision (P1) and recollect (R1) could be 57.14 and 80 and for a one-of-a-kind set of information, the gadget's

True effective (TP2) = 50

False fine (FP2) = 23

False bad (FN2) = 9

- Then precision (P2) and do not forget (R2) will be sixty eight.49 and 84.Seventy five.

- Now, the common precision and consider of the machine the use of the Micro-common approach is

Micro-average of precision = (TP1+TP2)/(TP1+TP2+FP1+FP2)
= (12+50)/(12+50+9+23) = 65.Ninety six

Micro-average of don't forget = (TP1+TP2)/(TP1+TP2+FN1+FN2)
= (12+50)/(12+50+Three+9) = Eighty three.Seventy eight

- The micro-common F-score might be simply the harmonic imply of those figures.

- Eg-In micro-average approach, you sum up the man or woman proper positives, fake positives and fake negatives of the system for exceptional sets and the practice them to get the facts. For example, for a set of statistics, then device's

  True tremendous (TP1) = 12

  False effective (FP1) = nine

  False terrible (FN1) = three

- Then precision (P1) and bear in mind (R1) could be fifty seven.14 and eighty and for a special set of information, the gadget's

  True tremendous (TP2) = 50

  False positive (FP2) = 23

  False bad (FN2) = nine

- Then precision (P2) and recollect (R2) can be 68.49 and eighty four.Seventy five.

- Now, the common precision and recall of the system the use of the micro-average technique is

Micro-average of precision = (TP1+TP2)/(TP1+TP2+FP1+FP2)
= (12+50)/(12+50+9+23) = 65.Ninety six

Micro-common of consider = (TP1+TP2)/(TP1+TP2+FN1+FN2)
= (12+50)/(12+50+Three+Nine) = 83.Seventy eight

The micro-average F-score can be virtually the harmonic imply of those two figures.

4.9 : Macro-Average

Q.31 Write short note on macro-average methods.

Ans. : • Macro-average approach can be used whilst you need to know how the machine performs usually across the units of facts. You have to no longer give you any particular choice with this average.

- On the other hand, micro-common may be a useful measure whilst your dataset varies in length.
- This method is simple. Just take the common of the precision and don't forget the system on specific sets. For example, the macro-average precision and don't forget of the gadget for the given example is

Macro-average precision = (P1+P2)/2 = (57.14+68.Forty nine)/2
= Sixty two. Eighty two

Macro-average don't forget = (R1+R2)/2 = (80+84.75)/2 = Eighty two.25

Suitability

- Macro-average approach can be used when you want to recognise how the gadget plays ordinary across the sets of information. You must no longer provide you with any unique decision with this common.
- On the opposite hand, micro-common can be a useful measure while your dataset varies in size.

END...

Unit V

5 Unsupervised Learning

5.1 : Introduction to Clustering

Q.1 Define the following :
Cluster, clustering, cluster analysis.

Ans. : • **Cluster :** Cluster is a group of objects that belong to the same class. In other words the similar object are grouped in one cluster and dissimilar are grouped in other cluster.

- **Clustering :** Clustering is a process of partitioning a set of data in a set of meaningful subclasses. Every data in the subclass shares a common trait. It helps a user understand the natural grouping or structure in a data set.
- **Cluster analysis :** Cluster analysis is process of grouping a set of data-objects into clusters.

Q.2 Write a note on clustering trees.

Ans. : • Clustering is an exploratory data analysis task. It aims to find the intrinsic structure of data by organizing data objects into similarity groups or clusters.

- Let P be a finite set of points in an om-space. If the distances between different pairs of points in P are of different orders of magnitude, then the om-space imposes a unique tree-like hierarchical structure on P.
- The points will naturally fall into clusters, each cluster C being a collection of points all of which are much closer to one another than to any point in P outside C.
- The collection of all the clusters over P forms a strict tree under the subset relation.

- Moreover, the structure of this tree and the comparative sizes of different clusters in the tree captures all of the order-of-magnitude relations between any pair of points in P.
- A cluster tree is a tree T such that :
  1. Every leaf of T is a distinct symbol.
  2. Every internal node of T has at least two children.
  3. Each internal node of T is labelled with a non - negative value. Two or more nodes may given the same value.
  4. Every leaf of the tree is labelled 0.
  5. The label of every internal node in the tree is less than the label of its parent.

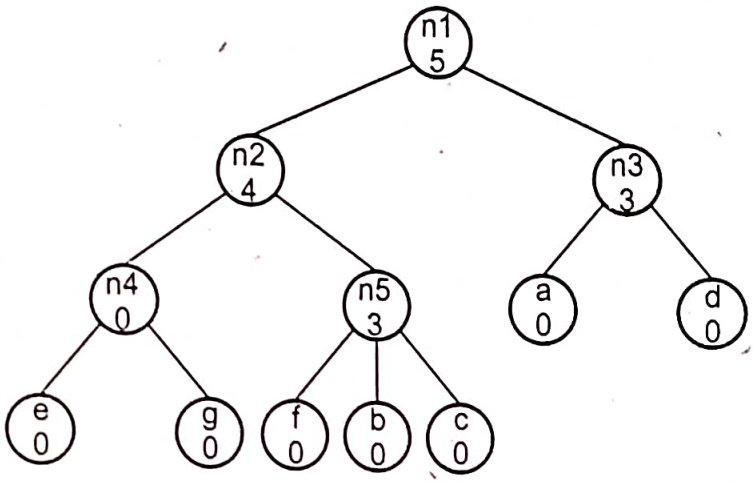


Fig. Q.2.1

- For any node N of T, the field "N.symbols" gives the set of symbols in the leaves in the subtree of T rooted at N, and the field "N.label" gives the integer label on node N.
- **Cluster tree construction :** This step uses a modified decision tree algorithm with a new purity function to construct a cluster tree to capture the natural distribution of the data without making any prior assumptions.
- **Cluster tree pruning :** After the tree is built, an interactive pruning step is preformed to simplify the tree to find meaningful / useful clusters. The final clusters are expressed as a list of hyper - rectangular regions.

Q.3 What is cluster analysis ? Explain desirable properties of clustering algorithm.

Ans. : • Given a set of objects, place them in groups such that the objects in a group are similar (or related) to one another and different from (or unrelated to) the objects in other groups.

- Cluster analysis can be a powerful data - mining tool for any organisation that needs to identify discrete groups of customers, sales transactions or other types of behaviors and things. For example, insurance providers use cluster analysis to detect fraudulent claims and banks use it for credit scoring.
- Cluster analysis uses mathematical models to discover groups of similar customers based on the smallest variations among customers within each group.
- Cluster is a group of objects that belong to the same class. In other words the similar object are grouped in one cluster and dissimilar are grouped in other cluster.
- Clustering is a process of partitioning a set of data in a set of meaningful subclasses. Every data in the subclass shares a common trait. It helps a user understand the natural grouping or structure in a data set.
- Various types of clustering methods are partitioning methods, hierarchical clustering, Fuzzy clustering, Density - based clustering and Model - based clustering.
- Cluster analysis is process of grouping a set of data - objects into clusters.
- **Desirable properties of a clustering algorithm are as follows :**

1. Scalability (in terms of both time and space).
2. Ability to deal with different data types.
3. Minimal requirements for domain knowledge to determine input parameters.
4. Interpretability and usability.

- Clustering of data is a method by which large sets of data are grouped into clusters of smaller sets of similar data. Clustering can be considered the most important unsupervised learning problem.

Q.4 Explain typical requirement of clustering.

Ans. :

1. **Scalability :** Many clustering algorithms work well on small data sets.
2. **Ability to deal with different types of attributes :** Many algorithms are designed to cluster interval-based data.
3. **Discovery of clusters with arbitrary shape :** Many clustering algorithms determine clusters based on Euclidean or Manhattan distance measures.
4. **Minimal requirements for domain knowledge to determine input parameters :** Many clustering algorithms require users to input certain parameters in cluster analysis.
5. **Ability to deal with noisy data :** Most real - world databases contain outliers or missing, unknown or erroneous data. Some clustering algorithms are sensitive to such data and may lead to clusters of poor quality.
6. **Incremental clustering and insensitivity to the order of input records :** Some clustering algorithms cannot incorporate newly inserted data into existing clustering structures.
7. **High dimensionality :** A database or a data warehouse can contain several dimensions or attributes.
8. **Constraint-based clustering :** Real-world applications may need to perform clustering under various kinds of constraints.
9. **Interpretability and usability :** Users expect clustering results to be interpretable, comprehensible and usable.

Q.5 Explain various types of clustering.

Ans. : • Type of clusters are as follows :

a) Well - separated clusters

b) Prototype - based clusters

c) Contiguity - based clusters

d) Density - based clusters

a) Well - separated clusters

- A cluster is a set of points such that any point in a cluster is closer to every other point in the cluster than to any point not in the cluster.
- Fig. Q.5.1 shows well-separated cluster.

Fig. Q.5.1 Well-separated cluster

- Sometimes a threshold is used to specify that all the objects in a cluster must sufficiently close to one another. Definition of a cluster is satisfied only when the data contains natural clusters.

b) Prototype - based cluster

- A cluster is a set of objects such that an object in a cluster is closer (more similar) to the prototype or "center" of a cluster, than to the center of any other cluster. Prototype based clusters can also be referred to as "Center-Based" Clusters.
- The center of a cluster is often a centroid, the average of all the points in the cluster, or a medoid, the most "representative" point of a cluster. Fig. Q.5.2 shows 4 center-based clusters.

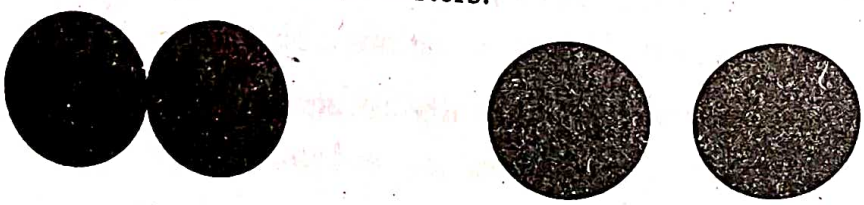


Fig. Q.5.2 4 Center-based clusters

- If the data is numerical, the prototype of the cluster is often a centroid i.e., the average of all the points in the cluster.

- If the data has categorical attributes, the prototype of the cluster is often a medoid i.e., the most representative point of the cluster.
- Objects in the cluster are closer to the prototype of the cluster than to the prototype of any other cluster.
- K-Means and K-Medoids are the examples of prototype-based clustering algorithm.

c) Contiguity - based clusters

- A cluster is a set of points such that a point in a cluster is closer (or more similar) to one or more other points in the cluster than to any point not in the cluster.

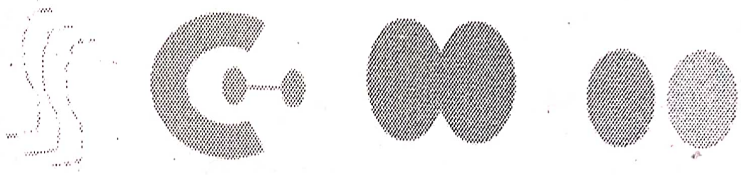
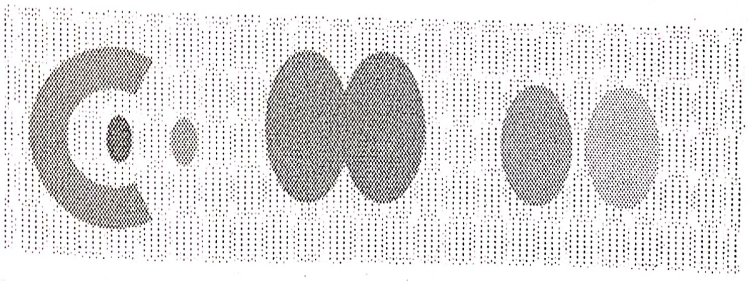


Fig. Q.5.3 8 contiguous clusters

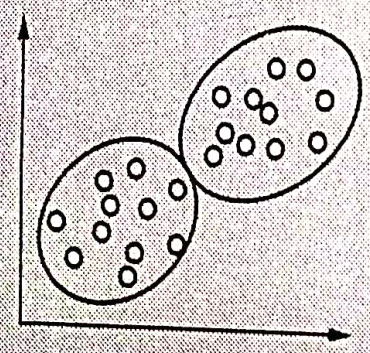
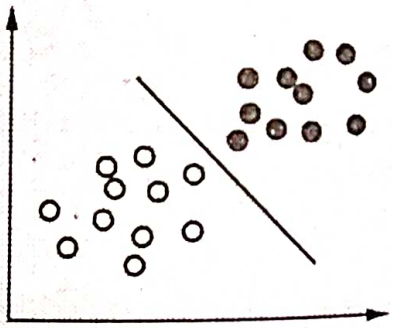
d) Density - based

- A cluster is a dense region of points, which is separated by low-density regions, from other regions of high density.
- Used when the clusters are irregular or intertwined, and when noise and outliers are present.



Q.6 Explain difference between clustering and classification.

Ans. :

Clustering	Classification
This function maps the data into one of several clusters which is the grouping of data items based on the similarities between them.	This model function classifies the data into one of several predefined categorical classes.
Involved in unsupervised learning.	Involved in supervised learning.
Training sample is not provided.	Training sample is provided.
The number of cluster is not known before clustering, These are identified after the completion of clustering.	The number of classes is known before classification as there is predefined output based on input data.
Data is not labeled.	Labeled data points.
Asks how can I group this set of items ?	Asks what class does this item belong to ?
Unknown number of classes.	Known number of classes.
Used to understand data.	Used to classify future observations.
	

Q.7 List the problems associated with clustering.

Ans. : Problems with clustering are as follows :

1. Current clustering techniques do not address all the requirements adequately.
2. Dealing with large number of dimensions and large number of data items can be problematic because of time complexity.

3. The effectiveness of the method depends on the definition of "distance".
4. If an obvious distance measure doesn't exist we must "define" it, which is not always easy, especially in multi-dimensional spaces.
5. The result of the clustering algorithm can be interpreted in different ways.

5.2 : K-Means and K-Medoids

Q.8 Write K-means algorithm.

Ans. : • K-Means clustering is heuristic method. Here each cluster is represented by the center of the cluster.

- The "K" stands for number of clusters, it is typically a user input to the algorithm; some criteria can be used to automatically estimate K.
- This method initially takes the number of components of the population equal to the final required number of clusters.
- In this step itself the final required number of clusters is chosen such that the points are mutually farthest apart.
- Given K, the K-means algorithm consists of four steps :
  1. Select initial centroids at random.
  2. Assign each object to the cluster with the nearest centroid.
  3. Compute each centroid as the mean of the objects assigned to it.
  4. Repeat previous 2 steps until no change.
- The $x_1, \ldots, x_N$ are data points or vectors of observations. Each observation (vector x_i) will be assigned to one and only one cluster. The C(i) denotes cluster number for the ith observation. K - means minimizes within cluster scatter :

$$W(C) = \frac{1}{2} \sum_{K=1}^{K} \sum_{C(i)=K} \sum_{C(j)=K} \left\| x_i - x_j \right\|^2$$

$$= \sum_{K=1}^{K} N_k \sum_{C(i)=K}^{K} \left\| x_i - x_j \right\|^2$$

where, N_k is the number of observations in K^{th} cluster.

K-Means Algorithm Properties :

1. There are always K clusters.
2. There is always at least one item in each cluster.
3. The clusters are non-hierarchical and they do not overlap.
4. Every member of a cluster is closer to its cluster than any other cluster because closeness does not always involve the 'center' of clusters.

The K-Means Algorithm Process :

1. The dataset is partitioned into K clusters and the data points are randomly assigned to the clusters resulting in clusters that have roughly the same number of data points.
2. For each data point-
   a. Calculate the distance from the data point to each cluster.
   b. If the data point is closest to its own cluster, leave it where it is.
   c. If the data point is not closest to its own cluster, move it into the closest cluster.
3. Repeat the above step until a complete pass through all the data points results in no data point moving from one cluster to another. At this point the clusters are stable and the clustering process ends.
4. The choice of initial partition can greatly affect the final clusters that result, in terms of inter-cluster and intra-cluster distances and cohesion.

- K-means algorithm is iterative in nature. It converges, however only a local minimum is obtained. It works only for numerical data. This method easy to implement.

Q.9 Explain single linkage, complete linkage and average linkage.

Ans. : • In the agglomerative hierarchical approach, we start by defining each data point to be a cluster and combine existing clusters at each step.

• Here are four different methods for doing this :

1. Single linkage :

- Smallest pairwise distance between elements from each cluster. Also referred to as nearest neighbour or minimum method.

- This measure defines the distance between two clusters as the minimum distance found between one case from the first cluster and one case from the second cluster.
- For example, if cluster 1 contains cases a and b, and cluster 2 contains cases c, d, and e, then the distance between cluster 1 and cluster 2 would be the smallest distance found between the following pairs of cases : (a, c), (a, d), (a, e), (b, c), (b, d), and (b, e).

2. **Complete linkage** : Largest distance between elements from each cluster.

- Also referred to as furthest neighbor or maximum method. This measure is similar to the single linkage measure described above, but instead of searching for the minimum distance between pairs of cases, it considers the furthest distance between pairs of cases.
- Although this solves the problem of chaining, it creates another problem.
- Imagine that in the above example cases a, b, c, and d are within close proximity to one another based upon the pre-established set of variables; however, if case e differs considerably from the rest, then cluster 1 and cluster 2 may no longer be joined together because of the difference in scores between (a, e) and (b, e).
- In complete linkage, outlying cases prevent close clusters to merge together because the measure of the furthest neighbour exacerbates the effects of outlying data.

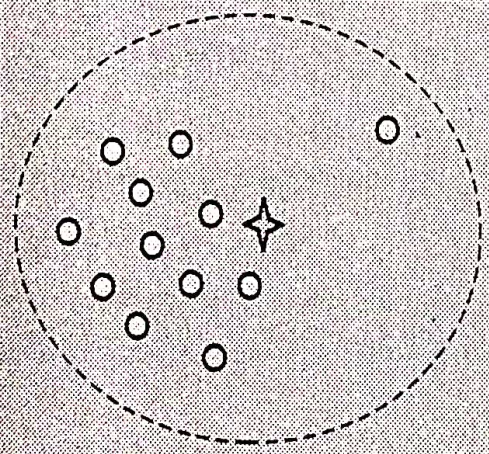
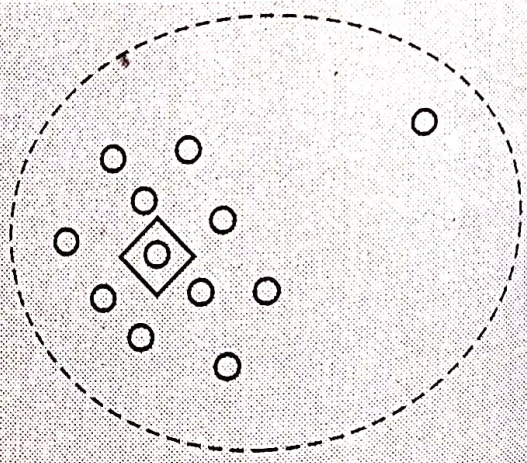
3. **Average linkage** : The average distance between elements from each cluster

- Also referred to as the unweighted pair-group method using Arithmetic averages.
- To overcome the limitations of single and complete linkage, Sokal and Michener proposed taking an average of the distance values between pairs of cases.

- This method is supposed to represent a natural compromise between the linkage measures to provide a more accurate evaluation of the distance between clusters.
- For average linkage, the distances between each case in the first cluster and every case in the second cluster are calculated and then averaged.

Q.10 Explain the difference between K-mean and K-medoids clustering.

Ans. :

K-means	K-medoids
Each cluster is represented by the center of the cluster.	Each cluster is represented by one of the objects in the cluster.
Simple centroid-based method.	Data point are chosen be the medoids.
K-means clustering is a non-hierarchical cluster analysis method that attempts to partition existing objects into one or more clusters or groups of objects based on their characteristics.	K-medoid is a classic partition clustering technique that groups data sets of ni objects into k groups known a priori.
K-means algorithm is very prone to the effects of outliers.	Normally less delicate to outliers than K-means.
Convex shape is required.	Convex shape is not required.
	

Q.11 What is difference between k-NN and K-means clustering ?

Ans. :

k-NN	K-means clustering
KNN is a supervised learning algorithm.	k-Means clustering is an unsupervised learning.
kNN used for classification.	k-Means is used for clustering.
K-nearest neighbors needs labelled data to train on.	K-means clustering needs unlabelled data to train.
'K' in KNN is the number of nearest neighbours used to classify or a test sample.	K' in K-Means is the number of clusters the algorithm is trying to identify/learn from the data.
Centroid is the point X to be classified.	Centroids are not necessarily data points.

Q.12 Consider the following data set consisting of the scores of two variables on each of seven individuals : Apply K-mean clustering to cluster this data into 2 clusters.

Subject	*A*	*B*
1	1.0	1.0
2	1.5	2.0
3	3.0	4.0
4	5.0	7.0
5	3.5	5.0
6	4.5	5
7	3.5	4.5

☞ [SPPU: June-22, End Sem, Marks 8]

Ans. : Given data set is grouped into two clusters. First find the initial partition, so the A & B values of the two individuals furthest apart (using Euclidean distance), define the initial cluster :

	Individual	Mean vector (Centroid)
Group 1	1	(1.0, 1.0)
Group 2	4	(5.0, 7.0)

- In sequence, remaining individuals are examined and allocated to the cluster to which they are closest, using Euclidean distance to the cluster mean. Each time, mean vector is recalculated and new member is added. Following are the steps :

	Cluster 1		Cluster 2	
		Mean		Mean
Step	Individual	Vector (Centroid)	Individual	Vector (Centroid)
1	1	(1.0, 1.0)	4	(5.0, 7.0)
2	1, 2	(1.2, 1.5)	4	(5.0, 7.0)
3	1, 2, 3	(1.8, 2.3)	4	(5.0, 7.0)
4	1, 2, 3	(1.8, 2.3)	4, 5	(4.2, 6.0)
5	1, 2, 3	(1.8, 2.3)	4, 5, 6	(4.3, 5.7)
6	1, 2, 3	(1.8, 2.3)	4, 5, 6, 7	(4.1, 5.4)

- Now the initial partition has changed, and the two clusters at this stage having the following characteristics :

	Individual	Mean vector (Centroid)
Cluster 1	1, 2, 3	(1.8, 2.3)
Cluster 2	4, 5, 6, 7	(4.1, 5.4)

- Compare each individuals distance to its own cluster mean and to that of the opposite cluster because individual cluster assign to right cluster is not confirmed.

Individual	Distance to mean (Centroid) of cluster 1	Distance to mean (Centroid) of cluster 2
1	1.5	5.4
2	0.4	4.3
3	2.1	1.8
4	5.7	1.8
5	3.2	0.7
6	3.8	0.6
7	2.8	1.1

- Each individual's distance to its own cluster mean should be smaller than the distance to the other cluster's mean (which is not the case with individual 3). Thus, individual 3 is relocated to cluster 2 resulting in the new partition :

	Individual	Mean vector (Centroid)
Cluster 1	1, 2	(1.3, 1.5)
Cluster 2	3, 4, 5, 6, 7	(3.9, 5.1)

Q.13 Consider following instance given as input to K-means clustering algorithm for k = 3. Find members of these 3 clusters after 2 iterations. X = {(2,10), (2,5), (8,4), (5,8), (7,5), (6,4), (1,2), (4,9)}

Ans. : We rewrite the given data set :

Data set : $A_1(2,10)$, $A_2(2,5)$, $A_3(8,5)$, $B_1(5,8)$, $B_2(7,5)$, $B_3(6,4)$, $C_1(1,2)$, $C_2(4,9)$

Centroids : $A_1(2,10)$, $B_1(5,8)$, $C_1(1,2)$.

Iteration 1 : We need to calculate the distance between each data points and the centroids using the Euclidean distance.

Two points (x_1, y_1), (x_2, y_2)

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

or = $|x_2 - x_1| + |y_2 - y_1|$

Mean formula : $((x_1 + x_2)/2, (y_1 + y_2)/2)$

1st ROW :

Distance calculate between the A_2 data point and the Centroids A_1, B_1, C_1

Distance between $A_2(2,5)$ and $A_1(2,10) = |2 - 2| + |5 - 10| = 0 + 5 = 5$

Distance between $A_2(2,5)$ and $B_1(5,8) = |2 - 5| + |5 - 8| = 3 + 3 = 6$

Distance between $A_2(2,5)$ and $C_1(1,2) = |2 - 1| + |5 - 2| = 1 + 3 = 4$

The A_2 nearby Cluster Center is C_1.

2nd ROW : Distance calculate between the A_3 data point and the Centroids A_1, B_1, C_1

Distance between $A_3(8,5)$ and $A_1(2,10) = 11$

Distance between $A_3(8,5)$ and $B_1(5,8) = 6$

Distance between $A_3(8,5)$ and $C_1(1,2) = 10$

The A_3 nearby Cluster Center is B_1.

3rd ROW : Distance calculate between the B_2 data point and the Centroids A_1, B_1, C_1

Distance between $B_2(7,5)$ and $A_1(2,10) = 10$

Distance between $B_2(7,5)$ and $B_1(5,8) = 5$

Distance between $B_2(7,5)$ and $C_1(1,2) = 9$

The B_2 nearby Cluster Center is B_1.

4th ROW : Distance calculate between the B_3 data point and the Centroids A_1, B_1, C_1

Distance between $B_3(6,4)$ and $A_1(2,10) = 10$

Distance between $B_3(6,4)$ and $B_1(5,8) = 5$

Distance between $B_3(6,4)$ and $C_1(1,2) = 7$

The B_3 nearby Cluster Center is B_1.

5th ROW : Distance calculate between the C_2 data point and the Centroids A_1, B_1, C_1

Distance between $C_2(4,9)$ and $A_1(2,10) = 3$

Distance between $C_2(4,9)$ and $B_1(5,8) = 2$

Distance between $C_2(4,9)$ and $C_1(1,2) = 10$

The C_1 nearby Cluster Center is B_1.

The above calculations are shown in the form of below table :

Data Sets	Centroids			Cluster
	$A_1(2,10)$	$B_1(5,8)$	$C_1(1,2)$	
$A_2(2,5)$	5	6	4	C_1
$A_3(8,5)$	11	6	10	B_1
$B_2(7,5)$	10	5	9	B_1
$B_3(6,4)$	10	5	7	B_1
$C_2(4,9)$	3	2	10	B_1

Thus we need to calculate the cluster mean values

Cluster $B_1(5,8)$ nearby points are $A_3(8,5)$, $B_2(7,5)$, $B_3(6,4)$, $C_2(4,9)$

B_1 Mean value = (6, 6.2)

Cluster $C_1(1,8)$ nearby points are $A_2(2,5)$

C_1 Mean value = (1.5, 3.5)

The updated Cluster points are : $A_1(2,10)$, $B_1(6, 6.2)$, $C_1(1.5, 3.5)$

Now we need to go for the next iteration with the updated cluster points.

Iteration 2 : Now we need to calculate the distance between the each data points to centroids

Data Sets	Centroids			Cluster
	$A_1(2,10)$	$B_1(6,6.2)$	$C_1(1.5,3.5)$	
$A_2(2,5)$	5	5.2	2	C_1
$A_3(8,5)$	11	3.2	8	B_1
$B_2(7,5)$	10	2.2	7	B_1
$B_3(6,4)$	10	2.2	5	B_1
$C_2(4,9)$	3	4.8	8	A_1

So, after completion of the iteration 2 the cluster points are not equal to the iteration 1 cluster points and then we need to go for the iteration 3 before that we need to calculate the cluster mean values.

Cluster $A_1(2,10)$ nearby points are $C_2(4,9)$

A_1 Mean value = (3, 9.5)

Cluster $B_1(6,6.2)$ nearby points are $A_3(8,5)$, $B_2(7,5)$, $B_3(6,4)$

B_1 Mean value = (6.7, 4)

Cluster $C_1(1.5,3.5)$ nearby points are $A_2(2,5)$

C_1 Mean value = (1.7, 4.2)

The updated Cluster points are : $A_1(3,9.5)$, $B_1(6.7, 4)$, C_1 = (1.7, 4.2)

5.3 : Hierarchical Clustering

Q.14 What is hierarchical clustering ? List its types.

Ans. : • Hierarchical clustering arranges items in a hierarchy with a tree like structure based on the distance or similarity between them. The graphical representation of the resulting hierarchy is a tree-structured graph called a dendrogram.

- The tree is not a single set of clusters, but rather a multilevel hierarchy, where clusters at one level are joined as clusters at the next level.
- The hierarchical clustering algorithm is an unsupervised Machine Learning technique.

- Hierarchical clustering starts with k = N clusters and proceed by merging the two closest objects into one cluster, obtaining k = N – 1 clusters. The process of merging two clusters to obtain k – 1 clusters is repeated until we reach the desired number of clusters K
- Hierarchical clustering algorithms are of two types : Divisive and Agglomerative
- Divisive methods initialize with all examples as members of a single cluster, and split this cluster recursively.
- Agglomerative methods begin instead with each point defining its own cluster, and iteratively link nearby points into larger and larger clusters.

Q.15 Define cluster tree ? Write and explain agglomerative clustering algorithm.

Ans. : • A cluster tree is a tree T such that : Every leaf of T is a distinct symbol. Every internal node of T has at least two children. Each internal node of T is labelled with a non-negative value. Two or more nodes may be given the same value.

- The agglomerative clustering is the most common type of hierarchical clustering used to group objects in clusters based on their similarity.
- This algorithm works by grouping the data one by one on the basis of the nearest distance measure of all the pairwise distance between the data point. Again distance between the data point is recalculated but which distance to consider when the groups has been formed? For this there are many available methods. Some of them are :
  1) Single-nearest distance or single linkage.
  2) Complete-farthest distance or complete linkage.
  3) Average-average distance or average linkage.
  4) Centroid distance.
- This way we go on grouping the data until one cluster is formed. Now on the basis of dendogram graph we can calculate how many number of clusters should be actually present.

- Steps for an agglomerative hierarchical cluster analysis :
  1. Find the similarity or dissimilarity between every pair of objects in the data set. In this step, we calculate the distance between objects using the pdist function. The pdist function supports many different ways to compute this measurement.
  2. Group the objects into a binary, hierarchical cluster tree. In this step, we link pairs of objects that are in close proximity using the linkage function. The linkage function uses the distance information generated in step 1 to determine the proximity of objects to each other. As objects are paired into binary clusters, the newly formed clusters are grouped into larger clusters until a hierarchical tree is formed.
  3. Determine where to cut the hierarchical tree into clusters. In this step, we use the cluster function to prune branches off the bottom of the hierarchical tree and assign all the objects below each cut to a single cluster. This creates a partition of the data. The cluster function can create these clusters by detecting natural groupings in the hierarchical tree or by cutting off the hierarchical tree at an arbitrary point.

Q.16 Explain advantages and disadvantages of hierarchical clustering.

Ans. : Advantages

- It is simple to implement.
- It is easy and results in a hierarchy, a structure that contains more information.
- It does not need us to pre-specify the number of clusters.

Disadvantages

- It breaks the large clusters.
- It is difficult to handle different sized clusters and convex shapes.
- It is sensitive to noise and outliers.
- The algorithm can never be changed or deleted once it was done previously.

Q.17 What is agglomerative clustering ? Explain with example.

☞ [SPPU : June-22, End Sem, Marks 8]

Ans. : • It is also known as AGNES (Agglomerative Nesting). It works in a bottom-up manner.

- That is, each object 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 member of just one single big cluster (root). The result is a tree which can be plotted as a dendrogram. Most hierarchical clustering methods belong to this category.
- Initially, AGNES places each objects into a cluster of its own. The clusters are then merged step-by-step according to some criterion.
- For example, cluster C_1 and C_2 may be merged if an object in C_1 and object in C_2 form the minimum Euclidean distance between any two objects from different clusters.
- In the agglomerative hierarchical approach, we start by defining each data point to be a cluster and combine existing clusters at each step. Here are four different methods for doing this :
  1. **Single linkage :** Smallest pairwise distance between elements from each cluster. It tends to produce long, "loose" clusters.
  2. **Complete linkage :** Largest distance between elements from each cluster. It tends to produce more compact clusters
  3. **Average linkage :** The average distance between elements from each cluster. It can vary in the compactness of the clusters it creates.
  4. **Centroid linkage :** Distance between cluster means.
- Agglomerative clustering is good at identifying small clusters.

Steps for an agglomerative hierarchical cluster analysis

1. Find the similarity or dissimilarity between every pair of objects in the data set. In this step, we calculate the distance between objects using the pdist function. The pdist function supports many different ways to compute this measurement.

2. Group the objects into a binary, hierarchical cluster tree. In this step, we link pairs of objects that are in close proximity using the linkage function. The linkage function uses the distance information generated in step 1 to determine the proximity of objects to each other. As objects are paired into binary clusters, the newly formed clusters are grouped into larger clusters until a hierarchical tree is formed.
3. Determine where to cut the hierarchical tree into clusters. In this step, we use the cluster function to prune branches off the bottom of the hierarchical tree, and assign all the objects below each cut to a single cluster. This creates a partition of the data. The cluster function can create these clusters by detecting natural groupings in the hierarchical tree or by cutting off the hierarchical tree at an arbitrary point.

Q.18 Explain difference between agglomerative and divisive clustering.

Ans. :

Sr. No.	Agglomerative clustering	Divisive clustering
1.	Agglomerative clustering is known as bottom-up approach.	Divisive clustering is known as the top-down approach.
2.	Agglomerative clustering is good at identifying small clusters.	Divisive clustering is good at identifying large clusters.
3.	Each cluster starts with only one object.	Start with all object in one cluster.
4.	It is also known as AGNES (Agglomerative Nesting).	It is also known as DIANA (Divise Analysis).
5.	Iteratively clusters are merged together.	Large clusters are successively divided.

5.4 : Density-Based Clustering

Q.19 What is density-based clustering ? Explain its working.

Ans. : • Density-based clustering is unsupervised learning methodologies used in model building and machine learning algorithms. This is a clustering method that is used in machine learning to separate clusters of high density from clusters of low density.

- DBSCAN stands for "Density-Based Spatial Clustering of Applications with Noise." DBSCAN groups together closely-packed points.
- There are two inputs to DBSCAN :
  1. The search distance around point (∈).
  2. The minimum number of points (minpts) required to form a density cluster.
- DBSCAN is based on this intuitive notion of "clusters" and "noise". The key idea is that for each point of a cluster, the neighborhood of a given radius has to contain at least a minimum number of points.
- The parameter (∈) defines the radius of neighborhood around a point x. It is called the ∈-neighborhood of x. The parameter MinPts is the minimum number of neighbors within "eps" radius.
- Any point x in the dataset, with a neighbor count greater than or equal to MinPts, is marked as a core point. We say that x is border point, if the number of its neighbors is less than MinPts, but it belongs to the ∈-neighborhood of some core point z. Finally, if a point is neither a core nor a border point, then it is called a noise point or an outlier.
- Fig. Q.19.1 shows DBSCAN.

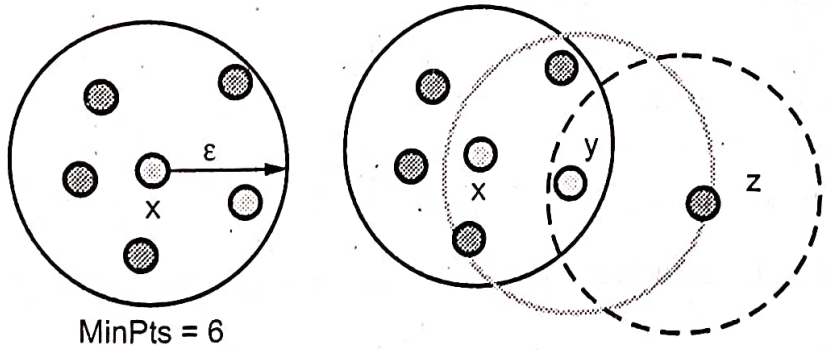


Fig. Q.19.1 DBSCAN

- We define following terms for understanding the DBSCAN algorithm :
  1. **Direct density reachable :** A point "A" is directly density reachable from another point "B" if : "A" is in the ∈-neighborhood of "B" and "B" is a core point.
  2. **Density reachable :** A point "A" is density reachable from "B" if there are a set of core points leading from "B" to "A.
  3. **Density connected :** Two points "A" and "B" are density connected if there are a core point "C", such that both "A" and "B" are density reachable from "C".
- DBSCAN is broadly used in many applications such as market research, pattern recognition, data analysis, and image processing.

Q.20 Explain spectral clustering with its advantages and disadvantages.

Ans. : • Spectral clustering techniques make use of the spectrum (eigenvalues) of the similarity matrix of the data to perform dimensionality reduction before clustering in fewer dimensions. The similarity matrix is provided as an input and consists of a quantitative assessment of the relative similarity of each pair of points in the dataset

- Given data points $x_1, \ldots x_N$, pairwise affinities $A_{ij} = A(x_i, x_j)$
- Build similarity graph shown in Fig. Q.20.1.
- Clustering = Find a cut through the graph
- Define a cut-type objective function.
- The low-dimensional space is determined by the data. Spectral clustering makes use of the spectrum of the graph for dimensionality reduction.

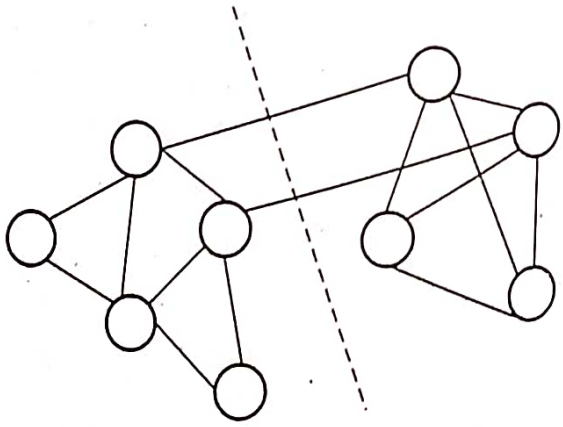


Fig. Q.20.1 Similarity graph

- Projection and clustering equates to graph partition by different min-cut criteria.

Advantages :

1. Does not make strong assumptions on the statistics of the clusters.
2. Easy to implement.
3. Good clustering results.
4. Reasonably fast for sparse data sets of several thousand elements.

Disadvantages :

1. May be sensitive to choice of parameters.
2. Computationaly expensive for large datasets.

5.5 : Outlier Analysis

Q.21 What is outlier analysis ? Explain.

Ans. : • A database may contain data objects that do not comply with the general behaviour model of the data. Such data objects, which are grossly different from or inconsistent with the remaining set of data, are called outliers.

- Outlier detection is the process of detecting and subsequently excluding outliers from a given set of data.
- Fig. Q.21.1 shows outliers detection. Here O_1 and O_2 seem outliers from the rest.

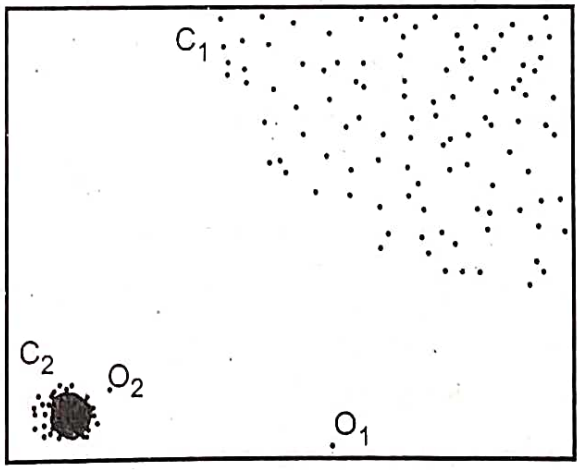


Fig. Q.21.1 Outliers detection

- An outlier may be defined as a piece of data or observation that deviates drastically from the given norm or average of the data set. An outlier may be caused simply by chance, but it may also indicate measurement error or that the given data set has a heavy-tailed distribution.
- Objective : Define what data can be considered as inconsistent in a given data set.

- Outlier analysis is used in various types of dataset, such as graphical dataset, numerical data set, text dataset and can also be used on the pictures etc.
- The identification of outlier can lead to the discovery of useful and meaningful knowledge. Outlier detection is the process of finding data objects with behaviours that are very different from expectation. Such objects are called outlier or anomalies.
- Finding outliers from as collection of pattern is a popular problem in the field of data mining. A key challenge with outlier analysis and detection is that it is not a well formulated problem like clustering so outlier detection as a branch of data mining requires more attention.
- Outlier analysis and detection has various applications in numerous fields such as fraud detection, credit card, discovering computer intrusion and criminal behaviours, medical and public health outlier detection, industrial damage detection.
- General idea of application is to find out data which deviates from normal behaviour of data set.

Q.22 What is local outlier factor ? How it is calculated ?

Ans. : • Local Outlier Factor (LOF) is an algorithm that identifies the outliers present in the dataset. Outlier detection methods can be distribution-based, depth-based, clustering-based and density-based. LOF allows to define outliers by doing density-based scoring. It is similar to the KNN (nearest neighbor search) algorithm.

- The Local Outlier Factor (LOF) algorithm is an unsupervised anomaly detection method which computes the local density deviation of a given data point with respect to its neighbors.
- The LOF algorithm can be used for outlier detection and novelty detection. The difference between outlier detection and novelty detection lies in the training dataset. Outlier detection includes outliers in the training dataset. The algorithm fits the areas with high-density data and ignores the outliers and anomalies.

- Novelty detection only includes the normal data points when training the model. Then the model will take a new dataset with outliers for prediction. The outliers in novelty detection are also called novelties.
- The local outlier factor method works by comparing the density of a point with the relative densities of its neighbours. If a point is relatively less dense than its neighbours, it is a potential outlier. If the ratio of density of neighbours to the density of point is too high, we end up with a high LOF signifying an outlier.
- We first define the K-distance of a point. K-distance(A) = Dist(A, K^{th} nearest neighbour).
- The K neighbourhood of a point is just the K closest points to a given point. $N_k(A) = \{P|Dist(A,P) <= K\text{-distance}(A)\}$.
- Reachability distance : It expresses the maximum of the distance of two points and the k-distance of the second point. The distance between the two points here can be Euclidean, Manhattan and Minkowski or any of the other distance measures.

Reachability $Distance_k$ (A,B) = max{k-distance(B), dist(A,B)}

- The local reachability densities found are compared to the local reacability densities of a's nearest k neighbors. The density of each neighbor is summed up and divided by the density of a. The value found is divided by the number of neighbors i.e. k.

$$LOF(a) = \frac{[(LRD(1^{st}.neighbor) + LRD(2^{nd}.neighbor) + ... + LRD(k^{th} neighbor))/LRD(a)]}{k}$$

5.6 : Evaluation Metrics and Score

Q.23 Explain evaluation metrics homogeneity and completeness.

Ans. : Homogeneity

- Homogeneity metric of a cluster labeling given a ground truth. A clustering result satisfies homogeneity if all of its clusters contain only data points which are members of a single class.

- This metric is independent of the absolute values of the labels : A permutation of the class or cluster label values won't change the score value in any way.
- To define the concepts of entropy H(X) and conditional entropy H(X|Y), which measures the uncertainty of X given the knowledge of Y.
- Therefore, if the class set is denoted as C and the cluster set as K, H(C|K) is a measure of the uncertainty in determining the right class after having clustered the dataset.
- To have a homogeneity score, it's necessary to normalize this value considering the initial entropy of the class set H(C) :

$$h = 1 - \frac{H(C|K)}{H(C)}$$

- In scikit-learn, there's the built-in function homogeneity_score() that can be used to compute this value : **from sklearn.metrics import homogeneity_score**

Completeness

- A complementary requirement is that each sample belonging to a class is assigned to the same cluster.
- A clustering result satisfies completeness if all the data points that are members of a given class are elements of the same cluster.
- This metric is independent of the absolute values of the labels : A permutation of the class or cluster label values won't change the score value in any way.
- This measure can be determined using the conditional entropy H(K|C), which is the uncertainty in determining the right cluster given the knowledge of the class. Like for the homogeneity score, we need to normalize this using the entropy H(K) :

$$c = 1 - \frac{H(C|K)}{H(K)}$$

- We can compute this score (on the same dataset) using the function completeness_score() :

```
from sklearn.metrics import completeness_score
```

Q.24 Explain elbow method for finding optimal number of clusters.

☞ [SPPU : June-22, End Sem, Marks 4]

Ans. : • The elbow method is used to determine the optimal number of clusters in k-means clustering. The elbow method plots the value of the cost function produced by different values of k.

- If k increases, average distortion will decrease, each cluster will have fewer constituent instances, and the instances will be closer to their respective centroids. However, the improvements in average distortion will decline as k increases. The value of k at which improvement in distortion declines the most is called the elbow, at which we should stop dividing the data into further clusters.
- It involves running the algorithm multiple times over a loop, with an increasing number of cluster choice and then plotting a clustering score as a function of the number of clusters.
- The Elbow and Silhouette methods are the two state-of-the-art methods used to identify the correct cluster number in the dataset.
- The Elbow method is the oldest method to distinguish the potential optimal cluster number for the analyzed dataset, whose basic idea is to specify K = 2 as the initial optimal cluster number K, and then keeps increasing K by step 1 to the maximal specified for the estimated potential optimal cluster number, and finally distinguish the potential optimal cluster number K corresponding to the plateau.
- The optimal cluster number K is distinguished by the fact that before reaching K, the cost rapidly decreases to the called cost peak value, and after exceeding K, it continues to increase with the called cost peak value almost unchanged, as shown in Fig. Q.24.1 (a) with an explicit elbow point.
- Meanwhile, the optimal cluster number corresponding to the elbow point depends on the manmade selection. There is, however, a problem

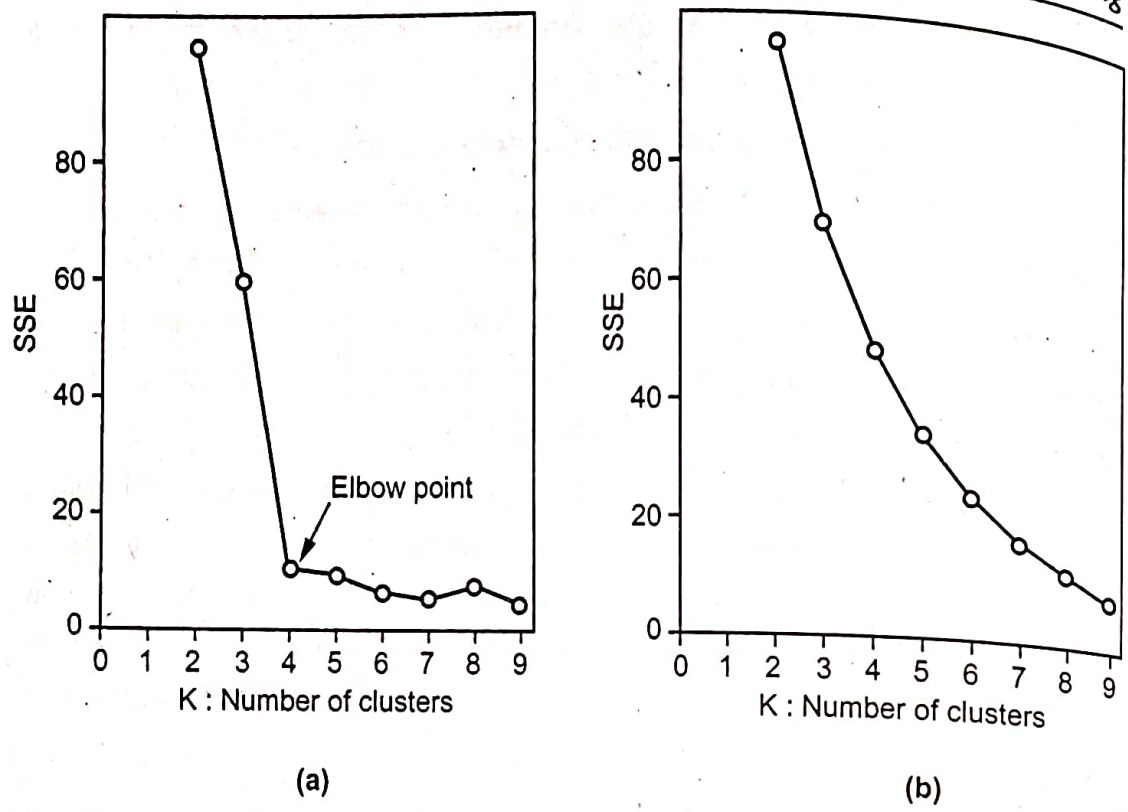


Fig. Q.24.1 Elbow point

with the Elbow method in that the elbow point cannot be unambiguously distinguished by the experienced analysts when the plotted curve is fairly smooth, as shown in Fig. 5.6.1 (b) with an ambiguous elbow point.

- To select the best K, we need to plot the mean in-cluster distance for each K. As K increases from 1, before reaching the optimal K, the decrease speed is relatively fast because the number of centers are too low from the very beginning and each new center will incur a large decrease in the mean distance.
- But after the optimal K, the decrease is slow since the correct cluster structure is already discovered and any newly added center will appear in a certain cluster already formed. That will not decrease the mean in-cluster distance too much. The entire curve looks like an L shape and the best K lies in the turning point or the elbow of the L shape.

Q.25 Explain evaluation metrics : Adjusted rand index.

Ans. : • The adjusted rand index measures the similarity between the original class partitioning (Y) and the clustering.

- If total number of samples in the dataset is n, the rand index is defined as :

$$R = \frac{a+b}{\binom{n}{2}}$$

- Rand index is defined as the number of pairs of objects that are either in the same group or in different groups in both partitions divided by the total number of pairs of objects.
- The Rand index lies between 0 and 1.
- When two partitions agree perfectly, the Rand index achieves the maximum value 1.
- A problem with Rand index is that the expected value of the Rand index between two random partitions is not a constant.
- This problem is corrected by the adjusted Rand index that assumes the generalized hyper-geometric distribution as the model of randomness.
- The adjusted Rand index has the maximum value 1, and its expected value is 0 in the case of random clusters.
- A larger adjusted Rand index means a higher agreement between two partitions. The adjusted Rand index is recommended for measuring agreement even when the partitions compared have different numbers of clusters.

Q.26 Explain evaluation metrics : Silhouette.

Ans. : • Silhouette refers to a method of interpretation and validation of clusters of data.

- Silhouettes are a general graphical aid for interpretation and validation of cluster analysis. This technique is available through the silhouette function. In order to calculate silhouettes, two types of data are needed :

1. The collection of all distances between objects. These distances are obtained from application of dist function on the coordinates of the elements in mat with argument method.
2. The partition obtained by the application of a clustering technique.

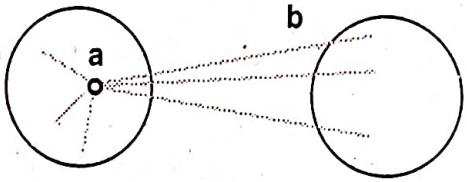
- For each element, a silhouette value is calculated and evaluates the degree of confidence in the assignment of the element :
  1. Well-clustered elements have a score near 1.
  2. Poorly-clustered elements have a score near – 1.
- Thus, silhouettes indicate the objects that are well or poorly clustered. Silhouette coefficient combines ideas of both cohesion and separation, but for individual points, as well as clusters and clustering's.
- For an individual point, I

  a = Average distance of i to the points in the same cluster

  b = min (average distance of i to points in another cluster)

Silhouette coefficient of i :

s = 1 – a/b if a < b



Silhouette coefficient

- **Cohesion :** Measures how closely related are objects in a cluster.
- **Separation :** Measure how distinct or well-separated a cluster is from other clusters.

Q.27 Explain evaluation methods for clustering algorithms.

☞ [SPPU : May-19, End Sem, Marks 4]

Ans. : • Since in every classification evaluation metric either it is a confusion matrix or log loss there is a need for a dependent variable or target variable.

- There are three commonly used evaluation metrics : Silhouette score, Calinski Harabaz index, Davies-Bouldin Index.
- The Calinski Harabaz index is based on the principle of variance ratio. This ratio is calculated between two parameters within-cluster diffusion and between cluster dispersion. The higher the index the better is clustering.
- Davies Bouldin index is based on the principle of with-cluster and between cluster distances. It is commonly used for deciding the number of clusters in which the data points should be labeled. It is different from the other two as the value of this index should be small. So the main motive is to decrease the DB index.
- Also refer Q.26.

END...

Unit VI

6 Introduction to Neural Networks

6.1 : Artificial Neural Networks

Q.1 What is artificial neural network ? List its characteristics.

Ans. : • Artificial Neural Network (ANN) is a computational system inspired by the structure, processing method, learning ability of a biological brain. An artificial neural network is composed of many artificial neurons that are linked together according to specific network architecture. The objective of the neural network is to transform the inputs into meaningful outputs.

- ANNs do not execute programmed instructions; they respond in parallel to the pattern of inputs presented to it. There are also no separate memory addresses for storing data. Instead, information is contained in the overall activation 'state' of the network. 'Knowledge' is thus represented by the network itself, which is quite literally more than the sum of its individual components.
- Fig Q.1.1 shows artificial neural network.

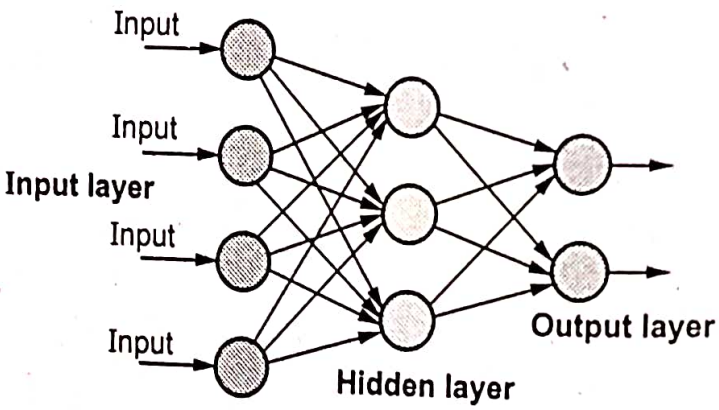


Fig. Q.1.1 Artificial neural network

- Elements of ANN are processing units, topology and learning algorithm.

- Tasks to be solved by artificial neural networks :
  1. Controlling the movements of a robot based on self-perception and other information;
  2. Deciding the category of potential food items in an artificial world;
  3. Recognizing a visual object;
  4. Predicting where a moving object goes, when a robot wants to catch it.
- Characteristics of artificial neural networks
  1. Large number of very simple processing neuron-like processing elements.
  2. Large number of weighted connections between the elements.
  3. Distributed representation of knowledge over the connections.
  4. Knowledge is acquired by network through a learning process.

Q.2 What is biological neuron ? Explain with diagram and its components.

Ans. : • Biological neurons, consisting of a cell body, axons, dendrites and synapses, are able to process and transmit neural activation.

- Artificial neural systems are inspired by biological neural systems. The elementary building block of biological neural systems is the neuron.
- The brain is a collection of about 10 billion interconnected neurons. Each neuron is a cell [right] that uses biochemical reactions to receive, process and transmit information. Fig. Q.2.1 shows biological neural systems.

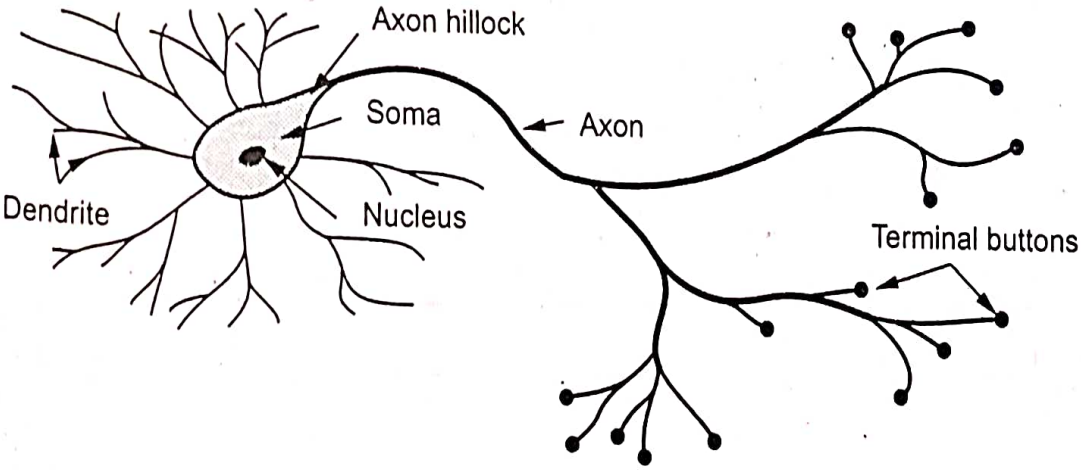


Fig. Q.2.1 Schematic of biological neuron

- The single cell neuron consists of the cell body or soma, the dendrites and the axon. The dendrites receive signals from the axons of other neurons. The small space between the axon of one neuron and the dendrite of another is the synapse. The afferent dendrites conduct impulses toward the soma. The efferent axon conducts impulses away from the soma.

Basic Components of Biological Neurons

1. The majority of ***neurons*** encode their activations or outputs as a series of brief electrical pulses (i.e. spikes or action potentials).
2. The neuron's ***cell body (soma)*** processes the incoming activations and converts them into output activations.
3. The neuron's ***nucleus*** contains the genetic material in the form of DNA. This exists in most types of cells, not just neurons.
4. ***Dendrites*** are fibres which emanate from the cell body and provide the receptive zones that receive activation from other neurons.
5. ***Axons*** are fibres acting as transmission lines that send activation to other neurons.
6. The junctions that allow signal transmission between the axons and dendrites are called ***synapses***. The process of transmission is by diffusion of chemicals called ***neurotransmitters*** across the synaptic cleft.

Q.3 Compare biological NN with artificial NN.

Ans. : Comparison between Biological NN and Artificial NN

Biological NN	Artificial NN
Soma	Unit
Axon, dendrite	Dendrite
Synapse	Weight
Potential	Weighted Sum
Threshold	Bias Weight
Signal	Activation

Q.4 Explain advantages and application of neural network.

Ans. : Advantages of neural network

- The advantages of neural networks are due to its adaptive and generalization ability.

a) Neural networks are adaptive methods that can learn without any prior assumption of the underlying data.

b) Neural network, namely the feed forward multilayer perception and radial basis function network have been proven to be universal functional approximations.

c) Neural networks are non-linear model with good generalization ability.

Applications of Neural Network

- Neural network applications can be grouped in following categories :

1. **Clustering :** A clustering algorithm explores the similarity between patterns and places similar patterns in a cluster. Best known applications include data compression and data mining.
2. **Classification/Pattern recognition :** The task of pattern recognition is to assign an input pattern (like handwritten symbol) to one of many classes. This category includes algorithmic implementations such as associative memory.
3. **Function approximation :** The tasks of function approximation is to find an estimate of the unknown function f() subject to noise. Various engineering and scientific disciplines require function approximation.
4. **Prediction/Dynamical systems :** The task is to forecast some future values of a time-sequenced data. Prediciton has a significant impact on decision support systems. Prediction differs from function approximation by considering time factor.

Q.5 What are the characteristics of ANN ?

Ans. : Characteristics of Artificial Neural Networks :

1. Large number of very simple processing neuron-like processing elements.
2. Large number of weighted connections between the elements.
3. Distributed representation of knowledge over the connections.
4. Knowledge is acquired by network through a learning process.

Q.6 Explain with example, the challenges in assigning synaptic weights for the interconnection between neurons ? How can this challenge be addressed ?

Ans. : • Artificial neural networks are specifically designed for a particular function like binary classification, multi class classification, pattern recognition and so on through learning processes. The weights of the synaptic connections of both the neural networks adjust with the learning process.

- Each connection between two neurons has a unique synapse with a unique weight attached to it. When we talk about updating weights in a network, we're really talking about adjusting the weights on these synapses.
- Weights are values that control the strength of the connection between two neurons. That is, inputs are typically multiplied by weights and that defines how much influence the input will have on the output.
- In other words, when the inputs are transmitted between neurons, the weights are applied to the inputs along with an additional value (the bias).
- The connection is controlled by the strength or amplitude of a connection between both nodes, also called the synaptic weight. Multiple synapses can connect the same neurons, with each synapse having a different level of influence (trigger) on whether that neuron is "fired" and activates the next neuron.
- the synapse is represented as a weight vector. The weights control how the output of the previous layer is fed through the activation function for each node. Using the convenient scalability of linear algebra, all the weight vectors of the synapses for an entire node layer can be represented as a single weight matrix.
- The resulting output of all these nodes is used as the input for the next layer of nodes and so on throughout the neural network "brain" until the final output layer is reached.

- The synapse is defined by these weights and expressed algebraically (for a linear synapse) as :

$$y_i = \sum_i w_{ij} x_i \quad \text{or} \quad y = wx$$

- Typically a bias term will be added and the output values will be passed through a choice of activation function before being used as inputs into the subsequent layer

Q.7 Explain types of Artificial Neural Networks.

Ans. : • Artificial neural networks (ANN) come in a variety of forms, and they all carry out tasks in a way that is comparable to how human brain neuron and network functions. Most artificial neural networks will share certain characteristics with a biological counterpart that is more complicated, and they are quite good at what they are meant to do segmentation or categorization, as examples.

1. **Feedback ANN :** In this kind of ANN, the output loops back into the network to achieve the best internally evolved results. The feedback networks are excellent for addressing optimization problems because they feed information back into themselves. Utilizing feedback ANNs, the internal system error repairs.

2. **Feed-Forward ANN :** A feed-forward network is a type of neural network that consists of at least one layer of neurons as well as input and output layers. The network's intensity can be observed based on the collective behaviour of the connected neurons, and the output is chosen by evaluating the network's output in the context of its input. The main benefit of this network is that it learns to assess and identify input patterns.

Q.8 Explain McCulloch Pitts neuron with diagram.

Ans. : • The first mathematical model of a biological neuron was presented by McCulloch and Pitts. This model is known as McCulloch Pitt model. It is basic building block of neural network.

- Directed weight graph is used for connecting neurons.
- McCulloch and Pitts describe a neuron as a logical threshold element with two possible states. Such a threshold element has "N" input

channels and one output channel. An input channel is either active (input 1) or silent (input 0).

- The activity states of all input channels thus encode the input information as a binary sequence of N bits. The state of the threshold element is then given by linear summation of all a different input signals x_i and comparison of the sum with a threshold value.s.
- The system of neurons is static and acts synchronously. A processor (system) with multiple inputs and a single output.
- Effective input : Weighted sum of all inputs.
- Bias or threshold : If the effective input is larger than the bias, the neuron outputs a one, otherwise, it outputs a zero.
- Fig. Q.8.1 shows McCulloch Pitt model.

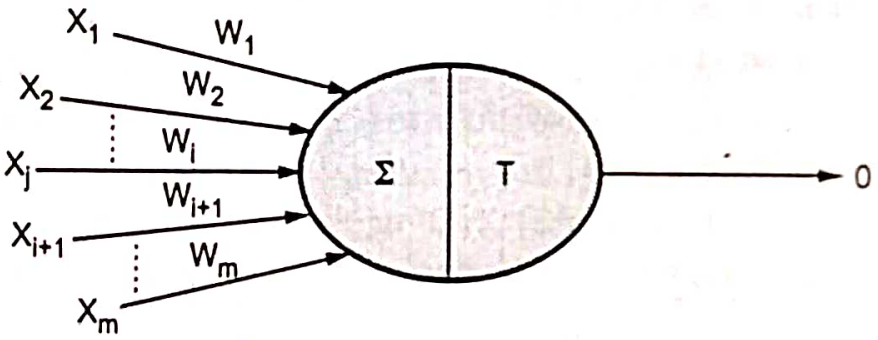


Fig. Q.8.1

- This model can be described in a mathematical formalism as follows :

$$0 = \theta(a)$$

Where, $a = \sum W_j X_j - T$

And

$\theta(x)$ is a function such that $\theta(x) = 1$ if $x > 0$, otherwise $\theta(x) = 1$.

- The parameters used to scale the inputs are called the weights. The effective input is the weighted sum of the inputs. The parameter to measure the switching level is the threshold or bias. Neuron fires (output of one) when its net input excitation exceeds a certain value called 'threshold.' Threshold is the minimum value of the sum of the weighted active inputs needed for the postsynaptic neuron to fire.

- The function for producing the final output is called the activation function, which is the step function in the McCulloch-Pitts model.

$$0 = f\left(\sum_{j=1}^{N} W_j X_j - T\right)$$

$$f(u) = \begin{cases} 1 & \text{if } u \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

- Their "neurons" operated under the following assumptions :
  1. They are binary devices (0,1) .
  2. Each neuron has a fixed threshold (theta).
  3. The neuron receives inputs from excitatory synapses, all having identical weights.
  4. Inhibitory inputs have an absolute veto power over any excitatory inputs.
  5. At each time step the neurons are simultaneously (synchronously) updated by summing the weighted excitatory inputs and setting the output to 1 iff the sum is greater than or equal to the threshold AND if the neuron receives no inhibitory input.

6.2 : Multilayer Perceptron

Q.9 What is perceptron ?

Ans. : • An arrangement of one input layer of McCulloch-Pitts neurons feeding forward to one output layer of McCulloch-Pitts neurons is known as a perceptron.

- The perceptron is a feed - forward network with one output neuron that learns a separating hyper - plane in a pattern space. Fig. Q.9.1 shows perceptron.
- Perceptron is an Artificial Neuron. It is the simplest possible Neural Network. The original Perceptron was designed to take a number of binary inputs, and produce one binary output (0 or 1).

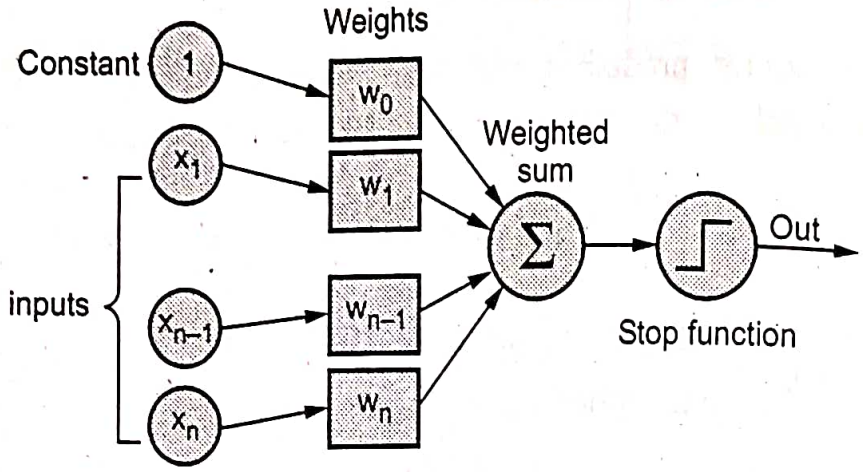


Fig. Q.9.1 Perceptron

- The idea was to use different weights to represent the importance of each input, and that the sum of the values should be greater than a threshold value before making a decision like true or false (0 or 1).

Q.10 Briefly discuss multilayer perceptron NN.

Ans. : • Multilayer perceptron is a neural network that learns the relationship between linear and non-linear data. Multilayer Perceptron has input and output layers, and one or more hidden layers with many neurons stacked together.

- Multilayer perceptron falls under the category of feed-forward algorithms, because inputs are combined with the initial weights in a weighted sum and subjected to the activation function.
- Fig. Q.10.1 shows Multilayer perceptron NN.
- Each layer is feeding the next one with the result of their computation, their internal representation of the data. This goes all the way through the hidden layers to the output layer.
- Multilayer perceptron is a typical example of a feedforward artificial neural network.
- The number of layers and the number of neurons are referred to as hyper-parameters of a neural network, and these need tuning. Cross-validation techniques must be used to find ideal values for these.
- The weight adjustment training is done via back-propagation. Deeper neural networks are better at processing data. However, deeper layers

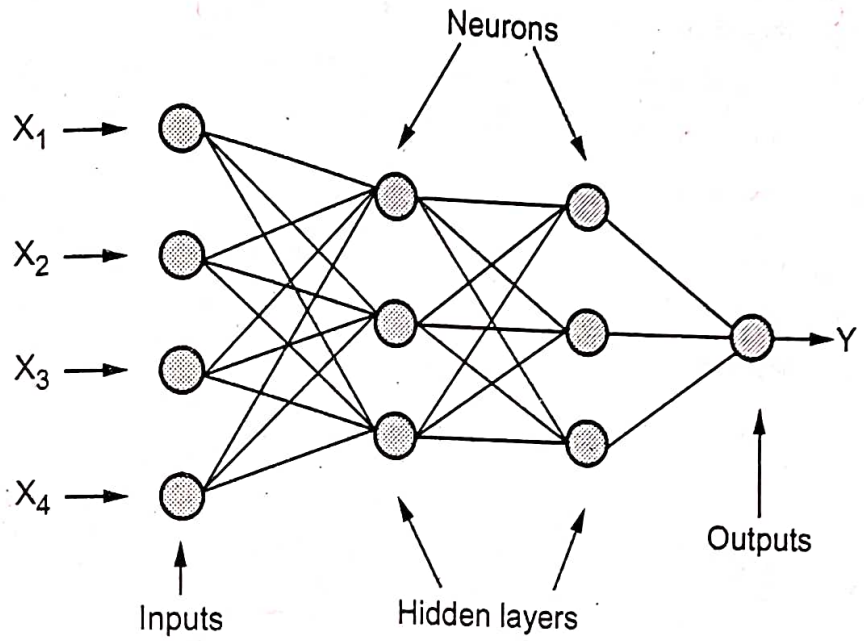


Fig. Q.10.1 Multilayer perceptron NN

can lead to vanishing gradient problems. Special algorithms are required to solve this issue.

- The multilayer perceptron learning procedure is as follows :
  a) Starting with the input layer, propagate data forward to the output layer. This step is the forward propagation.
  b) Based on the output, calculate the error. The error needs to be minimized.
  c) Backpropagate the error. Find its derivative with respect to each weight in the network, and update the model.
  d) Repeat the three steps given above over multiple epochs to learn ideal weights.
  e) Finally, the output is taken via a threshold function to obtain the predicted class labels.

6.3 : Back Propagation Learning

Q.11 Explain back-propagation neural networks.

Ans. : • Backpropagation is a training method used for a multi-layer neural network. It is also called the generalized delta rule. It is a gradient descent method which minimizes the total squared error of the output computed by the net.

- The backpropagation algorithm looks for the minimum value of the error function in weight space using a technique called the delta rule or gradient descent. The weights that minimize the error function is then considered to be a solution to the learning problem.
- Backpropagation is a systematic method for training multiple layer ANN. It is a generalization of Widrow-Hoff error correction rule. 80 % of ANN applications uses backpropagation.
- Fig. Q.11.1 shows backpropagation network.

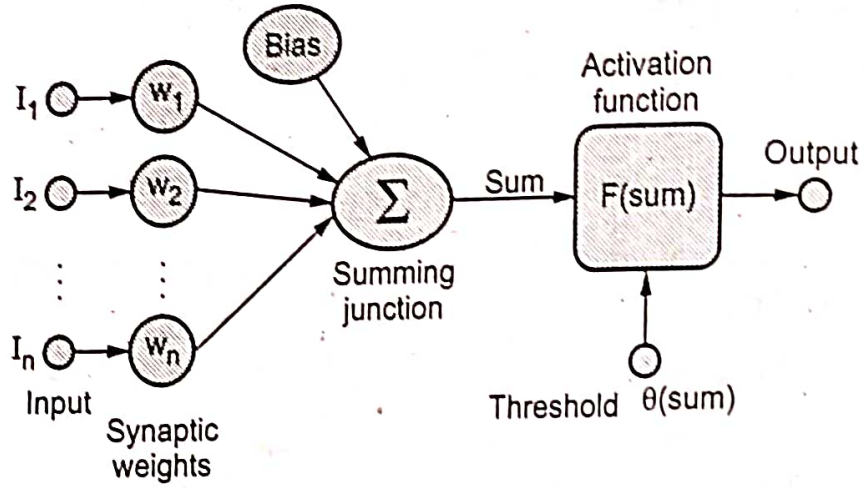


Fig. Q.11.1 Backpropagation network

- Consider a simple neuron :
  a. Neuron has a summing junction and activation function.
  b. Any non linear function which differentiable everywhere and increases everywhere with sum can be used as activation function.
  c. Examples : Logistic function, Arc tangent function, Hyperbolic tangent activation function.
- These activation function makes the multilayer network to have greater representational power than single layer network only when non-linearity is introduced.
- **Weights** connects unit (neuron) in one layer only to those in the next higher layer. The output of the unit is scaled by the value of the connecting weight and it is fed forward to provide a portion of the activation for the units in the next higher layer.

- Backpropagation can be applied to an artificial neural network with any number of hidden layers. The training objective is to adjust the weights so that the application of a set of inputs produces the desired outputs.

Q.12 Which factors are influencing backpropagation training ?

Ans. : Factors influencing backpropagation training :

- The training time can be reduced by using :

1. **Bias :** Networks with biases can represent relationships between inputs and outputs more easily than networks without biases. Adding a bias to each neuron is usually desirable to offset the origin of the activation function. The weight of the bias is trainable similar to weight except that the input is always +1.
2. **Momentum :** The use of momentum enhances the stability of the training process. Momentum is used to keep the training process going in the same general. In backpropagation with momentum, the weight change is a combination of the current gradient and the previous gradient.

Q.13 Explain advantages and disadvantages of backpropagation.

Ans. : Advantages of backpropagation

1. It is simple, fast and easy to program.
2. Only numbers of the input are tuned and not any other parameter.
3. No need to have prior knowledge about the network.
4. It is flexible.
5. A standard approach and works efficiently.
6. It does not require the user to learn special functions.

Disadvantages of backpropagation

1. Backpropagation possibly be sensitive to noisy data and irregularity.
2. The performance of this is highly reliant on the input data.
3. Needs excessive time for training.
4. The need for a matrix-based method for backpropagation instead of mini-batch.

Q.14 What is need of hidden layer in backpropagation ? How hidden units are selected ?

Ans. : • Need of hidden layers

1. A network with only two layers (input and output) can only represent the input with whatever representation already exists in the input data.
2. If the data is discontinuous or non-linearly separable, the innate representation is inconsistent and the mapping cannot be learned using two layers (Input and Output).
3. Therefore, hidden layers(s) are used between input and output layers.

- **Selection of number of hidden units :** The number of hidden units depends on the number of input units.
  1. Never choose h to be more than twice the number of input units.
  2. You can load p patterns of 1 elements into $\log_2 p$ hidden units.
  3. Ensure that we must have at least 1/e times as many training examples.
  4. Features extraction requires fewer hidden units than inputs.
  5. Learning many examples of disjointed inputs requires more hidden units than inputs.
  6. The number of hidden units required for a classification task increases with the number of classes in the task. Large network require longer training times.

6.4 : Functional Link Artificial Neural Network

Q.15 What is Functional Link Artificial Neural Network ? Explain architecture of FLANN.

Ans. : • A key paradigm for categorising patterns or simulating complex nonlinear process dynamics is the neural network (NN).

- These characteristics show that NN exhibit some intelligent behaviour and make them suitable candidates for modelling nonlinear phenomena, for which there is no ideal mathematical model.
- Neural network methods include multilayer perceptrons (MLP), radial basis functions (RBF), support vector machines (SVM) and others.

- These models have higher computing costs but better prediction competency. These models typically have significant computational costs since hidden layers are available.
- Polynomial perceptron networks (PPN) are one type of structure that helps to reduce the cost of computing shown in the Fig. Q.15.1.

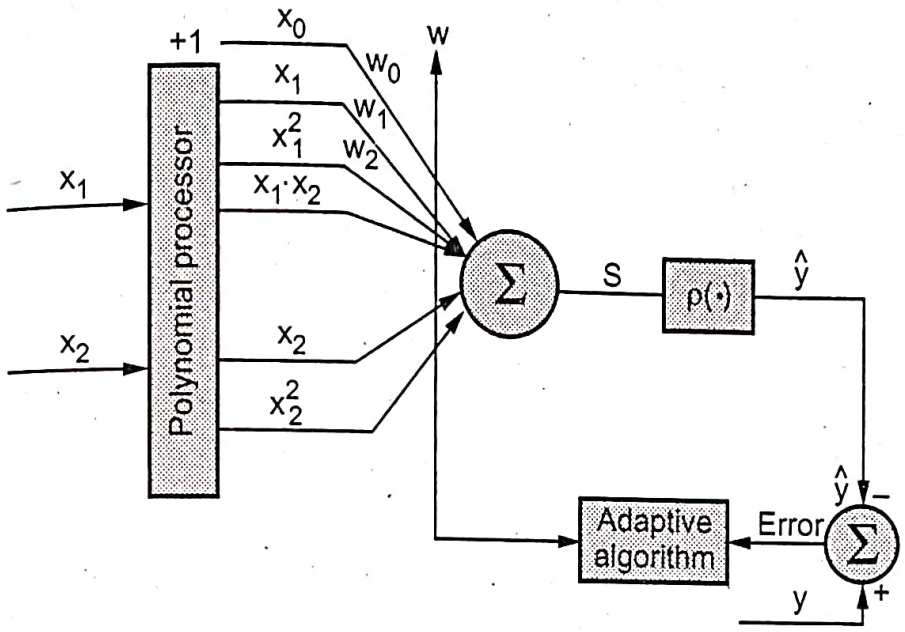


Fig. Q.15.1 PPN structure

- In general, single-layer ANN structures with a greater rate of convergence and less computational load than MLP structures were used to create functional link-based neural network models.

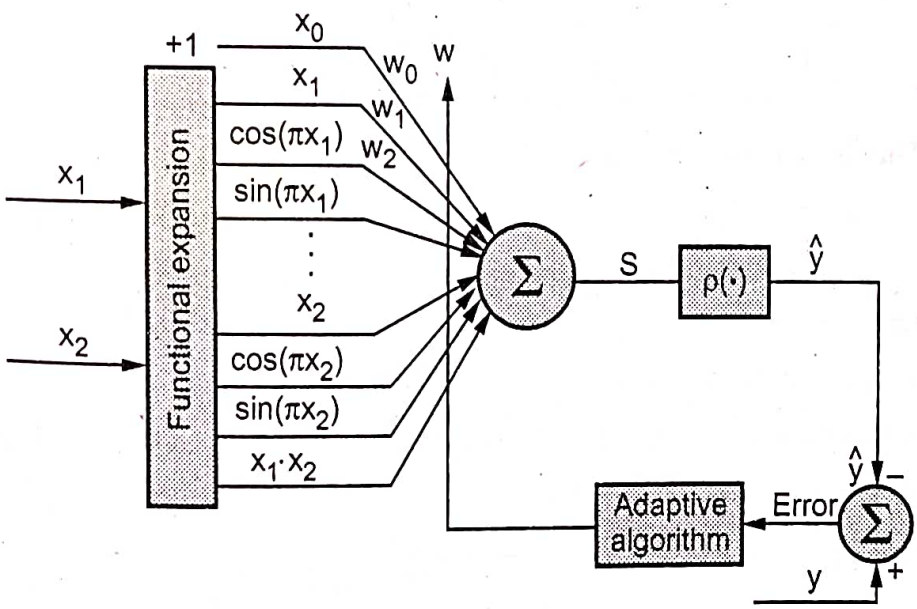


Fig. Q.15.2 FLANN structure

- FLANN eliminates the hidden layers. Due to its single-layer construction, the FLANN structure offers less computational complexity and faster convergence than MLP. Fig. Q.15.2 illustrates the FLANN structure.
- The original pattern and its outer products, as well as a subset of orthogonal sin and cos basis functions, are used in this case by the functional expansion block.

FLANN architecture

- Single layer neural networks can be thought of as an alternate strategy to get over the difficulties that come with multi-layer neural networks. However, due to its linear character, the single layer neural network frequently fails to map large nonlinear issues.
- Data mining's categorization task is incredibly nonlinear in design. Therefore, it is virtually difficult to solve such issues with a single layer feedforward artificial neural network.
- The FLANN architecture is proposed to close the gap between the linearity in the single layer neural network and the extremely complicated and computation-intensive multilayer neural network.
- The FLANN architecture employs a single layer feedforward neural network and functionally increases the input vector to get around linear mapping.

6.5 : Radial Basis Function Network

Q.16 What is Radial Basis Function network ?

Ans. : • Radial Basis Function (RBF) network is an artificial neural network that uses radial basis functions as activation functions. The output of the network is a linear combination of radial basis functions of the inputs and neuron parameters.

- RBF networks form a special class of neural networks, which consist of three layers.
- The input layer is used only to connect the network to its environment.

- The hidden layer contains a number of nodes, which apply a nonlinear transformation to the input variables, using a radial basis function, such as the Gaussian function, the thin plate spline function etc.
- The output layers is linear and serves as a summation unit.

Q.17 List the features of Radial Basis Function network.

Ans. : Features are as follows :

1. They are two-layer feed-forward networks.
2. The hidden nodes implement a set of radial basis functions (e.g. Gaussian functions).
3. The output nodes implement linear summation functions as in an MLP.
4. The network training is divided into two stages : first the weights from the input to hidden layer are determined and then the weights from the hidden to output layer.
5. The training / learning is very fast.
6. The networks are very good at interpolation.

Q.18 Compare RBF network and multilayer preceptron.

Ans. :

Sr. No.	RBF Networks	Multilayer Preceptrons
1.	An RBFN has a single hidden layer.	MLP may have one or more hidden layers.
2.	Hidden layer is nonlinear and output layer is linear.	Hidden and output layer used as pattern classifier are usually nonlinear.
3.	The argument of the activation function of each hidden unit computes the Eucliden norm between the input vector and the center of that unit.	The activation function of each hidden unit computes the inner product of the input vector and the synaptic weight vector of that unit.

4.	RBF networks using exponentially decaying localized nonlinearities construct local approximations to nonlinear input-output mappings.	MLPs construct global approximations to nonlinear input-output mapping.
5.	Computation nodes in the hidden layer of an RBF network are quite different and serve a different purpose from those in the output layer of the network.	Computation nodes of an MLP, located in a hidden or an output layer, share a common neuronal model.

6.6 : Activation Functions

Q.19 What is activation function ? Also explain logistic function and Arc tangent.

Ans. : • Activation functions also known as transfer function is used to map input nodes to output nodes in certain fashion.

- The activation function is the most important factor in a neural network which decided whether or not a neuron will be activated or not and transferred to the next layer.
- Activation functions help in normalizing the output between 0 to 1 or – 1 to 1. It helps in the process of backpropagation due to their differentiable property. During backpropagation, loss function gets updated, and activation function helps the gradient descent curves to achieve their local minima.
- Activation function basically decides in any neural network that given input or receiving information is relevant or it is irrelevant.
- These activation function makes the multilayer network to have greater representational power than single layer network only when non-linearity is introduced.

- The input to the activation function is sum which is defined by the following equation.

$$\text{sum} = I_1 W_1 + I_2 W_2 + \ldots + I_n W_n = \sum_{j=1}^{n} I_j W_j + b$$

- **Activation Function : Logistic Function**

$$f(\text{sum}) = \frac{1}{(1 + e^{-s * \text{sum}})} = (1 + s^{-s \times \text{sum}})^{-1}$$

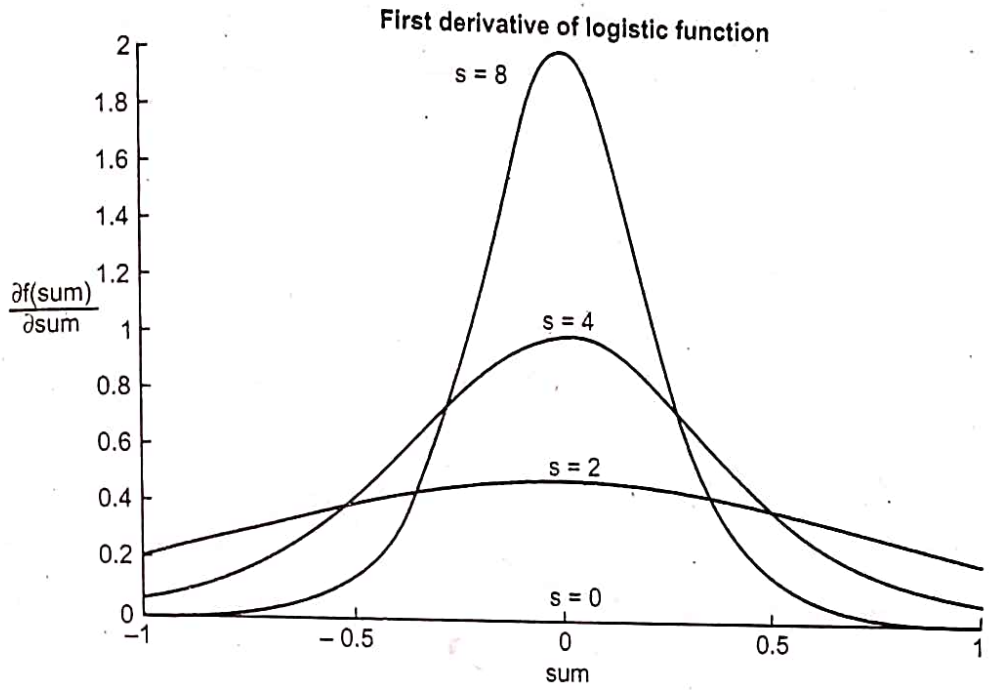


Fig. Q.19.1

- Logistic function monotonically increases from a lower limit (0 or – 1) to an upper limit (+1) as sum increases. In which values vary between 0 and 1, with a value of 0.5 when I is zero.

- **Activation Function : Arc Tangent**

$$f(sum) = \frac{2}{\pi}\tan^{-1}(s \times sum)$$

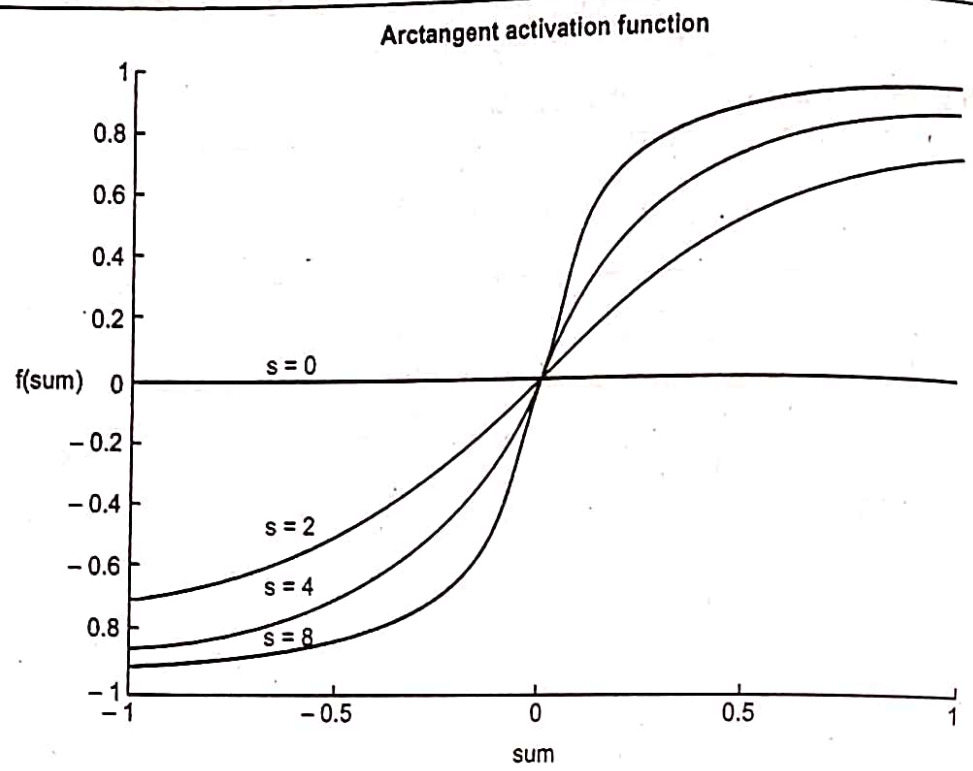


Fig. Q.19.2

Q.20 What are advantages and disadvantages of Sigmoid, Tanh and ReLU.

Ans. :

Function	Advantages	Disadvantages
Sigmoid	1. Output in range (0,1)	1. Saturated Neurons 2. Not zero entered 3. Small gradient 4. Vanishing gradient

Tanh	1. Zero centered, 2. Output in range (– 1,1)	1. Saturated Neurons
ReLU	1. Computational efficiency, 2. Accelerated convergence	1. Dead Neurons, 2. Not zero centered

Q.21 What is the function of a summation junction of a neuron ? What is threshold activation function ?

Ans. : • Summation junction for the input signals is weighted by the respective synaptic weight. Because it is a linear combiner or adder of the weighted input signals, the output of the summation junction (y) can be expressed as follows :

$$y_{sum} = \sum_{i=1}^{n} w_i \; x_i$$

- This summed function is applied over an Activation function. The output from this neuron is multiplied with the weight W_3 and supplied as input to the output layer.
- Fig. Q.21.1 shows summing function and activation function.

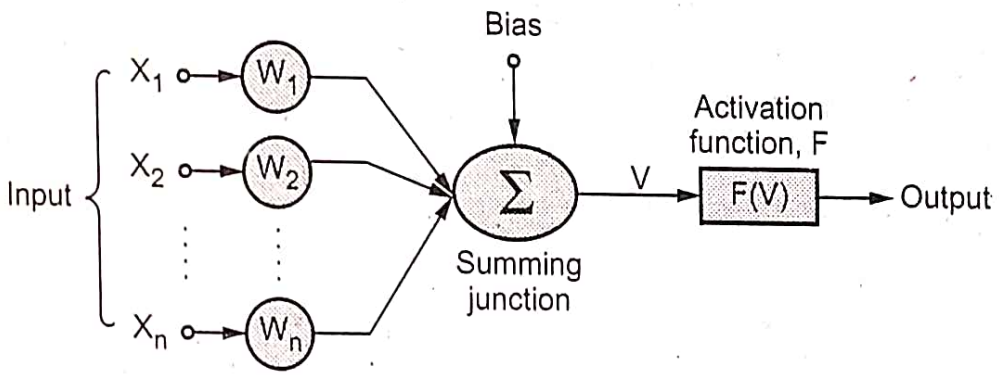


Fig. Q.21.1

- Threshold activation function also known as activation function, it results in an output signal only when an input signal exceeding a specific threshold value comes as an input. It is similar in behaviour to the biological neuron which transmits the signal only when the total input signal meets the firing threshold.

- Activation functions help in normalizing the output between 0 to 1 or – 1 to 1. It helps in the process of backpropagation due to their differentiable property. During backpropagation, loss function gets updated, and activation function helps the gradient descent curves to achieve their local minima.
- Activation function basically decides in any neural network that given input or receiving information is relevant or it is irrelevant.
- These activation function makes the multilayer network to have greater representational power than single layer network only when non-linearity is introduced.

6.7 : Introduction to Recurrent Neural Networks

Q.22 What is Recurrent Neural Networks ?

Ans. : • A class of neural networks for processing sequential data is known as Recurrent Neural Networks (RNN).

- RNN are neural networks that are specialized for processing a series of values x(1),..., x(τ), just like convolutional networks are neural networks that are specialized for processing a grid of values X, such as an image.
- RNNs are designed to recognize the sequential characteristics in data and use patterns to predict the next likely scenario. Unlike other neural networks, an RNN has an internal memory that enables it to remember historical input; this allows it to make decisions by considering current input alongside learning from previous input.

Q.23 Write short note on recurrent neural network.

Ans. : • A recurrent neural network is a type of neural network that contains loops, allowing information to be stored within the network.

- A RNN is particularly useful when a sequence of data is being processed to make a classification decision or regression estimate but it can also be used on non-sequential data. Recurrent neural networks are typically used to solve tasks related to time series data.

- Applications of recurrent neural networks include natural language processing, speech recognition, machine translation, character-level language modeling, image classification, image captioning, stock prediction, and financial engineering.
- Fig. Q.23.1 shows architecture of recurrent neural network.

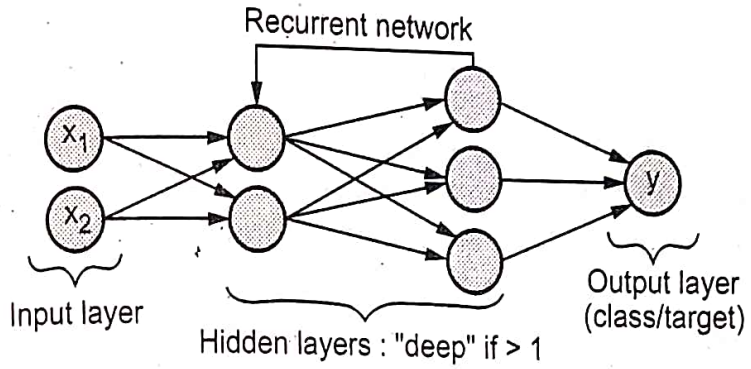


Fig. Q.23.1

- Recurrent Neural Networks can be thought of as a series of networks linked together. They often have a chain-like architecture, making them applicable for tasks such as speech recognition, language translation, etc.
- An RNN can be designed to operate across sequences of vectors in the input, output, or both. For example, a sequenced input may take a sentence as an input and output a positive or negative sentiment value. Alternatively, a sequenced output may take an image as an input, and produce a sentence as an output.

Q.24 Explain advantages and disadvantages of RNN.

Ans. : Advantages :

a) RNN can process inputs of any length.

b) RNN model is modeled to remember each information throughout the time which is very helpful in any time series predictor.

c) Even if the input size is larger, the model size does not increase.

d) The weights can be shared across the time steps.

e) RNN can use their internal memory for processing the arbitrary series of inputs which is not the case with feedforward neural networks.

Disadvantages :

a) Due to its recurrent nature, the computation is slow.

b) Training of RNN models can be difficult.

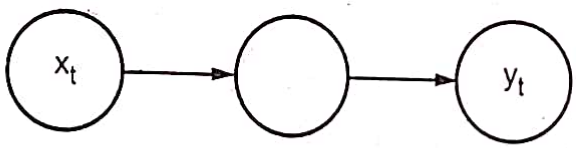
Prone to problems such as exploding and gradient vanishing.

Q.25 Explain types of Recurrent Neural Networks.

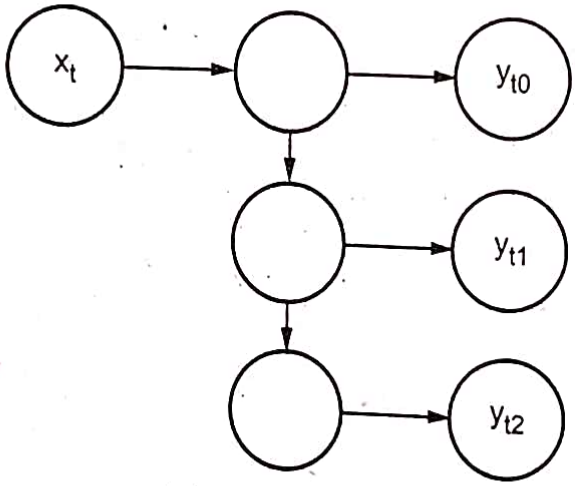
Ans. :

1. **One-to-one :** This neural network is used for fixed sized input to fixed sized output for example image classification. This was formerly known as Vanilla RNN, usually characterized by a single variety of input, such as a word or image. At the same time, the outputs are produced as a single token value. All traditional neural networks fall into this category.
2. **One-to-many :** A single input is used to create multiple outputs. A popular application for one to many is music generation.
3. **Many-to-one :** Consists of several inputs that used to create a single output. An example is sentiment analysis. Input is a movie's review (multiple words in input) and output is sentiment associated with the review.

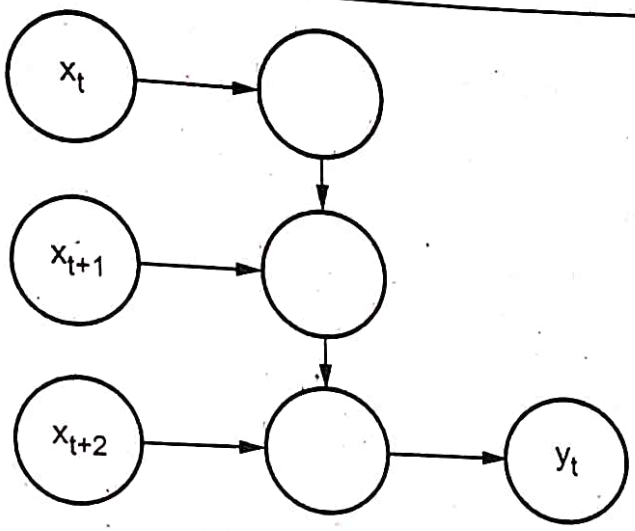
- **Many-to-many :** Several inputs are used for generating several outputs. Name entity recognition is a famous example of this category.
- Fig. Q.25.1 shows types of RNN.



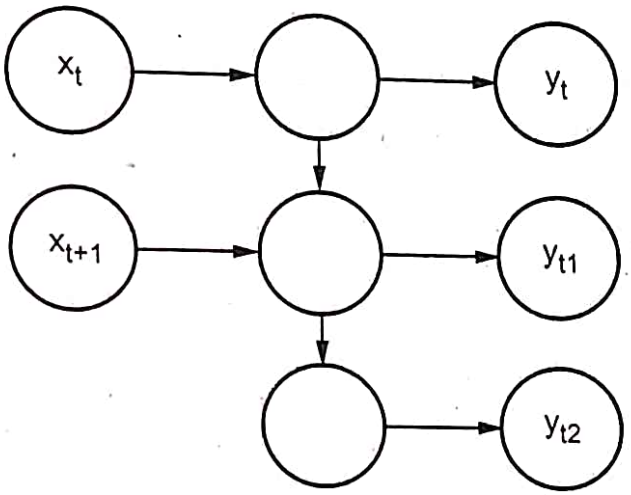
(a) One-to-one



(b) One-to-many



(c) Many-to-one



(d) Many-to-many

Fig. Q.25.1 Types of RNNs

6.8 : Convolutional Neural Networks

Q.26 What is Convolutional Neural Network ? Explain in detail.

Ans. : • Convolutional neural network (CNN) is a deep learning neural network designed for processing structured arrays of data such as images. A CNN is a feed-forward neural network, often with up to 20 or 30 layers. The power of a convolutional neural network comes from a special kind of layer called the convolutional layer.

- Convolutional neural network is also called ConvNet.
- In CNN, 'convolution' is referred to as the mathematical function. It's a type of linear operation in which you can multiply two functions to

create a third function that expresses how one function's shape can be changed by the other.

- In simple terms, two images that are represented in the form of two matrices, are multiplied to provide an output that is used to extract information from the image.
- CNN represents the input data in the form of multidimensional arrays. It works well for a large number of labeled data. CNN extract the each and every portion of input image, which is known as receptive field. It assigns weights for each neuron based on the significant role of the receptive field.
- Instead of preprocessing the data to derive features like textures and shapes, a CNN takes just the image's raw pixel data as input and "learns" how to extract these features, and ultimately infer what object they constitute.
- The goal of CNN is to reduce the images so that it would be easier to process without losing features that are valuable for accurate prediction.
- A convolutional neural network is made up of numerous layers, such as convolution layers, pooling layers and fully connected layers and it uses a back-propagation algorithm to learn spatial hierarchies of data automatically and adaptively.
- To understand the concept of Convolutional Neural Networks (CNNs), let us take an example of the images our brain can interpret.
- As soon as we see an image, our brain starts categorizing it based on the color, shape and sometimes also the message that image is conveying. Similar thing can be done through machines even after a rigorous training. but the difficulty is there is a huge difference in what humans interpret and what machine does. For a machine, the image is merely an array of pixels. There is a unique pattern included in each object present in the image and the computer tries to find out these patterns to get the information about the image.
- Machines can be trained giving tons of images to increase its ability to recognize the objects included in a given input image.

- Most of the digital companies have opted for CNNs for image recognition, some of these include Google, Amazon, Instagram, Pinterest, Facebook, etc.
- Hence, we define a convolutional neural network as : "A neural network consisting of multiple convolutional layers which are used mainly for image processing, classification, segmentation and other correlated data".

Q.27 Explain advantages and disadvantages of CNN.

Ans. : Advantages :

- CNN automatically detects the important features without any human supervision.
- CNN is also computationally efficient.
- Higher accuracy.
- Weight sharing is another major advantage of CNNs.
- Convolutional neural networks also minimize computation in comparison with a regular neural network.
- CNNs make use of the same knowledge across all image locations.

Disadvantages :

- Adversarial attacks are cases of feeding the network 'bad' examples to cause misclassification.
- CNN requires lot of training data.
- CNNs tend to be much slower because of operations like maxpool.

Q.28 Explain basic architecture of CNN.

Ans. : • Fig. Q.28.1 shows basic architecture of CNN.

- A convolutional neural network, as discussed above, has the following layers that are useful for various deep learning algorithms. Let us see the working of these layers taking an example of the image having dimension of 12 x 12 x 4. These are :

1. **Input layer :** This layer will accept the image of width 12, height 12 and depth 4.

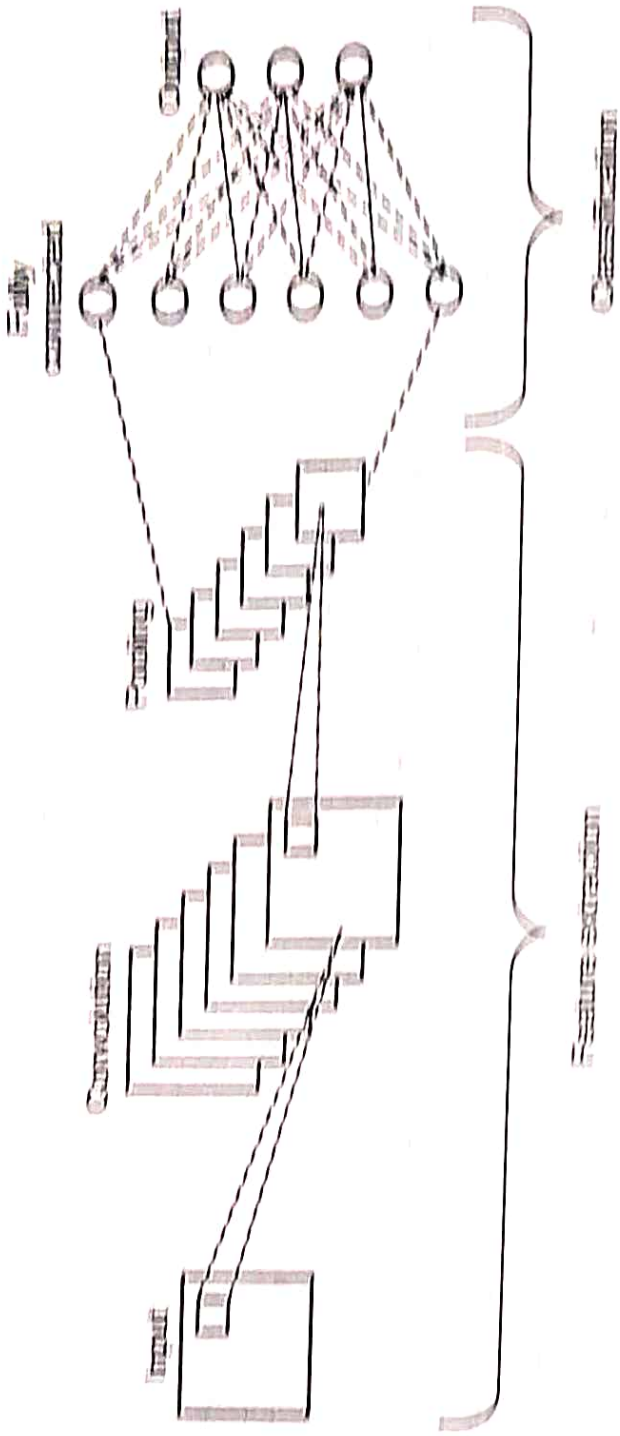


Fig. Q.28.1 basic architecture of CNN

2. **Convolution layer :** It computes the volume of the image by getting the dot product between the image filters possible and the image patch. For example, there are 10 filters possible, then the volume will be computed as 12 x 12 x 10.

3. **Activation function layer :** This layer applies activation function to each element in the output of the convolutional layer. Some of the well accepted activation functions are ReLu, Sigmoid, Tanh, Leaky ReLu, etc. These functions will not change the volume obtained at the convolutional layer and hence it will remain equal to $12 \times 12 \times 10$.

4. **Pool layer :** This function mainly reduces the volume of the intermediate outputs, which enables fast computation of the network model, thus preventing it from overfitting.

Q.29 What is difference between RNN and CNN ?

Ans. :

RNN	CNN
RNN is applicable for time series and sequential data.	CNN is applicable for sparse data like images.
RNN can have no restriction in length of inputs and outputs.	CNN has finite inputs and finite outputs.
RNN is primarily used for speech and text analysis.	CNN can be used for video and image processing.
RNN works on loops to handle sequential data.	CNN has a feedforward network.
While training the model, CNN uses a simple back-propagation.	While training the model, RNN uses back-propagation through time to calculate the loss.

END...

SOLVED MODEL QUESTION PAPER (End Sem)

Machine Learning

B.E. (Computer) Semester - VII [As Per 2019 Pattern]

Time : 2 $\frac{1}{2}$ Hours] **[Maximum Marks : 70**

N.B. : ***i) Attempt Q.1 or Q.2, Q.3 or Q.4, Q.5 or Q.6, Q.7 or Q.8.***

ii) Neat diagrams must be drawn wherever necessary.

iii) Figures to the right side indicate full marks.

iv) Assume suitable data, if necessary.

Q.1 a) *What do you mean by a linear regression ? Which applications are best modeled by linear regression ?* **(Refer Q.18 of Chapter - 3)** **[6]**

b) *Explain gradient descent algorithm. Also explain its limitation.* **(Refer Q.23 of Chapter - 3)** **[6]**

c) *What is bias in machine learning ? How to reduce the high bias ?* **(Refer Q.1 and Q.2 of Chapter - 3)** **[6]**

OR

Q.2 a) *What is overfitting and underfitting in machine learning model ? Explain with example.* **(Refer Q.6 of Chapter - 3)** **[6]**

b) *What do you mean by logistic regression ? Explain with example.* **(Refer Q.19 of Chapter - 3)** **[6]**

c) *How the performance of a regression function is measured ?* **(Refer Q.26 of Chapter - 3)** **[6]**

Q.3 a) *Explain kNN algorithm with its advantages and disadvantages.* **(Refer Q.2 of Chapter - 4)** **[6]**

b) *Explain ensemble learning.* **(Refer Q.9 of Chapter - 4)** **[6]**

c) *What is Random forest ? Explain working of random forest.* **(Refer Q.12 of Chapter - 4)** **[5]**

OR

Q.4 a) *What is cross - validation ? How it improves the accuracy of the outcome ?* **(Refer Q.28 of Chapter - 4)** **[5]**

b) *What do you mean by SVM ? Explain with example.* **(Refer Q.4 of Chapter - 4)** **[7]**

c) *Explain various multiclass classification techniques.* **(Refer Q.21 of Chapter - 4)** **[5]**

Q.5 a) *What is agglomerative clustering ? Explain with example.* **(Refer Q.17 of Chapter - 5)** **[8]**

b) *What is density-based clustering ? Explain its working.* **(Refer Q.19 of Chapter - 5)** **[6]**

c) *What is cluster analysis ? Explain desirable properties of clustering algorithm.* **(Refer Q.3 of Chapter - 5)** **[4]**

OR

Q.6 a) *Explain various types of clustering.* **(Refer Q.5 of Chapter - 5)** **[6]**

b) *Explain the difference between K-mean and K-medoids clustering.* **(Refer Q.10 of Chapter - 5)** **[6]**

c) *Write a note on clustering trees.* **(Refer Q.2 of Chapter - 5)** **[6]**

Q.7 a) *Explain back-propagation neural networks.* **(Refer Q.11 of Chapter - 6)** **[6]**

b) *What is Functional Link Artificial Neural Network ? Explain architecture of FLANN.* **(Refer Q.15 of Chapter - 6)** **[6]**

c) *What is biological neuron ? Explain with diagram and its components.* **(Refer Q.2 of Chapter - 6)** [5]

OR

Q.8 a) *What is activation function ? Also explain logistic function and Arc tangent.* **(Refer Q.19 of Chapter - 6)** [6]

b) *Explain McCulloch Pitts neuron with diagram.*
(Refer Q.8 of Chapter - 6) [8]

c) *Explain types of Artificial Neural Networks.*
(Refer Q.7 of Chapter - 6) [3]

END...