Unit III

CHAPTER 3

Classification and Clustering

Syllabus Topic : Classification Problems

3.1 Classification Problems

Q. 3.1.1 What is classification? What are the components of classification problem? (Ref. Sec. 3.1) (5 Marks)

- Classification problems are supervised learning methods. It is used to predict the target attribute.
- Classification application includes image and pattern recognition, medical diagnosis, loan approval, detecting faults and industry applications. Estimation and prediction are viewed as type of classification.
- Consider we have dataset N. It has x observations and y explanatory attributes and categorical target attribute.
- The explanatory attribute are termed as predictive variables. The target attribute is named as class or label. Observations are called as examples or instances.
- The purpose of classification model is to recognise recurring relationship between the predicted or explanatory variables. It describes the examples belonging to the same class.
- These relationships are interpreted into classification rules. It is used to predict the class of the three components of a classification problem: a generator of observations, a supervisor of the target class and a classification algorithm.

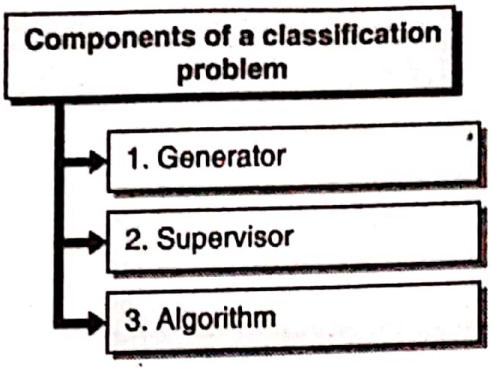


Fig. 3.1.1 : Components of a classification problem

➜ **1. Generator**

The role of the generator is to take out random vectors m of examples permitting to an unknown probability distribution Pm (m).

➜ **2. Supervisor**

The supervisor returns for each vector m of examples the value of the target class according to a conditional distribution is not known.

➜ **3. Algorithm**

A classification algorithm is called as classifier which chooses a function which helps to minimize loss of function.

3.1.1 Phases of Classification Model

Q. 3.1.2 What are the three phases of classification model ? (Ref. Sec. 3.1.1) (5 Marks)

The three main phases of classification model are as follows :

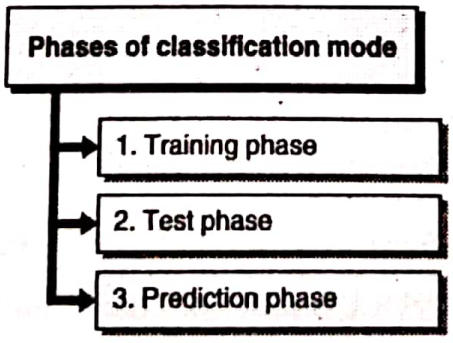


Fig. 3.1.2 : Phases of a classification model

➜ **1. Training phase**

The classification algorithm is applied to the subset of N which is called as training set. To derive classification rules it allow the corresponding target class z to be involved to each observation m.

➜ **2. Test phase**

The rules are generated during the training phase. It is used to classify the observations of N. It is not included in the training set, for which the target class value is already known. The training set and test set should be different.

➜ **3. Prediction phase**

A prediction is achieved by applying the rules generated during the training phase to the explanatory variables that describe the new instance.

3.1.2 Taxonomy of Classification Model

Q. 3.1.3 What are the main components of classification model ? (Ref. Sec. 3.1.2) (5 Marks)

There are four main components of classification model.

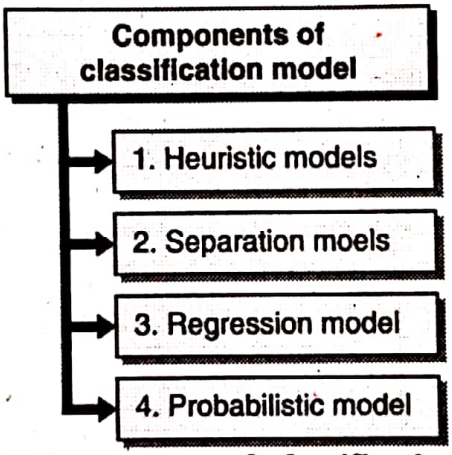


Fig. 3.1.3 : Components of classification model

➜ **1. Heuristic models**

- It includes nearest neighbour methods. It is based on the conception of distance between observations, and classification trees.
- Distance between observations and classification trees is used to divide-and-conquer schemes to derive groups of observations that are as homogeneous as possible with respect to the target class.

➜ **2. Separation model**

- The classification models which belongs to separation model category differ from each other with respect to the type of separation regions, loss function etc.
- The most popular separation techniques include discriminant analysis, perceptron methods, neural networks and support vector machines. Some variants of classification trees can also be placed in this category

➜ **3. Regression model**

It is the prediction of continuous target variables. It considers the functional form of the conditional probabilities, which correspond to the assignment of the target class by the supervisor.

➜ **4. Probabilistic models**

- In probabilistic models, a hypothesis is formulated regarding the observations given the target class, known as class-conditional probabilities.
- Subsequently, using Bayes' theorem, probabilities of the target class assigned by the supervisor.

Syllabus Topic : Evaluation of Classification Models

3.2 Evaluation of Classification Model

Q. 3.2.1 How you evaluate classification method? (Ref. Sec. 3.2) (5 Marks)

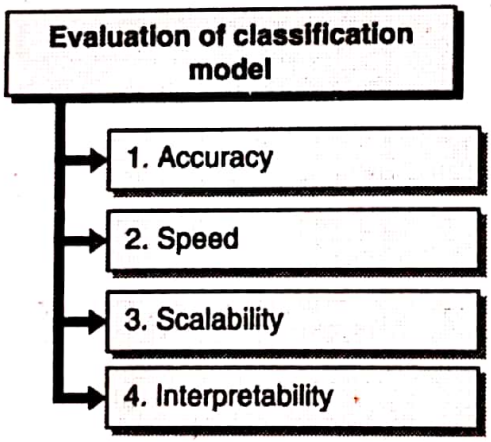


Fig. 3.2.1 : Evaluation of classification model

➜ **1. Accuracy**

The accuracy of a model is to forecast the target class for future observations. Based on accuracy values, it is possible to compare different models in order to select the classifier.

➜ **2. Speed**

- Classification methods characterized by computation times. It is applied to a small-size training set obtained from a large number of observations by selecting of random samplings.
- A classification method is strong if the classification rules generated, and corresponding accuracy, do not vary significantly as the choice of the training set. It is expected to handle missing data and outliers.

➜ **3. Scalability**

It is the ability of classifier to learn from large datasets.

➜ **4. Interpretability**

The objective of a classification analysis is to interpret as well as predict. The rules generated should be simple knowledge workers and experts in the application domain should understand it easily.

3.2.1 Holdout Method

Q. 3.2.2 Explain the Holdout method. (Ref. Sec. 3.2.1) (4 Marks)

- The holdout method reserves a certain amount of data set for testing and the remainder for training. Usually one third for testing, the rest for training .
- The holdout method offers an evaluation of the true error rate (accuracy) of a classifier. We have a (small) data sample of the whole data (population). Sampling is used to divide the data in test set and training set .
- That is why true error rate is difficult to calculate.

3.2.2 Repeated Random Sampling

Q. 3.2.3 Explain the Repeated random sampling. (Ref. Sec. 3.2.2) (4 Marks)

- In Holdout estimate, the process of repeating different subsamples make the method more reliable. In each iteration, a certain proportion is arbitrarily selected for training (possibly with stratification).
- The error rates (or some other performance measure) on the different iterations are averaged to produce an overall error rate.
- The disadvantage of repeated holdout method is that it is still not optimum. The different set may overlap.

☞ **Formula for repeated random sampling**

- There are m observations in two disjoint sets T and V. T is for training and V is for testing purpose. Repeated random sampling involves replicating the holdout method r number of times.
- For each repetition a sample T_k is extracted and corresponding accuracy is calculated T_k involves t observation where $V_k = D - T_k$

$$acc_A = acc_{AF} = \frac{1}{r} \sum_{k=1}^{r} acc_{AF}(V_K)$$

3.2.3 Cross-Validation

Q. 3.2.4 Explain the cross validation. (Ref. Sec. 3.2.3) (4 Marks)

- Cross validation evades overlapping test sets. It assures that each observation of dataset D appears the same number of times.
- The cross validation is based on dataset D. There are r disjoint subsets L_1, L_2, L_3...L_r and require r iterations. At j^{th} iteration L_j is selected as the test set and union of all other subsets in the partition as the training set.

$$V_j = L_j \quad T_j = \bigcup_{j \neq k} L_j$$

- Standard method for evaluation is ten fold cross validation. Extensive experiments have shown that 10 is the best choice to get accurate estimate.
- Repeated stratified cross validation even better. Ten fold cross validation repeated 10 times and results are averaged (reduces the variance). Leave one out is a particular form of cross validation. In this case m test sets include only one observation and each example in turn measure accuracy.

3.2.4 Confusion Matrices

Q. 3.2.5 Explain the confusion matrices. (Ref. Sec. 3.2.4) (5 Marks)

- A binary classifier produces output with two class values or labels, such as Yes/No and 1/0, for given input data. The class of interest is usually denoted as "positive" and the other as "negative".
- A test dataset is used for performance evaluation. It should hold the correct labels (observed labels) for all data instances. These labels are used to compare with the predicted labels for performance evaluation after classification.
- The predicted labels will be exactly the same if the performance of a binary classifier is perfect. but it is not common in practical situation.
- A binary classifier predicts all data instances of a test dataset as either positive or negative. This classification (or prediction) produces four outcomes - true positive, true negative, false positive and false negative.
- First two basic measures from the confusion matrix.
- Error rate (ERR) and accuracy (ACC) are the most common and intuitive measures derived from the confusion matrix.

☞ **Error rate**

- The best error rate is 0.0, whereas the worst is 1.0.

- Error rate is calculated as the total number of two incorrect predictions (FAN + FAP) divided by total number of dataset (F + N).

$$\text{Error rate} = \text{ERR} = \frac{\text{FAP} + \text{FAN}}{\text{TRP} + \text{TAN} + \text{FAN} + \text{FAP}} = \frac{\text{FAP} + \text{FAN}}{\text{P} + \text{N}}$$

☞ Accuracy

Accuracy is calculated as the number of all correct predictions divided by the total number of dataset. The best accuracy is 1.0 whereas the worst is 0.0. It can be calculated as, 1 – EPR.

$$\text{ACC} = = \frac{\text{TRP} + \text{TAN}}{\text{TRP} + \text{TAN} + \text{FAN} + \text{FAP}} = \frac{\text{TRP} + \text{TAN}}{\text{P} + \text{N}}$$

☞ True positive rate

True positive rate or sensitivity is calculated as the number of correct positive predictions divided by the total number of positives.

The best true positive rate is 1.0 and worst is 0.0.

$$\text{True positive rate} = \frac{\text{TRP}}{\text{TRP} + \text{FAN}}$$

☞ True negative rate or specificity

It is the number of correct negative predictions divided by the total number of negatives.

$$\text{SP} = \frac{\text{TAN}}{\text{TAN} + \text{FAP}}$$

☞ Precision

It is calculated as the total number of correct positive predictions divided by the total number of positive predictions. The best precision is 1.0 whereas the worst is 0.0.

$$\text{Precision} = \frac{\text{TRP}}{\text{TRP} + \text{FAP}}$$

☞ False positive rate

It is calculated as the number of incorrect positive predictions divided by the total number of negatives.

$$\text{False positive rate} = 1 - \text{Specificity}$$

$$\text{FPR} = \frac{\text{FAP}}{\text{TAN} + \text{FAP}} = 1 - \text{SP}$$

F score is harmonic mean of precision and recall.

$$F_B = \frac{(1 + \beta^2)\,(\text{PREC} \cdot \text{REC})}{(\beta^2 \cdot \text{PREC} + \text{REC})}$$

β is commonly 0.5, 1 or 2.

3.2.5 ROC Curve Charts

Q. 3.2.6 Explain the ROC curve chart. (Ref. Sec. 3.2.5) (5 Marks)

- Receiver Operating Characteristics plot measure is based on two basic evaluation measures - specificity and sensitivity. Specificity is a performance measure of the whole negative part of a dataset.
- Sensitivity is a performance measure of the whole positive part. Receiver Operating Characteristic (ROC) curve charts allow the user to visually evaluate the accuracy of a classifier.
- It is used to compare different classification models. They visually express the information content of a sequence of confusion matrices.
- It allow the ideal trade-off between the number of correctly classified positive observations and the number of incorrectly classified negative observations to be assessed. In this respect, they are an alternative to the assignment of misclassification costs.

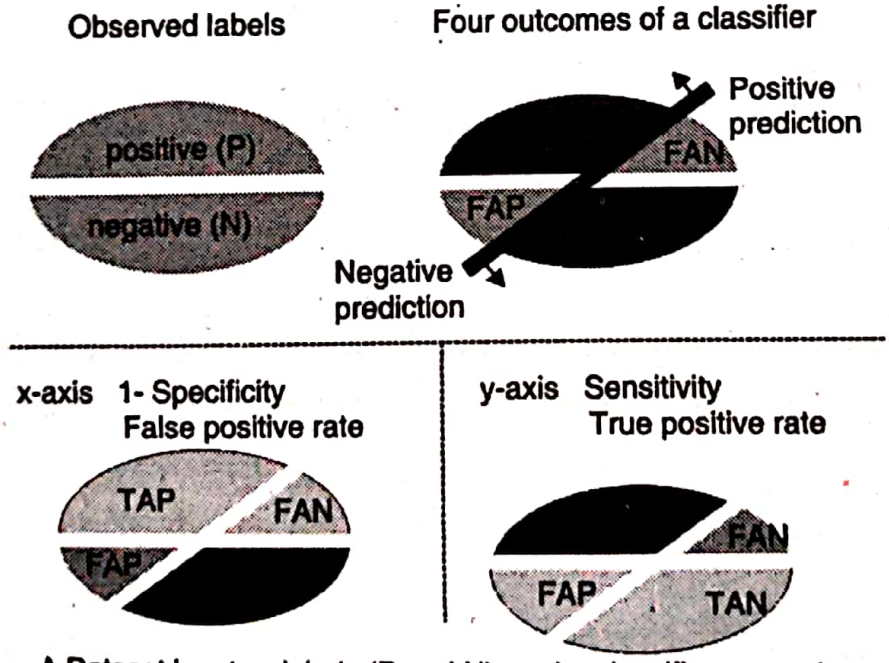
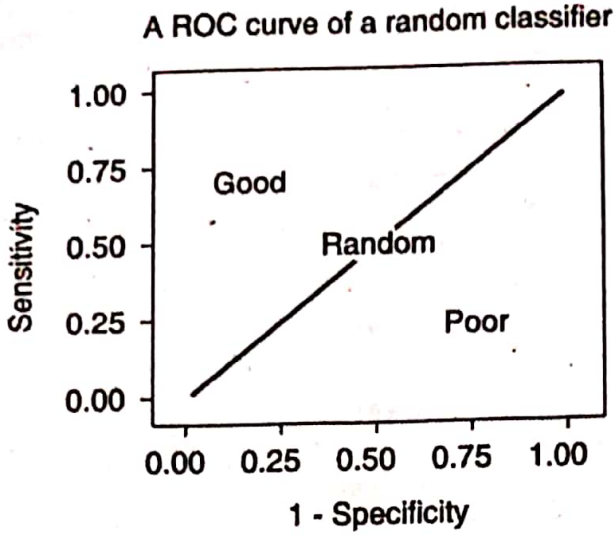


Fig. 3.2.2

- ROC curves with the top left corner area (0.0, 1.0) show good performance levels. ROC curves bottom right corner (1.0, 0.0) area indicate poor performance levels.



A ROC curve represents a classifier with the random performance level. The curve separates the space into two areas for good and poor performance levels.

Fig. 3.2.3

3.2.6 Cumulative Gain and Lift Charts

Q. 3.2.7 Explain the Cumulative gain and lift chart. (Ref. Sec. 3.2.6) (5 Marks)

- Gain or lift is the measure of the effectiveness of classification model. It is calculated as the ratio between the results obtained with or without model.
- It is visual aid for calculating performance of classification model. Both charts consist of lift curve and base line.
- For example, An educational institute wants to do mail marketing drive for new course. It costs institute 1rs for each item mailed. They have information of 1,00,000 students. Out of 1 lac 20000 students showed positive response.
- Suppose we use response model to assign score.
- Prediction of response model.

Cost	Total Number of Students Contacted	Positive Response
10000	10000	6000
20000	20000	10000
30000	30000	13000
40000	40000	15800

Cost	Total Number of Students Contacted	Positive Response
150000	50000	17000
60000	60000	18000
70000	70000	18800
80000	80000	19400
90000	90000	198000
1,00,000	1,00,000	20,000

☞ **Cumulative gain chart**

- The y axis shows the percentage of positive response and x axis shows the percentage of students contacted.
- Baseline – overall response rate-It means if institute contact n number of students then n number of students are positive.
- Lift curve-Using prediction of response model calculate the percentage of positive response for the percentage of the students contacted. e.g. [6000/20000]*100 = 30 %.

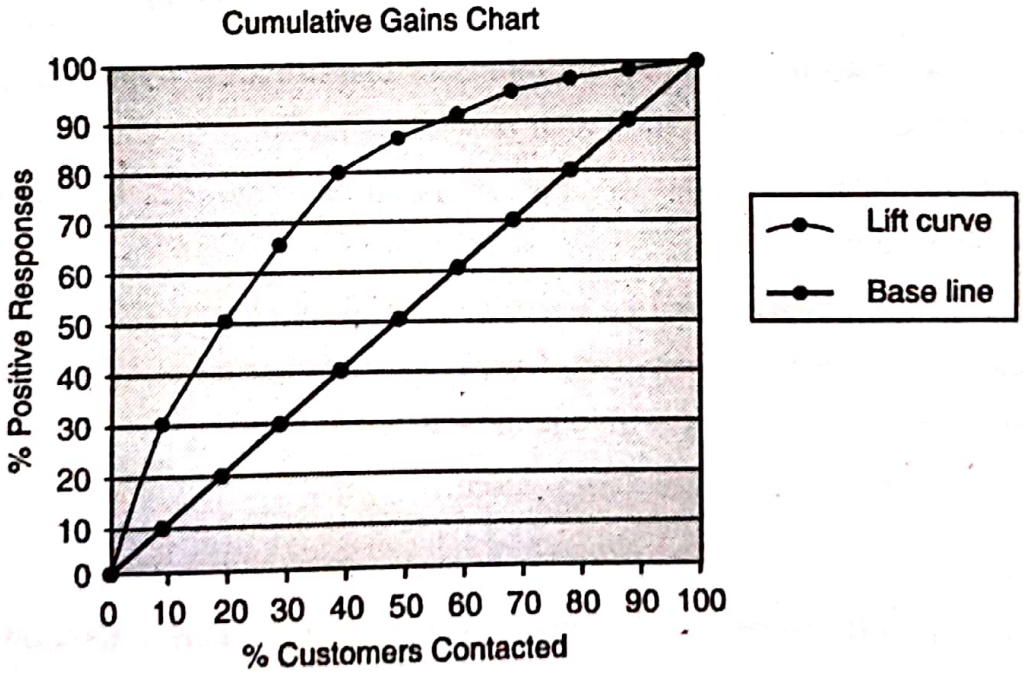


Fig. 3.2.4

☞ **Lift chart**

- It shows actual lift.

- For contacting 10% of students using no model we should get 10% of the responders and using model 30% of the responders so y value of the lift curve is 30/10 = 3. Similarly for 20% of students 50% of the responders so 50/20 = 2.5.
- The cumulative and lift chart gives an idea that which customers to contact.

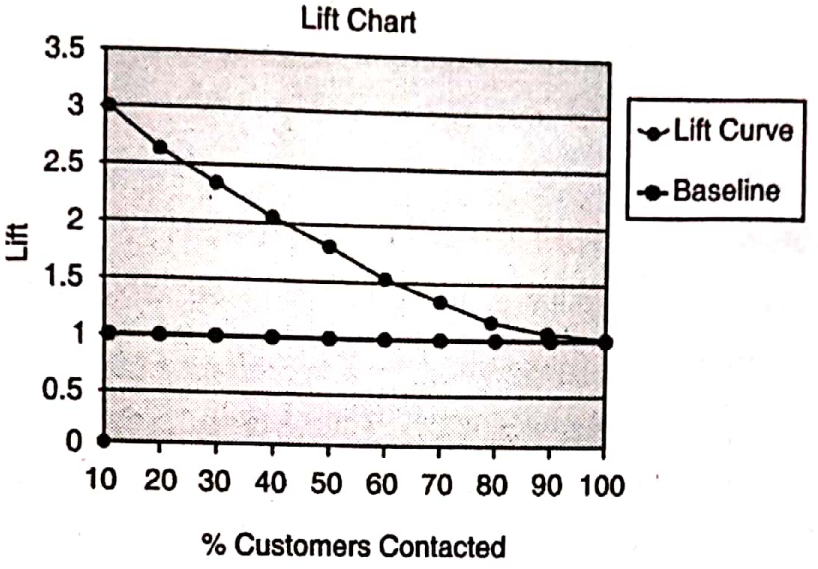


Fig. 3.2.5

Syllabus Topic : Bayesian Methods

3.3 Bayesian Methods

Q. 3.3.1 Write short note on Bayesian methods. (Ref. Sec. 3.3) (4 Marks)

- Bayes' theorem is one of the earliest probabilistic inference algorithms developed by Reverend Bayes'. It is a classification technique based on Bayes' Theorem.
- It assumes that there is independence among predictors. In simple terms, a Naive Bayes' classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.
- For example, a fruit may be considered to be an apple if it is red, round, and about 3 inches in diameter. Even if these features depend on each other or upon the existence of the other features, all of these properties independently contribute to the probability that this fruit is an apple and that is why it is known as **'Naive'**.

$p(class/data) = p(data/class) \cdot p(class)\ p(data)$

3.3.1 Bayes' Theorem Implementation

- Let us implement the Bayes' Theorem using a simple example. Suppose we want to find the odds of an individual having high blood pressure, given that he or she was tested for it and got a positive result.
- In the medical field, such probabilities play a very important role as it usually deals with life and death situations.

We assume the following :

- P(Bp) is the probability of a person having Blood pressure.
- Assume 1% of the general population has Blood pressure: So p(Bp)= 0.01
- P(Pos) is the probability of getting a positive test result.
- P(Neg) is the probability of getting a negative test result.
- P(Pos|Bp) is the probability of getting a positive result on a test done for detecting Blood pressure, given that you have Blood pressure. This has a value 0.9. In other words the test is correct 90% of the time. This is also called the Sensitivity or True Positive Rate.
- P(Negl ~ Bp) is the probability of getting a negative result on a test done for detecting diabetes, given that you do not have diabetes. This also has a value of 0.9 and is therefore correct, 90% of the time. This is also called the Specificity or True Negative Rate.
- The Bayes formula is as follows :

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

- P(A) is the prior probability of A occurring independently. In our example this is P(Bp). This value is given to us.
- P(B) is the prior probability of B occurring independently. In our example this is P(Pos).
- P(A|B) is the posterior probability that A occurs given B. In our example this is P(Bp|Pos).
- That is, the probability of an individual having Blood pressure, given that, that individual got a positive test result. This is the value that we are looking to calculate.
- P(B|A) is the likelihood probability of B occurring, given A. In our example this is P(Pos|Bp). This value is given to us.
- Putting our values into the formula for Bayes theorem we get:

$$P(Bp|Pos) = (P(Bp) * P(Pos|Bp) / P(Pos)$$

- The probability of getting a positive test result P (Pos) can be calculated using the Sensitivity and Specificity.

Using specificity and sensitivity are as follows :

P(Pos) = [P(Bp) * Sensitivity] + [P(~Bp) * (1– Specificity))]

P(Bp) = Probability having blood pressure = 0.01

P(~Bp) = Probability of not having blood pressure = 0.99

Sensitivity = P(Pos/Bp) = getting positive result = 0.9

P(Negl~Bp) = 0.9 = getting negative result

P(Pos) = Probability of getting positive test result = [P(Bp) * Sensitivity] + [P(~Bp) * (1– Specificity))]

3.3.2 Naive Bayes Classifier (Simplification)

Q. 3.3.2 Explain naive Bayes classifier with example. (Ref. Sec. 3.3.2) (5 Marks)

- **The naive Bayes algorithm reduces the complexity of Bayes' theorem by assuming conditional independence over the training dataset.**
- This assumption makes the Bayes algorithm, naive.
- Given, n different attribute values, the likelihood now can be written as,

$$P(X_1 \ldots X_n|Y) = \prod_{i=1}^{n} P(X_i|Y)$$

- In Naive Bayes algorithm considers the features that particular feature in a class is independent or not related to the presence of any other feature.
- For example, a fruit may be considered to be an apple if it is red, round, and about 3 inches in diameter. In this case all properties or features are independently contribute to the probability that this fruit is an apple and that is why it is known as 'Naive'.
- So in the above example, we are considering only one feature, that is the test result. If we add another feature, 'exercise'.
- Let's say this feature has a binary value of 0 and 1, where the former signifies that the individual exercises less than or equal to 2 days a week and the latter signifies that the individual exercises greater than or equal to 3 days a week.
- If we had to use both of these features, namely the test result and the value of the 'exercise' feature, to compute our final probabilities, Bayes' theorem would fail. Naive Bayes' is an extension of Bayes' theorem that assumes that all the features are independent of each other.

☞ **Advantages**

- It is easy and fast to predict class of test data set. It performs well in multi class prediction.
- When assumption of independence holds, a Naive Bayes classifier performs better compare to other models like logistic regression and you need less training data.
- It perform well in case of categorical input variables compared to numerical variable(s). For numerical variable, normal distribution is assumed (bell curve, which is a strong assumption).

☞ **Disadvantages**

- If categorical variable in test data set has a category ,which was not observed in training data set, then model will assign a 0 (zero) probability. It will be unable to make a prediction. This is often known as "Zero Frequency". To solve this, one of the simplest techniques is called Laplace estimation.
- The limitation of Naive Bayes is the assumption of independent predictors. In real life situation, it is not possible to get a set of predictors which are completely independent.

☞ **Applications of Naive Bayes Algorithms**

- Naive Bayes is used for making predictions in real time. It is very fast.
- It is used for multi class prediction feature. It predict the probability of multiple classes of target variable.
- Naive Bayes classifiers mostly used in text classification (due to better result in multi class problems and independence rule) have higher success rate as compared to other algorithms. As a result, it is widely used in Spam filtering (identify spam e-mail) and Sentiment Analysis (in social media analysis, to identify positive and negative customer sentiments).
- Naive Bayes Classifier and Collaborative Filtering together builds a Recommendation System. It uses machine learning and data mining techniques to to predict whether a user would like a given resource or not.

☞ **Example of Naive Bayes Classifier**

Sr. No	Age	Income	Student	Credit card performance	Class- Buys computer
1	< 30	High	No	Fair	no
2	< 30	High	No	Excellent	No

Sr. No	Age	Income	Student	Credit card performance	Class- Buys computer
3	30 To 59	High	No	Fair	Yes
4	> 60	Medium	No	Fair	Yes
5	> 60	Low	Yes	Fair	Yes
6	> 60	Low	Yes	Excellent	No
7	30 To 59	Low	Yes	Excellent	Yes
8	< 30	Medium	No	Fair	No
9	< 30	Low	Yes	Fair	Yes
10.	> 60	Medium	Yes	Fair	Yes
11	< 30	Medium	Yes	Excellent	Yes
12	30 To 59	Medium	No	Excellent	Yes
13	30 To 59	High	Yes	Fair	Yes
14	>60	Medium	No	excellent	N0

X = (Age = '< = 30', Income = medium, student = yes, credit_rating = fair)

P(c1) = p(Buys_computer = yes)= 9/14 = 0.643

P(c2) = p(buys_computer = no) = 5/14 = 0.357

P(age < = 30/buys_computer = yes)

$$= \frac{\text{(number of rows where age < = 30 buys computer = yes)}}{\text{(number of rows which buys computer = yes)}}$$

P(age < = 30/buys_computer = yes) = 2/9 = 0.222

P(age < = 30/buys_computer = no) = 3/5 = 0.6000

P(Income = medium/buys_computer = yes) = 4/9 = 0.444

P(income = medium/buys_computer = no) = 2/5 = 0.400

P(student = yes/buys_computer = yes)6/9 = 0.667

P(student = ues/buys_computer = no) = 1/5 =.2000

P(credit = fair/buys_computer = yes) = 6/9 = 0.667

P(credit = fair/buys_computer = no) = 2/5 = 400

X = (Age = ' < = 30',Income= medium,student=yes,credit_rating= fair)

To find p(X/buys computer = yes) = p(age<30/buys computer yes)

*p(income = medium/buys computer yes)*p(student = yes/buys computer = yes)

*p(credit ration = fair/buys computer = yes)

= 0.222*0.444*0.667*0.667 = 0.044

3.3.3 Bayesian Networks

Q. 3.3.3 What is Bayesian networks ? (Ref. Sec. 3.3.3) **(4 Marks)**

- Bayesian networks are a type of **Probabilistic Graphical Model**. It is used to build models from data and/or expert opinion.
- It can be used for a wide range of tasks including time series prediction, decision under uncertainty, diagnostics, automated insight anomaly detection and reasoning.
- A Bayesian network consist of two main components. The first is an acyclic oriented graph where the nodes correspond to the predictive variables and the arcs indicate relationships of stochastic dependence.
- The variable X_j associated with node $\mathbf{a_j}$ in the network which is dependent on predecessor nodes of a_j.
- The second component consists of the table associated with the variable Xj indicates the conditional distribution of $P(X_j | C_j)$, where C_j represents the set of explanatory variables associated with the predecessor nodes of node $\mathbf{a_j}$ in the network and is estimated based on the relative frequencies in the dataset.

Syllabus Topic : Logistic Regression

3.4 Logistic Regression

Q. 3.4.1 Write short note on logistic regression. (Ref. Sec. 3.4) **(5 Marks)**

- Logistic regression is used to :
  - Estimate the probability of an event occurs for a randomly selected observation verses the probability that the event does not occur.
  - Predict the effect of variables on binary response variable.
  - Classify observation by estimating the probability that an observation is in particular category.
  - Model the probability of an event occurring depending on the values of the independent variable, which can be numerical.

- Logistic regression is generally used where the dependent variable is Binary. That means the dependent variable can take only two possible values such as "Yes or No", "Default or No Default", "Living or Dead", "Responder or Non Responder", "Yes or No" etc.
- Independent factors or variables can be categorical or numerical variables.

Example of logistic regression

Example 1

- If a credit card company is going to build a model to decide whether to issue a credit card to a customer or not, it will model for whether the customer is going to "Default" or "Not Default" on this credit card. This is called "Default Propensity Modeling" .
- The probability of any event lies between 0 and 1 (or 0% to 100%). when we plot the probability of dependent variable by independent factors, it will demonstrate an 'S' shape curve.

Example 2

- Suppose we have to predict the probability of a given candidate to get admission in a college of his or her choice by the score candidates receives in the admission test. The dependent variable is binary- "Admission "or "No Admission".
- Since the relationship between the Score and Probability of Selection is not linear it shows an 'S' shape, we can't use a linear model to predict probability of selection by a score. We need to do Logit transformation of the dependent variable to make the correlation between the predictor and dependent variable linear. Use a logistic regression model to predict the probability of getting the "Admission.

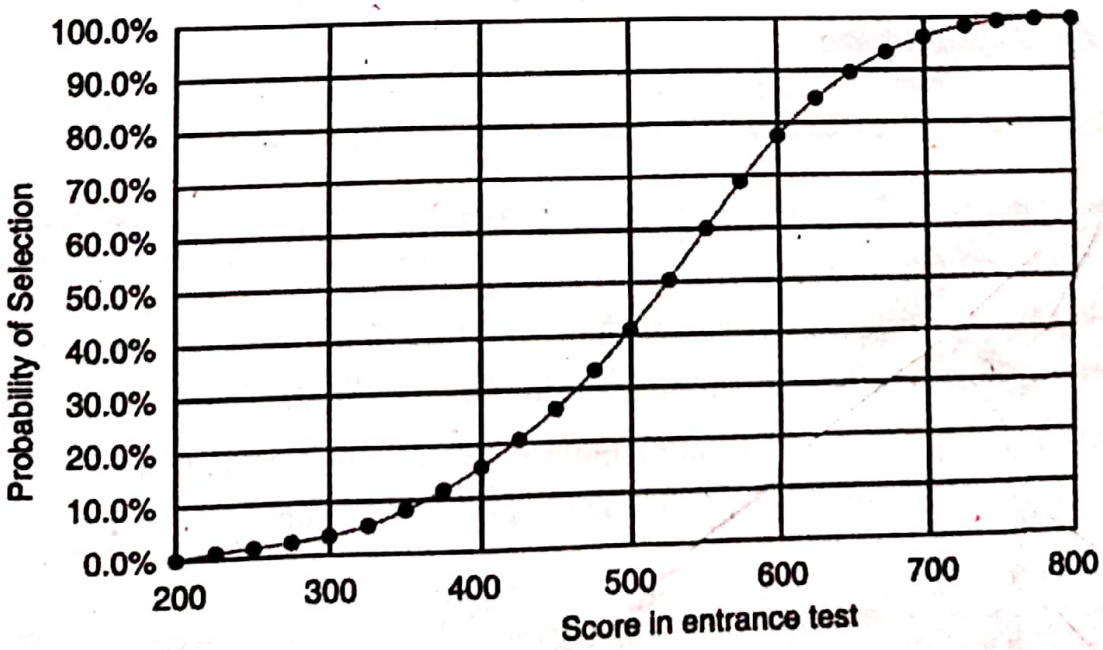


Fig. 3.4.1 : Graph for selection of college

- The above graph is called as Sigmoid function and it gives S-shaped curve. It gives value between $0 < p < 1$.
- The logistic function is defined as :

  Transformed = 1 / (1 + e^ − x)
- Where e is the numerical constant Euler's number and x is a input we plug into the function. Logit expression can be expressed as,

  log(p(x)/(1− p(x))
- where the left hand side is called the logit or log odds function. The odds signifies the ratio of probability of success to probability of failure.

Syllabus Topic : Neural Networks

3.5 Neural Networks

Q. 3.5.1 Write short note on neural network. (Ref. Sec. 3.5) (5 Marks)

- A neural network comprises of units (neurons) which is arranged in layers. It converts an input vector into some output.
- Each unit takes an input, it applies a nonlinear function to it and then passes the output on to the next layer. Generally the networks are defined to be feed-forward: a unit feeds its output to all the units on the next layer, but there is no feedback to the previous layer.
- Weightings are applied to the signals which passes from one unit to another, and in these weightings which are tuned in the training phase to adapt a neural network to the particular problem at hand.

3.5.1 The Rosenblatt Perceptron

- Perceptron were popularised by Frank Rosenblatt in the 1960. They appeared to have very powerful learning algorithm.
- A perceptron is a neural network unit (an artificial neuron) which does certain computations to detect features or business intelligence in the input data.
- It consists of single neuron with adjustable synaptic weights and bias. It can be used to classify linearly separated pattern. A simple perceptron can be used to classify into two classes.
- A Perceptron is a supervised learning algorithm for binary classifiers. This algorithm enables neurons to learn and processes elements in the training set one at a time.

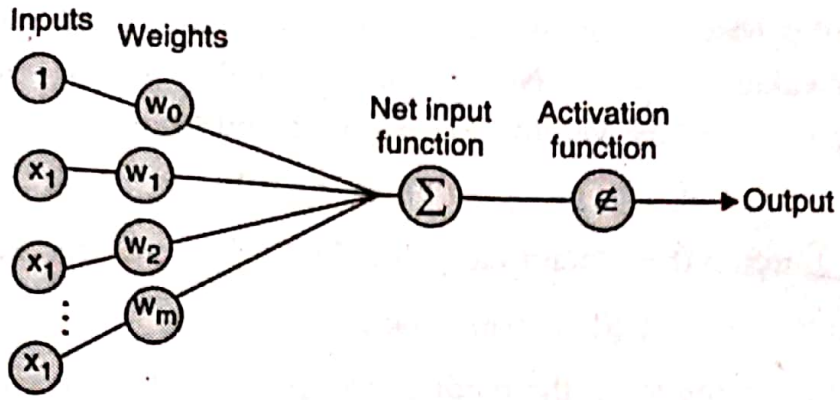


Fig. 3.5.1

- There are two types of Perceptrons: Single layer and Multilayer.
- Single layer Perceptrons can learn only linearly separable patterns.
- Multilayer Perceptrons or feed forward neural networks with two or more layers have the greater processing power.

Perceptron Function

- Perceptron is a function that maps its input "x" which is multiplied with the learned weight coefficient; an output value "f(x)"is generated.

$$f(x) = \begin{cases} 1 & \text{if } \omega \cdot x + b > 0 \\ 0 & \text{otherwise} \end{cases}$$

Where,

"ω" = vector of real-valued weights.

"b" = bias (an element that adjusts the boundary away from origin without any dependence on the input value).

"x" = vector of input x values.

$$\sum_{i=1}^{m} \omega_i x_i$$

Where, "m" = number of inputs to the Perceptron.

- The output can be represented as "1" or "0." It can also be represented as "1" or "–1" depending on which activation function is used.

Inputs of a Perceptron

- A Perceptron accepts inputs, moderates them with certain weight values, then applies the transformation function to output the final result.

- A Boolean output is based on inputs such as salaried, married, age, past credit profile, etc. It has only two values: Yes and No or True and False. The summation function "Σ" multiplies all inputs of "x" by weights "w" and then adds them up as follows :

$$\omega_0 + \omega_1 x_1 + \omega_2 x_2 + \ldots + \omega_n x_n$$

- For example: If $\sum \omega_i x_i > 0$ => then final output "o" = 1 (issue bank loan).

  Else, final output "o" = -1 (deny bank loan).

- In the Perceptron Learning Rule, the predicted output is compared with the known output. If it does not match, the error is propagated backward to allow weight adjustment to happen.
- Perceptron has the following characteristics :
  - Perceptron is an algorithm for Supervised Learning of single layer binary linear classifier.
  - Optimal weight coefficients are automatically learned.
  - Weights are multiplied with the input features and decision is made if the neuron is fired or not.
  - Activation function applies a step rule to check if the output of the weighting function is greater than zero.
  - Linear decision boundary is drawn enabling the distinction between the two linearly separable classes +1 and -1.
- If the sum of the input signals exceeds a certain threshold, it outputs a signal; otherwise, there is no output.

3.5.2 Multi-Level Feed-Forward Networks

- Multilayer Perceptron (MLP) includes at least one hidden layer (except for one input layer and one output layer).
- Multi-level feed-forward neural network, is a more complex structure than the perceptron, since it includes input nodes, hidden nodes and output nodes use a neural network with two input nodes i_1 and i_2, two hidden neurons h_1 and h_2, two output neurons o_1 and o_2.

- Here's the basic structure :

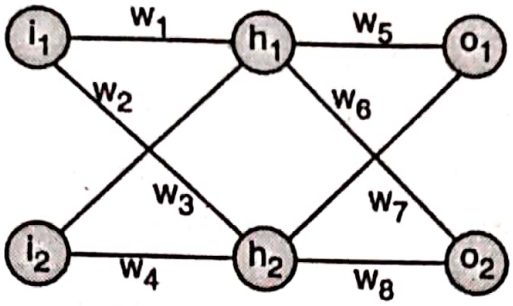


Fig. 3.5.2

- The goal of back propagation is to optimize the weights so that the neural network can learn how to correctly map arbitrary inputs to outputs.
- **Input nodes :** Input nodes receive input the values of the explanatory attributes for each observation. Usually, the number of input nodes equals the number of explanatory variables.
- **Hidden nodes :** Hidden nodes receives the information from input nodes and transforms the input values inside the network. Each node is connected with outgoing arcs to output nodes or to other hidden nodes.
- **Output nodes :** Output nodes receive connections from hidden nodes or from input nodes and return an output value that corresponds to the prediction of the response variable.
- Each node of the network has given weights which are associated with the input arcs. Each node is associated with a distortion or bias coefficient and an activation function.
- Back propagation algorithm is used in multilevel feed forward network.

Syllabus Topic : Support Vector Machines

3.6 Support Vector Machine

Q. 3.6.1 Write short note on support vector machine. (Ref. Sec. 3.6) (5 Marks)

The simply way to describe SVM is a binary classifier. It attempts to find a hyperplane that can separate two class of data by the largest margin. Quazi Marufur Rahman gives a very good example of what is margin, and Janice Gates points kernel trick. I think the kernel trick is most important part of SVM, it distinct SVM with other classifiers.

3.6.1 Structural Risk Minimization

- Structural Risk Minimization (SRM) (Vapnik and Chervonekis, 1974) is an inductive principle for model selection used for learning from finite training data sets.
- It describes a general model of capacity control and provides a trade-off between hypothesis space complexity (the VC dimension of approximating functions) and the quality of fitting the training data
- Suppose we have two dimensional data with different features x_1 so x_2.

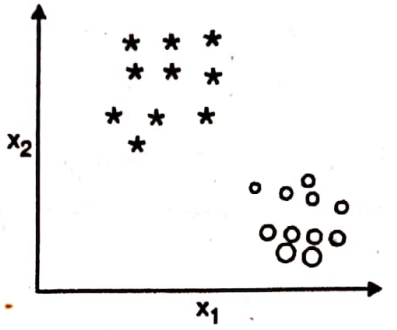


Fig. 3.6.1

- The above data can be divided into two classes class 1 and class 2. The above data is linearly separable.

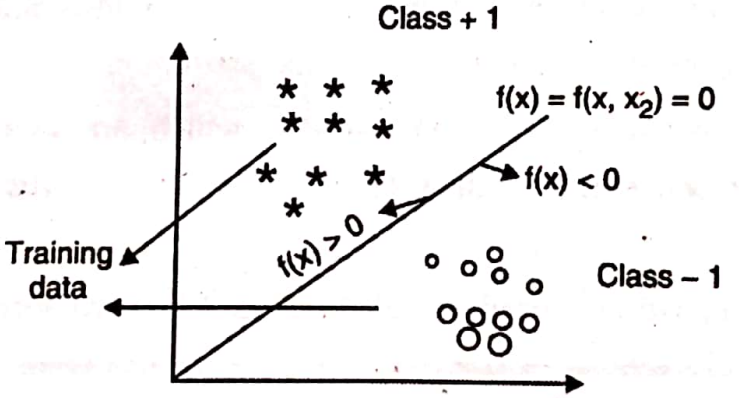


Fig. 3.6.2

- A straight line will classify data into two classes. The equation is $f(x) = f(x_1, x_2) = 0$.
- The classifier is called as linear classifier. Data is called training sets.

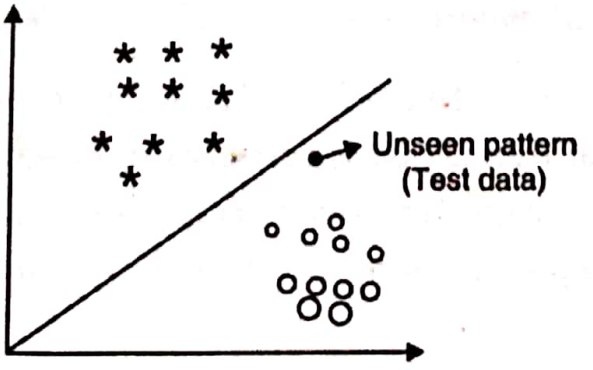


Fig. 3.6.3

- Suppose we have unseen data set. The value of unseen dataset $f(x) < 0$ then it is classified as class – I.
- Now we are in position to define tow quantities training error and test error.
- Since during training phase classifier learns the distribution of data, the low value of training error is required. Test error also should be low. Because it controls the unseen data pattern.
  1. We have to always look for test error along with training error.
  2. Improving on training error not always improves test error.
  3. Increase in machine capacity may result in poor test performance.
  4. It is difficult to estimate true test error of classifier.
- To ensure low test value of classifier.

$$\text{VC dimension is used} = \text{Test error} \leq \text{training error} + \sqrt{\frac{a\left(\log\left(\frac{2m}{a}\right)\right) + 1 - \log\left(\frac{n}{4}\right)}{m}}$$

- It gives upper bound of test error with probability 1 – n.

Where, M = Number of training samples.

a = related capacity of machine n is called as VC dimension

VC = (Vapnik – Crervonekis ⇒ test error ≤ training error complexity

- The graph of VC dimension with fixed sample size.

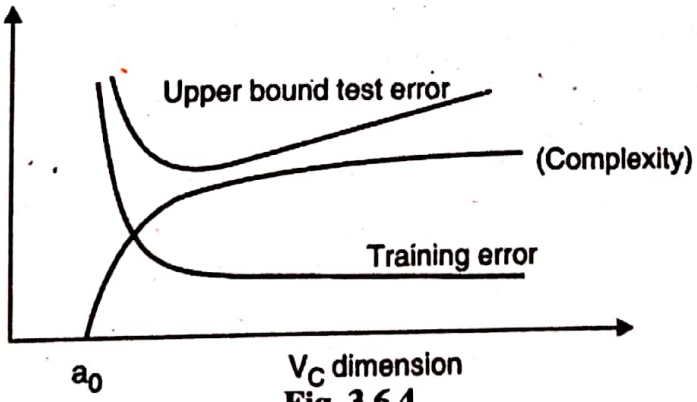


Fig. 3.6.4

- As we increase VC dimension, the training error will be reduced. Complexity increases with VC dimension.
- Upper round (dimension) first decreases later on it increases. For efficient classifier, the value of test error should be minimum. To achieve this sum of penalty error or complexity error and training error should be minimum.

☞ **Points in general position**

– In n dimensional feature space a set of m points (m > n) is in general position iff no subset (n + 1) points lie on (n – 1) dimensional hyperplane.

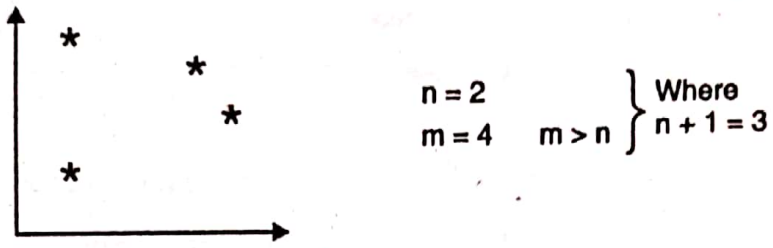


Fig. 3.6.5

– If we add one more points.

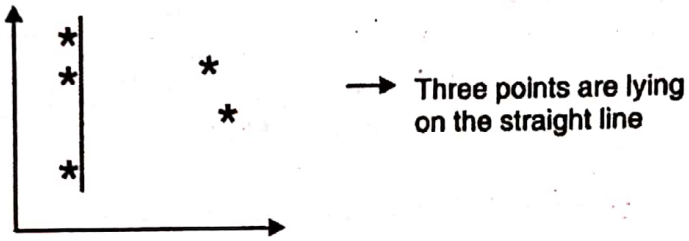


Fig. 3.6.6

– So we can say all above 3 points are not in general position.

☞ **Shattering**

– Hypothesis (H) shatters m points in n-dimensional space if all possible combinations of m points in n-dimensional space are correctly classified by H.

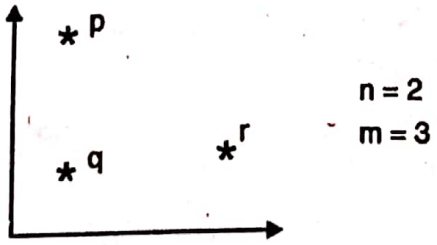


Fig. 3.6.7

$2^3 = 8$ possible arrangements as this points can take two values 0 or1.

– So VC dimension is cardinality of the largest set of points that the hypothesis can shatter.

– VC dimension of linear classifier, (n + 1) {points should be in general position}.

– For non linear classifier VC dimension is difficult to compare. VC dimension is directly related to machine/hypothesis capacity error VC dimension gives probabilistic upper bound test error.

3.6.2 Maximal Margin Hyperplane for Linear Separation

- The following is an example of hyper plane that separates training instances with no errors.

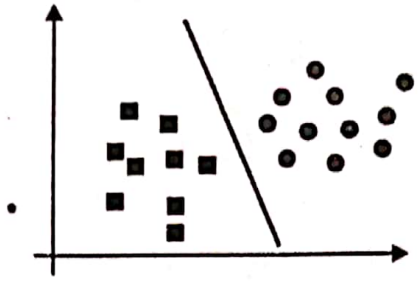


Fig. 3.6.8

- If we think then there are multiple hyper planes which can be choose for separating two data points.

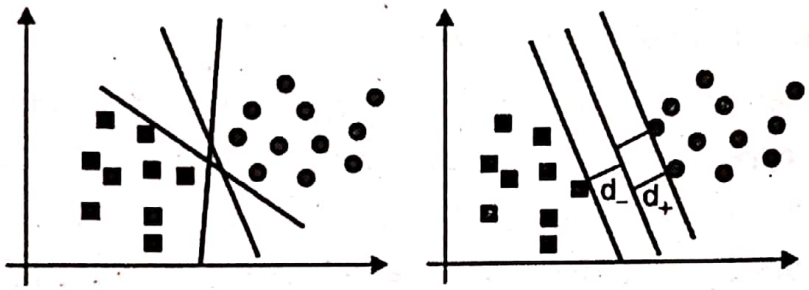


Fig. 3.6.9

- For the maximum margin hyper plane only examples on the margin matter (only these affect the distances). These are called support vectors.

☞ **Definition**

Define the hyper planes H such that :

$w \cdot x_i + b \geq +1$, when $y_i = +1$.

$w \cdot x_i + b \geq -1$, when $y_i = -1$.

H_1 and H_2 are the planes :

$H_1 : w \cdot x_i + b \geq +1$

$H_2 : w \cdot x_i + b \geq -1$

The points on the planes H_1 and H_2 are the tips of the support vectors.

The planes H_0 is the median in between, where $w \cdot x_i + b = 0$.

d^+ = the shortest distance to the closest positive point.

d^- = the shortest distance to the closest negative point.

The margin (gutter) of a separating hyper plane is $d^+ + d^-$.

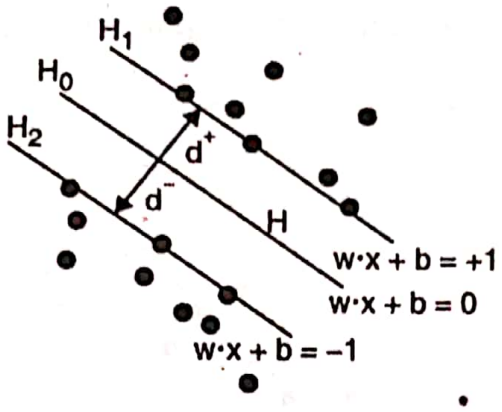


Fig. 3.6.10

3.6.3 Nonlinear Separation

- **Nonlinear Classification :** Classes may not be separable by a linear boundary.
- **Kernels:** Make linear models work in nonlinear settings By mapping data to higher dimensions where it exhibits linear patterns.
- The simplest way to separate two groups of data is with straight line, flat plane an N-dimensional hyper plane. However there are situations where a non linear region can separate the groups more efficiently.
- SVM handles this by using kernel function(non linear) to map the data into different space where a hyper plane (linear) cannot be used to do the separations.
- It means a non linear function is learned by linear learning machine in a high dimensional feature space which the capacity of the system is controlled by a parameter that does not depend on the dimensionality of the space.
- This is called as kernel trick which means kernel function transform the data into higher dimensional feature space to make it possible to perform the linear separation.
- Kernel function map the data into new space. It take the inner product of new vectors. The image of the inner product of the data is the linear product of the images of the data. Two kernel function are shown as below :

☞ **Polynomial kernel**

$$k(x_i, x_h) = \left(x_i' x_h + 1\right)^d$$

☞ **Gaussian kernels**

$$k(x_i, x_h) = \exp\left(\frac{-\| x_i - x_h \|^2}{2\sigma^2}\right)$$

3.7 Clustering

Cluster analysis or **clustering** is the task of grouping a set of objects in such a way that objects in the same group (called a **cluster**) are more similar (in some sense) to each other than to those in other groups (clusters).

Syllabus Topic : Clustering Methods

3.7.1 Clustering Methods

Q. 3.7.1 What are the characteristics of clustering method? (Ref. Sec. 3.7.1) (4 Marks)

Clustering methods must satisfy a few general necessities, as indicated below.

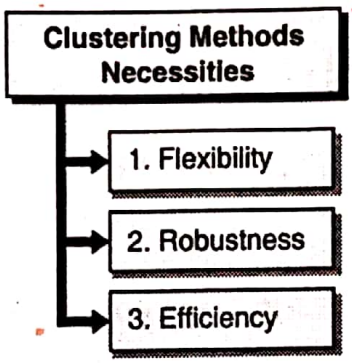


Fig. 3.7.1 : Clustering Methods Necessities

➜ **1. Flexibility**

- There are clustering methods which can be used on numerical characteristics only. In such cases most of the time Euclidean metrics is used to determine the distances between observations.
- A flexible clustering algorithm is used to analyse datasets containing categorical attributes.

➜ **2. Robustness**

The robustness of an algorithm is the stability of the clusters generated with respect to small changes in the values of the attributes of each observation.

➜ **3. Efficiency**

In some applications there are large number of observations In such case clustering algorithms must generate clusters efficiently in order to guarantee reasonable computing times for large problems.

3.7.2 Taxonomy of Clustering Methods

Q. 3.7.2 What is taxonomy of clustering method? (Ref. Sec. 3.7.3) (4 Marks)

The different types of Clustering based on the logic are partition methods, hierarchical methods, density based methods and grid methods.

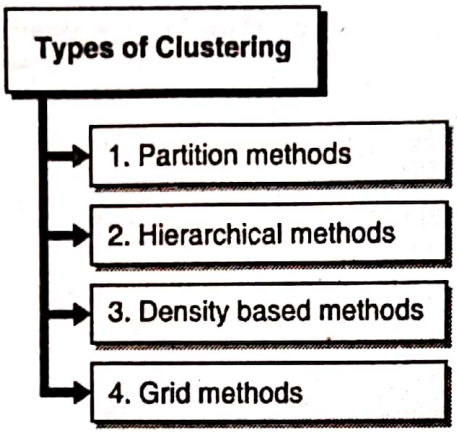


Fig. 3.7.2 : Types of Clustering

➜ **1. Partition methods**

Partition methods, is a division of the given dataset into a predetermined number K of non-empty subsets. They generate a spherical or at most convex shape after grouping.

➜ **2. Hierarchical methods**

In Hierarchical methods, subset is divided into tree structure. It categorized clusters by different homogeneity thresholds. Predetermined clusters are not required.

➜ **3. Density-based methods**

- Hierarchal and partition methods are founded on the distance between observations. Density-based methods determine clusters from the number of observations locally falling in a neighbourhood of each observation.
- For each member which belongs to a specific cluster, a neighbourhood with a specified diameter should contain a number of observations which should not be less than a minimum threshold value.
- Density-based methods identify clusters of non-convex shape which helps them to isolate any possible outliers.

➜ **4. Grid methods**

Grid methods obtain a grid structure consisting of cells. The grid structure is achieved to reduce computing times, despite a lower accuracy in the clusters generated.

3.7.3 Affinity Measures

In Hierarchical clustering clusters are repeatedly links to pairs of clusters so that every data object is included in the hierarchy. To determine the similarity between the clusters the ***distance functions***, such as the ***Manhattan*** and ***Euclidian*** distance functions, are used

3.7.3.1 Distance Functions

- Given two *p*-dimensional data objects $i = (x_{i1}, x_{i2}, \ldots x_{ip})$ and $j = (x_{j1}, x_{j2}, \ldots, x_{jp})$, the following common distance functions can be defined :

1. Euclidian Distance Function
2. Manhattan Distance Function

➜ **1. Euclidian Distance Function**

$$d(i, j) = \sqrt{|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \ldots + |x_{ip} - x_{jp}|^2}$$

➜ **2. Manhattan Distance Function**

$$d(i, j) = \sqrt{|x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \ldots + |x_{ip} - x_{jp}|}$$

- Distances are always positive numbers. In the Euclidian distance function, attributes with larger scales of measurement may overcome attributes measured on a smaller scale. To prevent this problem, the attribute values are often normalized to lie between 0 and 1.
- A third option which generalizes both the Euclidean and Manhattan metrics. The Minkowski distance defined as,

$$\text{dist}(i, j) = q\sqrt{\sum_{j=1}^{m} |x_{ij} - x_{kj}|^q}$$

$$= q\sqrt{|x_{i1} - x_{k1}|^q + |x_{i2} - x_{k2}|^q + \ldots + |x_{ip} - x_{kn}|^q}$$

- **Example :**
  To calculate a distance between two points **p** (x_1, y_1) and **q** (x_2, y_2) in xy-plane.

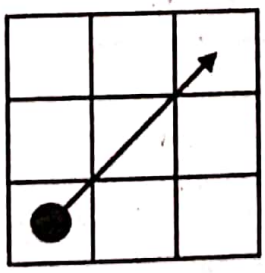
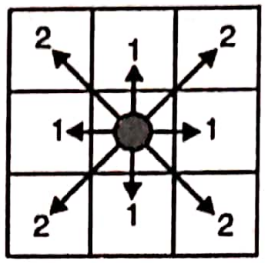


Fig. 3.7.3

- The distance between two points is the sum of the (absolute) differences of their coordinates. E.g. it counts 1 unit for a straight move, and it counts cost as 2 if one takes crossed move.

Manhattan Distance



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

Fig. 3.7.4

- In chess, the distance between squares on the chessboard for rooks is measured in Manhattan distance

3.7.4 Attribute

- An attribute is a data field, which represents a characteristic or feature of a data object. The nouns attribute, dimension, feature, and variable are commonly recognized as attribute in literature.
- In data warehousing attributes are referred as dimension. In Machine learning literature it is referred as feature, while statisticians call this term as variable.
- Data mining and database professionals commonly use the term attribute. Attributes describing a customer object can include, for example, customer ID, name, and address.
- Univatiate distribution involves only one attribute. The distribution of data having two attributes is known as bivariate.

The type of an attribute is determined by the set of possible values the attribute can have. Attributes can be nominal, binary, ordinal, or numeric. In the following subsections, we introduce each type.

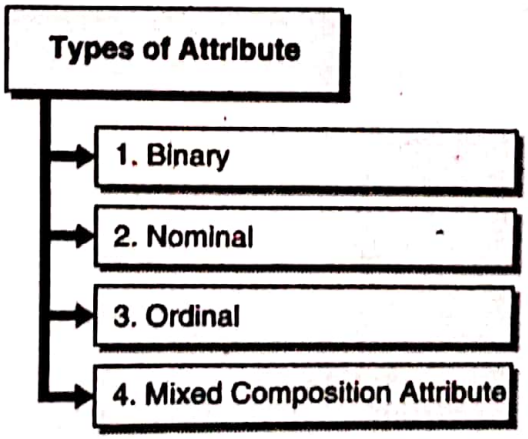


Fig. 3.7.5

1. Binary Attributes

Q. 3.7.3 Write short note on Binary attribute. (Ref. Sec. 3.7.5(1)) **(5 Marks)**

- Nominal attribute is treated as binary attribute. It has two categories or states 0 or 1.
- 0 means attribute is absent and 1 means it is present. Binary attributes are referred to as Boolean as two states correspond to true and false. 1 means that it is present.
- E.g. Smoker describing a patient object, 1 indicates that the patient smokes, while 0 indicates that the patient does not.
- A similarity measure for two objects, i and j, will typically return the value 0 if the objects are unalike. The higher the similarity value, the greater the similarity between objects. (Typically, a value of 1 indicates complete similarity, that is, that the objects are identical.)
- A dissimilarity measure works the opposite way. It returns a value of 0 if the objects are the same (and therefore, far from being dissimilar. The higher the dissimilarity value, the more dissimilar the two objects are.
- A nominal attribute can take on two or more states. For example, flower color is a nominal attribute that may have, say, five states: red, yellow, green, pink, and blue
- Let the number of states of a nominal attribute be M. The states can be denoted by letters, symbols, or a set of integers, such as 1, 2, . . . , M. The dissimilarity between two objects i and j can be computed based on the ratio of mismatches :

$$d(i, j) = \frac{p - m}{p}$$

- Where m is the number of matches (i.e., the number of attributes for which i and j are in the same state), and p is the total number of attributes describing the objects. Weights can be assigned to increase the effect of m or to assign greater weight to the matches in attributes having a larger number of states
- There is another approach which involves computing a dissimilarity matrix from the given binary data.

Table 3.7.1 : A contingency table for binary attributes

	Object j			
Object i		1	0	sum
	1	q	R	q + r
	0	s	t	s + t
	sum	q + s	r + t	P

- Where q is the number of attributes that equal 1 for both objects i and j. r is the number of attributes that equal 1 for object i but that are 0 for object j. s is the number of attributes that equal 0 for object i but equal 1 for object j. And t is the number of attributes that equal 0 for both objects i and j.
- The total number of attributes is p. Where $p = q + r + s + t$. Recall that for symmetric binary attributes, each state is equally valuable.
- If objects i and j are described by symmetric binary attributes, then the dissimilarity between i and j is,

$$d(i, j) = \frac{r + s}{q + r + s + t}$$

- The above equation states a degree of similarity between pairs(i,j) of observations through the **coefficient of similarity.**
- Assume that all *n* attributes are binary and asymmetric. In such case, for a pair of asymmetric attributes it is interesting to match **positives,** records possessing the property relative to each attribute.
- For binary variables, the Jaccard coefficient is therefore used

$$d(i, j) = r + s/q + r + s$$

➜ 2. Nominal Attribute

Q. 3.7.4 Write short note on Nominal attribute. (Ref. Sec. 3.7.5(2)) (4 Marks)

- Nominal attributes means "relating to names." Nominal attribute are symbols or names of things. Each value denotes some kind of category, code, or state. Nominal attributes are also referred as categorical. In computer science, the values are also known as enumerations.
- Nominal attributes. Suppose that Hair color and Marital status are two attributes describing person objects. In our application, possible values for Hair color are black, brown blond, red, auburn, grey, and white.
- It is symmetric attribute where the value is greater than 2.We use similarity coefficient in extended form, dist (i ,j) = (n – f)/n

  Where, f is the number of attributes in which observations i and j take the same value.

➜ 3. Ordinal Attribute

Q. 3.7.5 Write short note on Ordinal attribute. (Ref. Sec. 3.7.5(3)) (4 Marks)

- Values of ordinal attribute has possible values and have a meaningful order or ranking among them. The magnitude between consecutive values is not known.

- Suppose that Drink size corresponds to the size of drinks available at a restaurant. This ordinal attribute has three possible values – small, medium, and large. However, we cannot tell from the values how much bigger, say, a medium is from a large.
- Ordinal variable can be discrete or continuous.
- Order is important and can be treated like interval scaled.
- Replace ordinal variables value by its rank $^{r}r \in \{1....M_f\}$
- Map the range of variable[0,1].

$$Z_d = \frac{r_r - 1}{M_f - 1}$$

➜ 4. Mixed Composition attribute

- A dataset contain all attribute types nominal, ordinal, symmetric binary, asymmetric binary etc. To define an overall affinity measure which defines similarity between observations d_i and d_j. One can use weighted formula as follows,

$$d(I, j) = \frac{\sum_{f=1}^{P} w_{ij}^{(f)} d_{ij}^{(f)}}{\sum_{f=1}^{P} w_{ij}^{(f)}}$$

- If f is numeric it uses the normalized distance.
- If f is binary or nominal $d_{ij} = 0$ if $x_{if} = x_{jf}$.
- If f is ordinal then it computes rank z_{if} .

$$Z_d = \frac{r_d - 1}{M_f - 1}$$

Syllabus Topic : Partition Methods

3.8 Partition Methods

- Partition methods are heuristic nature. They are based on greedy methods where at each step they make the choice that locally appears the most advantageous.
- There is guarantee that a good subdivision will be obtained for the majority of the datasets. The **K-means** method and the **K-medoids** method, , are two of the best-known partition algorithms

3.8.1 *K*-means algorithm

Q. 3.8.1 Explain K-means method. (Ref. Sec. 3.8.1) **(4 Marks)**

- K means clustering is an algorithm is used to classify or group the objects based on features or attributes. Algorithm is used to classify into k number of groups.
- K is positive integer. The grouping is done by minimizing the sum of squares of distances between data and the corresponding cluster centroid.
- The algorithm assumes two clusters, and each individual's scores include two variables (as in the example above
- In non-hierarchical clustering such as the k-means algorithm. The relationship between clusters is undetermined. Distance functions such as Manhattan and Euclidian distance functions, are used to determine similarity.

Distance Functions :

- Given two *p*-dimensional data objects $i = (x_{i1}, x_{i2}, ..., x_{ip})$ and $j = (x_{j1}, x_{j2}, ..., x_{jp})$, the following common distance functions can be defined:

1. Euclidian Distance Function :

$$d(i, j) = \sqrt{|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \ldots + |x_{ip} - x_{jp}|^2}$$

2. Manhattan Distance Function :

$$d(i, j) = \sqrt{|x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \ldots + |x_{ip} - x_{jp}|}$$

3. Steps of k-means Algorithm :

1. Choose *k* clusters arbitrarily.
2. Initialize cluster centres with those *k* clusters.
3. loop
   a) Partition by assigning or reassigning all data objects to their closest cluster center.
   b) Compute new cluster centers as mean value of the objects in each cluster.
   c) Until no change in cluster center calculation.

☞ **Example of implementation of k means algorithm using k=2(partitions)**

	Variable 1	Variable 2
1	1	1.0
2	1.5	2.0
3	3	4.0

	Variable 1	Variable 2
4	5	7.0
5	3.5	5.0
6	4.5	5.0
7	3.5	4.5

Step 1 :

Randomly we choose two centroids for k = 2.

In this case two centroids are c_1 and c_2 where c_1 = (1.0,1.0) and c_2 = (5.07.0).

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

$$d(m_1, 2) = \sqrt{|1.0 - 1.5|^2 + |1.0 - 2.0|^2} = 1.12$$

$$d(m_2, 2) = \sqrt{|5.0 - 1.5|^2 + |7.0 - 2.0|^2} = 6.10$$

Step 2 :

We obtain clusters containg {1,2,3} and centroids{4,5,6,7}.

	centroid 1	centroid 2
1	0	7.21
2(1.5,2.0)	1.12	6.10
3	3.61	3.61
4	7.21	0
5	4.72	2.06
6	5.31	2.06
7	4.30	3

L_1 = (1/3(1.0+1.5+3.0), 1/3(1.0+2.0+4.0) = (1.83,2.33) = cluster 1

L_2 = 1/4(5.0+3.5+3.5), 1/3(7.0+5.0+4.5) = (4.12,5.38)) = cluster 2

$$\sqrt{(m - x)^2 + (m - y)^2}$$

$$d(m_1, 2) = \sqrt{|1.0 - 1.5|^2 + |1.0 - 2.0|^2} = 1.12$$

$$d(m_2, 2) = \sqrt{|5.0 - 1.5|^2 + |7.0 - 2.0|^2} = 6.10$$

We are still not sure that each individual has been assigned to the right cluster. So, we compare each individual's distance to its own cluster mean and to that of the opposite cluster.

And we find :

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

Individual 3 is closer to the mean of the opposite cluster (Cluster 2) than its own (Cluster 1). In other words, each individual's distance to its own cluster mean should be smaller that 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)

3.8.2 K-medoids Algorithm

Q. 3.8.2 Explain *K*-medoids algorithm. (Ref. Sec. 3.8.2) **(5 Marks)**

- K-means tries to minimize the total squared error. While *k*-medoids minimizes the sum of dissimilarities between points labelled to be in a cluster and a point designated as the center of that cluster. In contrast to the *k*-means algorithm, *k*-medoids chooses datapoints as centers
- Instead of taking mean value of the object in a cluster as reference point , mediods can be used, which is the most centrally located object in cluster.
- K medoids is called as **Partitioning Around Medoids (PAM)** algorithm. :
- All the items from the input data set are examined by one to see that they are medoids are not.

1. Initialize : arbitrarily select *k* out of the *n* data points as the medoids.

2. Associate each data point to the nearest medoid.
3. For each medoid *m* and each data point *h* associated to *m*, swap *m* and h and compute the total cost (that is, the average dissimilarity of *h* to all the data points associated to *m*). Select the medoid *h* with the lowest cost of the configuration.

- Repeat alternating steps 2 and 3 until there is no change in the assignments.
- In more simpler terms for each pair of a medoid *m* and a non-medoid object *h*, measure whether *h* is better than *m* as a medoid.
- Use the squared-error criterion.

$$E = \sum_{i=1}^{k} \sum_{P \varepsilon C_j} d(p, m_1)^2$$

Compute E_h-E_m.

Choose the minimum swapping cost.

☛ **Four Swapping Cases**

- When a medoid *m* is to be swapped with a non-medoid object *h*, check each of other non-medoid objects *j*.

f is in cluster of *m* ⇒ reassign *j*.

Case 1 : *j* is closer to some *k* than to *h*; after swapping *m* and *h*, *j* relocates to cluster represented by *k*.

$$C_{jmh} = d(j, k) - d(j, m) \geq 0$$

Case 2 : *j* is closer to *h* than to *k*; after swapping *m* and *h*, *j* is in cluster represented by *h*.

$$C_{jmh} = d(j, h) - d(j, m)$$

j is in cluster of some *k*, not m⇒compare *k* with *h*.

Case 3 : *j* is closer to some *k* than to *h*; after swapping *m* and *h*, *j* remains in cluster represented by *k*.

$$C_{jmh} = d(j, k) - d(j, k) = 0$$

Case 4 : *j* is closer to *h* than to *k*; after swapping *m* and *h*, *j* is in cluster represented by *h*.

$$C_{jmh} = d(j, h) - d(j, k) < 0$$

The K-medoids algorithm requires a large number of iterations and is not suited to deriving clusters for large datasets.

Syllabus Topic : Hierarchical Methods

3.9 Hierarchical Methods

Q. 3.9.1 Explain single linkage, complete linkage, average linkage and ward distance. (Ref. Sec. 3.9) **(5 Marks)**

- Hierarchical clustering generates hierarchy in clusters. No need to specify k. It is more deterministic.
- The graphical representation of the resulting hierarchy is a tree-structured graph called a dendrogram.
- In order to calculate the distance between two clusters, the hierarchical algorithms resort to one of five alternative measures: minimum distance, maximum distance, mean distance, distance between centroids, and ward distance.

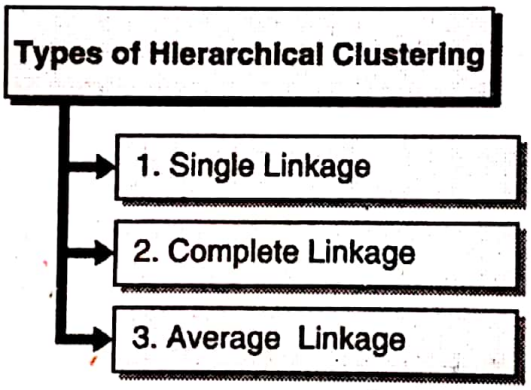
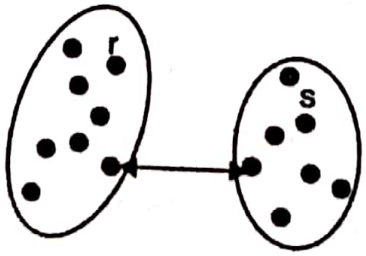


Fig. 3.9.1 : Types of hierarchical clustering

➜ 1. Single linkage

- In single linkage hierarchical clustering, the shortest distance between two points in each cluster is defined.
- For example, the distance between clusters "r" and "s" is equal to the length of the arrow between their two closest points.

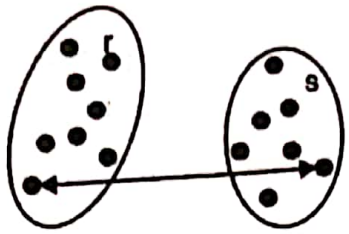


$L(r,s) = \min(D(x_{ri}, y_{sj}))$

Fig. 3.9.2

2. Complete linkage

- In complete linkage hierarchical clustering, longest distance between two points in each cluster is defined.
- For example, the distance between clusters "r" and "s" is equal to the length of the between their two furthest points.

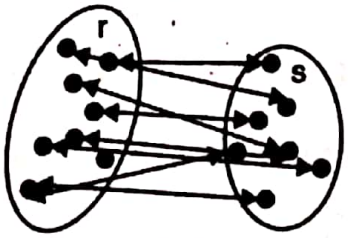


$$L(r,s) = \max(D(x_{ri}, x_{sj}))$$

Fig. 3.9.3

3. Average Linkage

- In average linkage hierarchical clustering, the average distance between each point in one cluster to every point in the other cluster is defined.
- For example, the distance between clusters "r" and "s" to the left is equal to the average length each arrow between connecting the points of one cluster to the other.



$$L(r,s) = \frac{1}{n_r n_s} \sum_{j=1}^{n_r} \sum_{i=1}^{n_s} (D(x_{ri}, x_{sj}))$$

Fig. 3.9.4

Ward distance

- The Ward distance, based on the analysis of the variance of the Euclidean distances between the observations.
- Methods based on the Ward distance tend to generate a large number of clusters, each containing a few observations.

Centroid Method

- In centroid method, distance between the two mean_vectors of the clusters is consider as the distance between two clusters. At each stage of the process we combine the two clusters that have the smallest centroid distance.

- Hierarchical methods can be subdivided into two main groups: agglomerative and divisive methods.

3.9.1 Agglomerative and Divisive Hierarchical Methods

3.9.1.1 Agglomerative Method

- Agglomerative method is bottom up clustering. Suppose there is set of N observations.
- Calculate the distances (similarities) between the clusters equal the distances (similarities) between the items they contain. Join the two most similar clusters.
- In agglomerative or bottom-up clustering method we assign each observation to its own cluster. Then,

Step 1 : Calculate the similarity (e.g., distance) between each of the clusters and join the two most similar clusters.

Step 2 : Find the nearest (most similar) pair of clusters and merge them into a single cluster, so that now you have one less cluster.

Step 3 : Compute distances (similarities) between the new cluster and eachof the old clusters.

Step 4 : Repeat steps 2 and 3 until all items are clustered into a single cluster of size N.

3.9.1.2 Divisive Hierarchical Methods

- In divisive or top-down clustering method we allocate all of the observations to a single cluster. We partition the cluster to two least similar clusters.
- Finally, we proceed repetitively on each cluster until there is one cluster for each observation. There is evidence that divisive algorithms produce more accurate hierarchies than agglomerative algorithms in some circumstances but is conceptually more complex.
- In Divisible hierarchical clustering, top down approach is used. It starts with all objects in one cluster. Clusters are subdivided into smaller and smaller clusters until each object forms a cluster on its own. Certain termination condition is satisfied.
- A cluster is split according to some principle, e.g., the maximum Euclidian distance between the closest neighbouring objects in the cluster. Start with single cluster at the top of the tree and continue splitting it into smaller and smaller
- Clusters till the bottom is reached where there are n clusters with one member each. Dendrogram is a tree data structure which illustrates hierarchical clustering techniques.

- Each level shows clusters for that level. Leaf- individual cluster, Root- one cluster. A cluster at level i is the Union of its children clusters at level i + 1.

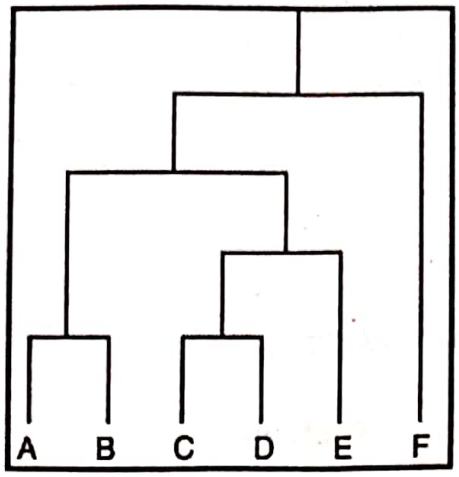


Fig. 3.9.5

Syllabus Topic : Evaluation of Clustering Models

3.10 Evaluation of Clustering Models

Q. 3.10.1 How one evaluates clustering model? (Ref. Sec. 3.10) (5 Marks)

- To measure of performance of a clustering method, one need to verify the clusters generated correspond to an actual regular pattern in the data. It is appropriate to apply other clustering algorithms and to compare the results obtained by different methods.
- In this way it is also possible to evaluate if the number of identified clusters is robust with respect to the different techniques applied.

Cluster cohesion : Measures how closely related are objects in cluster.

Cluster separation : Measures how distinct or well separated cluster is from other cluster.

Let $X = \{x_1, x_2, \ldots, x_k\}$ be the set of K clusters generated.

$$\text{Cohesion is defined as } (X_h)\ \text{coh} = \sum_{\substack{C_i \in X_h \\ C_k \in X_h}} \text{dist}(C_i, C_j)$$

Separation between a pair of clusters is defined as,

$$\text{Sep}(X_p, X_q) = \sum_{\substack{C_i \in X_p \\ C_k \in X_q}} \text{dist}(C_i, C_k)$$

- Silhouette refers a method of interpretation and validation of consistency of clusters of data. The silhouette value is a measure of how similar an object is to its own cluster (cohesion) compared to other cluster (separation).
- The coefficient value ranges from – 1 to + 1. The high value indicate that the object is well matched with its own cluster and poorly matched with neighbouring cluster. Silhouette can be calculated with distance metric such as eclulidean or Manhattan distance.

3.11 Exam Pack (Review Questions)

☞ **Syllabus Topic : Classification Problems**

Q. 1 What is classification? What are the components of classification problem? (Refer Section 3.1) **(5 Marks)**

Q. 2 What are the three phases of classification model ? (Refer Section 3.1.1) **(5 Marks)**

Q. 3 What are the main components of classification model ? (Refer Section 3.1.2) **(5 Marks)**

☞ **Syllabus Topic : Evaluation of Classification Models**

Q. 4 How you evaluate classification method? (Refer Section 3.2) **(5 Marks)**

Q. 5 Explain the Holdout method. (Refer Section 3.2.1) **(4 Marks)**

Q. 6 Explain the Repeated random sampling. (Refer Section 3.2.2) **(4 Marks)**

Q. 7 Explain the cross validation. (Refer Section 3.2.3) **(4 Marks)**

Q. 8 Explain the confusion matrices. (Refer Section 3.2.4) **(5 Marks)**

Q. 9 Explain the ROC curve chart. (Refer Section 3.2.5) **(5 Marks)**

Q. 10 Explain the Cumulative gain and lift chart. (Refer Section 3.2.6) **(5 Marks)**

☞ **Syllabus Topic : Bayesian Methods**

Q. 11 Write short note on Bayesian methods. (Refer Section 3.3) **(4 Marks)**

Q. 12 Explain naive Bayes classifier with example. (Refer Section 3.3.2) **(5 Marks)**

Q. 13 What is Bayesian networks ? (Refer Section 3.3.3) **(4 Marks)**

☞ **Syllabus Topic : Logistic Regression**

Q. 14 Write short note on logistic regression. (Refer Section 3.4) **(5 Marks)**

☞ **Syllabus Topic : Neural Networks**

Q. 15 Write short note on neural network. (Refer Section 3.5) **(5 Marks)**

☞ **Syllabus Topic : Support Vector Machines**

Q. 16 Write short note on support vector machine. (Refer Section 3.6) **(5 Marks)**

☞ **Syllabus Topic : Clustering Methods**

Q. 17 What are the characteristics of clustering method? (Refer Section 3.7.1) **(4 Marks)**

Q. 18 What is taxonomy of clustering method? (Refer Section 3.7.3) **(4 Marks)**

Q. 19 Write short note on Binary attribute. (Refer Section 3.7.5(1.)) **(5 Marks)**

Q. 20 Write short note on Nominal attribute. (Refer Section 3.7.5(2.)) **(4 Marks)**

Q. 21 Write short note on Ordinal attribute. (Refer Section 3.7.5(3.)) **(4 Marks)**

☞ **Syllabus Topic : Partition Methods**

Q. 22 Explain K-means method. (Refer Section 3.8.1) **(4 Marks)**

Q. 23 Explain *K*-medoids algorithm. (Refer Section 3.8.2) **(5 Marks)**

☞ **Syllabus Topic : Hierarchical Methods**

Q. 24 Explain single linkage, complete linkage, average linkage and ward distance. (Refer Section 3.9) **(5 Marks)**

☞ **Syllabus Topic : Evaluation of Clustering Models**

Q. 25 How one evaluates clustering model? (Refer Section 3.10) **(5 Marks)**

❑❑❑

Chapter Ends...