UNIT 4

UNSUPERVISED LEARNING AND ASSOCIATION RULES

SYLLABUS

Unsupervised Learning: Clustering Introduction, Similarity and Distance Measures, Agglomerative Algorithms, Divisive Clustering, Minimum Spanning Tree.

Association Rules: Introduction, Large Itemsets, Apriori Algorithm.

LEARNING OBJECTIVES

- Introduction to Clustering
- Application and Problems of Clustering Concept
- Classification of Clustering Algorithms
- Similarity and Distance Measures
- Agglomerative Algorithm and its Types
- Introduction to Divisive Clustering
- Spanning Tree and Minimum Spanning Tree
- Introduction to Association Rules
- Large Itemsets and its Algorithm
- Introduction to Apriori Algorithm.

INTRODUCTION

Clustering is a technique of combining a group of physical objects into classes of homogeneous objects. Clustering of data objects into single class is equivalent to data compression. Various fields such as data analysis, market research image processing, and pattern recognition uses the concept of cluster analysis.

Clustering can be done in two ways, one method is hierarchical clustering and other method is k-means clustering. A hierarchical clustering method creates a hierarchical structure from data objects. The hierarchical clustering methods can be classified into two types. They are Agglomerative clustering and Divisive clustering.

Association rules refer to the probability of customer purchasing one product when he purchases some other product. It is mainly used to show the relationships among various data items. It is also called as a rule of detecting common usage of items. The discovery of association rules in market basket analysis helps the retail stores to assist in marketing, advertising, designing store layout, inventory control, sales promotion strategies etc.

PART - A

SHORT QUESTIONS AND ANSWERS

Q1. What is unsupervised learning?

Answer :

(Model Paper-I, Q10 | June/Aug.-22, Q10 [OU])

Unsupervised learning is the process of training the machine through data that is not classified and labeled through algorithm to act on data without any guidance. The machine will group all the unordered data according to similarities, patterns and differences without prior training of data. The machine is not provided any type of training. So it cannot find the hidden structure in unlabeled data by itself. The unsupervised learning algorithms allows to perform more complex processing tasks. It can be more unpredicted compared to other natural learning, deep learning and reinforcement learning methods.

Q2. Write about Clustering.

Answer :

(Model Paper-II, Q10 | July/Aug.-22, Q4 [MGU])

Clustering is a technique of combining a group of physical objects into classes of homogeneous objects. That is, on the basis of 'similarity', huge data sets are partitioned into groups. It is known as 'data segmentation'. Clustering of data objects into single class is equivalent to data compression. Various fields such as data analysis, market research, image processing, pattern recognition, business, statistics and biology uses the concept of cluster analysis. Apart from this, outlier detection applications also use clustering, these include, the detection of credit card fraud and the monitoring of criminal activities in electronic commerce.

Q3. Classify the clustering algorithms.

Answer :

Model Paper-III, Q10

Clustering algorithms are classified as follows,

1. **Hierarchical Clustering**

It creates a nested set of clusters. It again divided into agglomerative and divisive algorithms.

2. **Partitional Clustering**

It creates only a particular set of clusters.

3. **Categorical Clustering**

It creates the clusters based on the category of data attributes.

4. **Large DB Clustering**

It creates the clusters by using the large databases. It again divided into sampling and compression clustering algorithms.

Q4. Define distance measures.

Answer :

Distance measure can be defined as a measure that helps in the clustering process to move the data points to a specific group based on the distance metrics. Consider dis(t_i, t_j) is the distance measure. Then, the desirable property of the clustering problem can be given as,

$$\forall\, t_{jl}, t_{jm} \in K_j \text{ and } t_i \notin K_j$$

$$\text{then, } dis(t_{jl}, t_{jm}) \leq dis(t_{jl}, t_i)$$

Generally, some of the algorithms performs clustering by using some metric data points, which inturn satisfies the triangular inequality. Thus, various characteristic values used to describe the cluster.

Q5. Define centroid and medoid.

Answer :

Centroid

Model Paper-I, Q11

It is the distance measured between the centroid of one cluster to centroid of another cluster. This can be represented as,

$$dis(K_i, K_j) = dis(C_i, C_j)$$

Here, C_i, C_j are the centroids for K_i and K_j respectively.

Medoid

It is the distance measured between the medoid of one cluster to medoid of another cluster. It can be represented as,

$$dis(K_i, K_j) = dis(M_i, M_j)$$

Here, M_i, M_j are the medoids for K_i and K_j respectively.

Q6. Define complete link algorithm.

Answer :

Model Paper-II, Q11

The complete link algorithm is also same as single link technique. The only difference is that it works on cliques instead of connected components. A clique can be defined as a maximal graph which consists of an edge among two vertices. In this algorithm, a specific procedure is used to merge two clusters if the maximum distance between them is less than or equal to distance threshold. The space and time complexity of this algorithm is $O(n^2)$.

This algorithm is more efficient than the single link algorithm because the clusters generated using this algorithm are more compact.

Q7. What is a divisive clustering and spanning tree?

Answer :

Model Paper-III, Q11

Divisive clustering

Divisive clustering is used to divide the data in a cluster into two parts repeatedly until, all the data points are pushed into their respective cluster. The main objective of this concept is used to divide the cluster which are not related or close to other elements in the cluster.

Spanning Tree

A spanning tree for a connected, undirected graph, $G = (V, E)$ is a subgraph of G that is an undirected tree and contains all the vertices of G. It can also be defined as "A sub graph $T = (V, E)$ is a spanning tree of G if T is a tree. A spanning tree of a graph should include all the vertices and a subset of edges (E)".

Q8. Discuss about various applications of association rules.

Answer :

(Model Paper-I, Q12 | June/Aug.-22, Q12 [OU])

The various applications of association rules are as follows,

1. They are used by retail stores to help in advertising, marketing, inventory control and floor placement.
2. They are used to detect the faults in telecommunication.
3. They are used to determine the relationship among the data items.

Q9. How to calculate support and confidence value for given itemsets.

Answer :

(Model Paper-II, Q12 | June/Aug.-22, Q11 [OU])

Support of Association Rule

Support of association rule is defined as proportions of transactions in the data set that contain a particular item set. Support is denoted by, sup.

Confidence of Association Rule

Confidence is calculated as support of two item sets R and S divided by the support of R.

OR

Confidence of association rule is nothing but probability of finding the R.H.S of association rule in the transactions with a condition that these transactions also include L.H.S of association rule.

The formula for confidence is,

$$\boxed{\frac{\sup(R \cup S)}{\sup(R)} = \text{conf}(R \Rightarrow S)}$$

$\text{conf}(R \Rightarrow S)$– Confidence of association rule

$$\sup(R \cup S) = \frac{\sup(R) + \sup(S)}{\text{Number of transactions in which } R \text{ or } S \text{ have occured more number of times}}$$

Q10. Write about Apriori-gen algorithm.

Answer :

Model Paper-III, Q12

The main objective of a priori gen algorithm is to produce the candidate item set for every level starting from first. Here, the candidate items are generated by joining the large itemsets (L) in the prior level i.e., L_{i-1} by itself. Thus, each independent itemset must have all items except one item that is in common to be combined.

Algorithm

Input:

L_{i-1} //Large itemsets of size $i-1$

Output:

C_i //Candidates of size i

Apriori-gen algorithm:

$C_i = \phi$;

for each $I \in L_{i-1}$ do

for each $J \neq I \in L_{i-1}$ do

if $i-2$ of the elements in I and J are equal then

$C_k = C_k \cup \{I \cup J\}$;

PART - B

ESSAY QUESTIONS AND ANSWERS

4.1 UNSUPERVISED LEARNING

Q11. Discuss about unsupervised learning.

Answer :

Unsupervised Learning

Unsupervised learning is the process of training the machine through data that is not classified and labeled through algorithm to act on data without any guidance. The machine will group all the unordered data according to similarities, patterns and differences without prior training of data. The machine is not provided any type of training. So it cannot find the hidden structure in unlabeled data by itself. The unsupervised learning algorithms allows to perform more complex processing tasks. It can be more unpredicted compared to other natural learning, deep learning and reinforcement learning methods. Unsupervised learning is classified into two categories of algorithms. They are as follows,

1. **Clustering**

   The clustering is useful to discover the inherent groupings in the data like grouping of customers by purchasing behavior.

2. **Association**

   An association rule learning problem is useful to discover the rules that describe large part of data like people who purchase X and Y.

4.1.1 Clustering Introduction

Q12. Explain in detail about clustering.

Answer : **Model Paper-II, Q16(a)**

Clustering

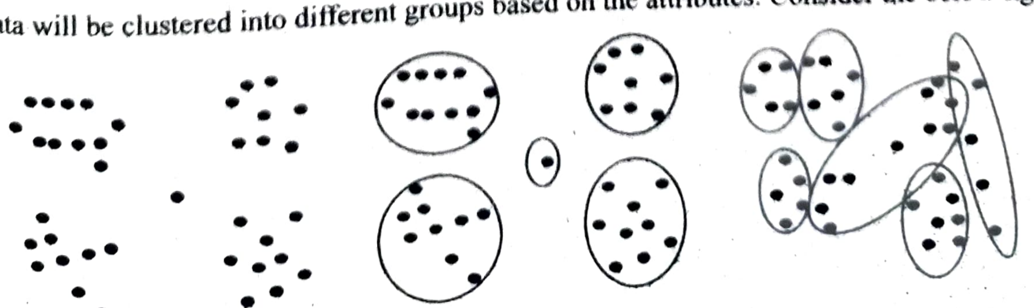
Clustering is a technique of combining a group of physical objects into classes of homogeneous objects. That is, on the basis of 'similarity', huge data sets are partitioned into groups. It is known as 'data segmentation' Clustering of data objects into single class is equivalent to data compression. Various fields such as data analysis, market research, image processing, pattern recognition, business, statistics and biology uses the concept of cluster analysis. Apart from this, outlier detection applications also use clustering, these include, the detection of credit card fraud and the monitoring of criminal activities in electronic commerce.

Database Segmentation

Database segmentation can be defined as grouping of rows or records in a database. This helps in providing a general view of the data in a database.

Different Clustering Attributes Example

The data will be clustered into different groups based on the attributes. Consider the below figure,



(a) Group of Homes (b) Geographic Distance-based (c) Size-based

In the above figures, figure (a) represents the clustering that is done on the location criteria, figure (b) represents the clustering that is done on the geographical location criteria, and figure (c) represents the clustering that is done based on the size.

Applications

The various applications where the concept of clustering being used are as follows,

1. Pattern recognition
2. Plant and animal classification
3. Image processing
4. Disease classification
5. Biological taxonomy
6. Document retrieval and
7. Examining the usage patterns from web log data.

Q13. List the various problems that may occur when the clustering is applied to real-world databases.

Answer :

The various problems that occur when the clustering applied to real-world databases are as follows,

1. **Frequent Updations in Data**

   The regular modifications in the data leads to the change in cluster membership over a period of time.

2. **Outlier Handling is Difficult**

   Generally, an element cannot directly moved into any cluster, initially it is viewed as a solitary cluster. The clustering may want the element to be pushed into a particular cluster but the outlier forcely pushes the element into another cluster. Surprisingly, this creates weak clusters.

3. **Interpretation of Cluster Name**

   Assigning the label or name to each cluster is difficult as the exact meaning of the cluster is unknown.

4. **Need of Domain Expert**

   There is always a requirement of domain expert inorder to know the number of clusters required.

Q14. Write short notes on the classification of clustering algorithms.

Answer :

The classification of clustering algorithms is as shown in the below figure,

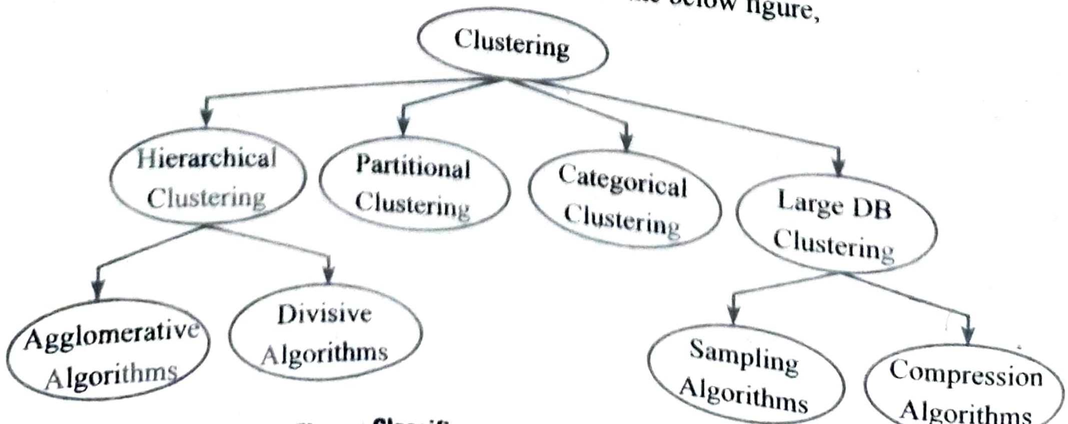


Figure: Classification of Clustering Algorithms

1. **Hierarchical Clustering**

It creates a nested set of clusters. It again divided into agglomerative and divisive algorithms.

2. **Partitional Clustering**

It creates only a particular set of clusters.

3. **Categorical Clustering**

It creates the clusters based on the category of data attributes.

4. **Large DB Clustering**

It creates the clusters by using the large databases. It again divided into sampling and compression clustering algorithms.

4.1.2 Similarity and Distance Measures

Q15. Explain about similarity and distance measures.

Answer :

Model Paper-II, Q16(b)

Similarity Measure

Similarity measure can be defined as a measure that helps in the clustering process to move the data points with similar features into a specific group. Technically, this can be defined as the similarity measure i.e., sim(t_1, t_2) between two tuples belongs to D i.e., $(t_1, t_2 \in D)$.

Distance Measure

Distance measure can be defined as a measure that helps in the clustering process to move the data points to a specific group based on the distance metrics. Consider dis(t_i, t_j) is the distance measure. Then, the desirable property of the clustering problem can be given as,

$$\forall\, t_{jl}, t_{jm} \in K_j \text{ and } t_i \notin K_j$$

then, $\text{dis}(t_{jl}, t_{jm}) \leq \text{dis}(t_{jl}, t_i)$

Generally, some of the algorithms performs clustering by using some metric data points, which inturn satisfies the triangular inequality. Thus, various characteristic values used to describe the cluster.

Consider K_m as a cluster with N points i.e., $\{t_{m1}, t_{m2}, t_{m3}, \ldots, t_{mN}\}$. Then, some of the characteristic values used are as follows,

(a) Centroid (C_m)

Centroid refers to the middle or centre of a cluster, which is not mandatory to be a central point in a cluster. But, some of the algorithms assume that the centrally located cluster data point represents the cluster.

$$\text{Centroid, } C_m = \frac{\sum_{i=1}^{N}(t_{mi})}{N}$$

(b) Radius (R_m)

Radius refers to the square root of average mean squared distance from centroid to any specific point in the cluster. This can be mathematically represented as,

$$\text{Radius, } R_m = \sqrt{\frac{\sum_{i=1}^{N}(t_{mi} - C_m)^2}{N}}$$

(c) Diameter (D_m)

Diameter refers to the square root of average mean squared distance between all paris of points in a cluster. This can be mathematically represented as,

$$\text{Diameter, } D_m = \sqrt{\frac{\sum_{i=1}^{N}\sum_{j=1}^{N}(t_{mi} - t_{mj})^2}{(N)(N-1)}}$$

Q16. Explain various standard alternatives to calculate the distance between clusters.

Answer :

Standard Alternatives to Calculate the Distance between Clusters

The various standard alternatives that are used to calculate the distance between the clusters are as follows,

1. **Single Link**

   It is the minimum distance measured between the element in one cluster to another element in another cluster. This can be represented as,

   $dis(K_i, K_j) = \min(dis(t_{il}, t_{jm}))\ \forall\ t_{il} \in K_i \notin K_j$ and $\forall\ t_{jm} \in K_j \notin K_i$

2. **Complete Link**

   It is the maximum distance measured between the element in one cluster to another element in another cluster. This can be represented as,

   $dis(K_i, K_j) = \max(dis(t_{il}, t_{jm}))\ \forall\ t_{il} \in K_i \notin K_j$ and $\forall\ t_{jm} \in K_j \notin K_i$

3. **Average Link**

   It is the average distance measured between the element in one cluster to another element in another cluster. This can be represented as,

   $dis(K_i, K_j) = \text{mean}(dis(t_{il}, t_{jm}))\ \forall\ t_{il} \in K_i \notin K_j$ and $\forall\ t_{jm} \in K_j \notin K_i$

4. **Centroid**

   It is the distance measured between the centroid of one cluster to centroid of another cluster. This can be represented as,

   $dis(K_i, K_j) = dis(C_i, C_j)$

   Here, C_i, C_j are the centroids for K_i and K_j respectively.

5. **Medoid**

   It is the distance measured between the medoid of one cluster to medoid of another cluster. It can be represented as,

   $dis(K_i, K_j) = dis(M_i, M_j)$

   Here, M_i, M_j are the medoids for K_i and K_j respectively.

4.1.3 Agglomerative Algorithms, Divisive Clustering

Q17. Explain agglomerative algorithms.

Answer :

Agglomerative algorithms are used to merge the small clusters into a big/large cluster. Initially, it starts with every object forming a separate group and then successively combines the objects or groups that are near to one another until all groups are combined into one termination condition is satisfied. This method is also referred to as 'bottom-up' approach.

Algorithm

The agglomerative algorithm is as given below,

Input: $D = \{t_1, t_2, t_3, \ldots, t_n\}$ //It represents a set of data elements

A //It represents an adjacency matrix with distance between elements i.e., $A[i, j] = dis(t_i, t_j)$

Output: DE //It represents a dendrogram which is a set of ordered triples, i.e., $\langle d, k, K\rangle$

//Here, d represents threshold distance

k represents number of clusters

K represents the set of clusters.

$d = 0;$

$k = n;$

$K = \{\{t_1\}, \{t_2\}, \ldots, \{t_n\}\};$

$DE = \{\langle d, k, K\rangle\};$

repeat

  old$k$ = k;

  $d = d + 1$;

  A_d = The vertex adjacency matrix with d as threshold distance d

  $\langle k, K\rangle$ = NewClusters (A_d, D);

  if old$k \neq K$ then

    $DE = DE \cup \langle d, k, K\rangle$;   //This represents a new cluster that is added to dendrogram.

    until $k = 1$;

The above algorithm calls a procedure called New Cluster which creates new clusters by using the clusters in the prior level. There may be a case that what are the clusters from the prior level need to be merged. This can be done by using the various algorithms. The various agglomerative algorithms used to find the distance between the clusters are as follows,

1. Single Link Technique/Algorithm

The single link technique is used to determine the maximal connected components in a graph. Here, the connected graph defines a graph which has a connection between any two of its vertices.

This technique is used to merge two clusters when there exists an edge that connects two clusters which means that the minimum distance between any two individual clusters is less than or equal to the given threshold distance.

Algorithm

Input: $D = \{t_1, t_2, t_3, \ldots, t_n\}$ //It represents a set of data elements

  A //It represents an adjacency matrix with distance between elements i.e., $A[i, j] = \text{dis}(t_i, t_j)$

Output: DE //It represents a dendrogram which is a set of ordered triples, i.e., $\langle d, k, K\rangle$

  //Here, d represents threshold distance

  k represents number of clusters

  K represents the set of clusters.

  $d = 0$;

  $k = n$;

  $K = \{\{t_1\}, \ldots, \{t_n\}\}$;

  $DE = \{\langle d, k, K\rangle\}$;

  $M = MST(A)$;

repeat

  old$k$ = k;

  K_i, K_j = Two clusters closest together in MST;

  $K = K - \{K_i\} - \{K_j\} \cup \{K_i \cup K_j\}$;

  k = old$k$ − 1;

  $d = \text{dis}(K_i, K_j)$;

  $DE = DE \cup \langle d, k, K\rangle$; //This represents a new cluster that is added to dendrogram.

  $\text{dis}(K_i, K_j) = \infty$

  until $k = 1$;

2. Complete Link Algorithm

The complete link algorithm is also same as single link technique. The only difference is that it works on cliques instead of connected components. A clique can be defined as a maximal graph which consists of an edge among two vertices. In this algorithm, a specific procedure is used to merge two clusters if the maximum distance between them is less than or equal to distance threshold. The space and time complexity of this algorithm is $O(n^2)$.

This algorithm is more efficient than the single link algorithm because the clusters generated using this algorithm are more compact.

3. Average Link Algorithm

The average link algorithm used to combine two clusters when the average distance between the data points of a cluster to data point of another cluster is less than the threshold distance.

Algorithm

Input: $D = \{t_1, t_2, t_3, \ldots, t_n\}$ //It represents a set of data elements

A //It represents an adjacency matrix with distance between elements i.e., $A[i, j] = dis(t_i, t_j)$

Output: DE //It represents a dendrogram which is a set of ordered triples, i.e., $\langle d, k, K\rangle$

//Here, d represents threshold distance

k represents number of clusters

K represents the set of clusters.

$d = 0;$

$k = n;$

$K = \{\{t_1\}, \ldots, \{t_n\}\};$

$DE = \{\langle d, k, K\rangle\};$

repeat

oldk $= k;$

$d = d + 0.5;$

for each pair of $K_i, K_j \in K$ do

ave = average distance between all $t_i \in K_i$ and $t_j \in K_j$;

if ave $< d$, then

$K = K - \{K_i\} - \{K_j\} \cup \{K_i \cup K_j\};$

$k =$ oldk $- 1;$

$DE = DE \cup \langle d, k, K\rangle;$ //This represents a new cluster that is added to dendrogram.

until $k = 1;$

In the above algorithm, the value of d is increased by 0.5 instead of 1 as it is an arbitrary decision.

Q18. Explain Agglomerative Algorithms and Divisive Clustering.

Answer :

(Model Paper-I, Q16(a) | July/Aug.-22, Q13 [MGU])

Agglomerative Algorithms

For answer refer Unit-IV, Page No. 80, Q.No. 17.

Divisive Clustering

Divisive clustering is used to divide the data in a cluster into two parts repeatedly until, all the data points are pushed into their respective cluster. The main objective of this concept is used to divide the cluster which are not related or close to other elements in the cluster.

Example

Consider the below figure,

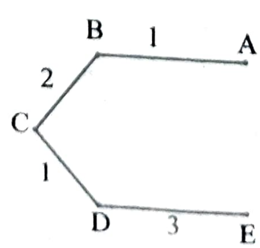


Figure: Graph

In the above graph, initially all the elements are available in single cluster i.e., {A, B, C, D, E}. As per the MST version of single link technique, the maximum edge {D, E} will be split into {A, B, C, D}, and {E}. Then the next large edge BC is removed inturn dividing the one large cluster into two clusters i.e., {A, B}, {C, D}.

In the similar way, the cluster is divided until the requirement is satisfied.

4.1.4 Minimum Spanning Tree

Q19. Define spanning tree and minimum spanning tree. Discuss about partitional MST algorithm.

Answer :

Spanning Tree

A spanning tree for a connected, undirected graph, $G = (V, E)$ is a subgraph of G that is an undirected tree and contains all the vertices of G. It can also be defined as "A sub graph $T = (V, E)$ is a spanning tree of G if T is a tree. A spanning tree of a graph should include all the vertices and a subset of edges (E)".

Minimum Spanning Tree (MST)

A minimum spanning tree is a spanning tree, that has weights with the edges and the total weight of the tree (the sum of the weights of its edges) as minimum.

Partitional MST Algorithm

The result of partitional MST algorithm represents the clusters in the form of ordered pairs (t_i, j) where $f(t_i) = K_j$.

Input: D, A, k

$D = \{t_1, t_2, \ldots, t_n\}$

Where k is the number of clusters required.

Output: f (set of ordered pairs representing the mapping).

$M = \text{MST}(A)$

Determine inconsistent edges existing in M

Eliminate $k - 1$ inconsistent edges.

Represent the output format.

The inconsistent edges can be determined by considering the distance. However, a better approach suggested by *Zahn* is to consider edge weight. The weight of an inconsistent edge would be larger than the average adjacent edges. The partitional MST algorithm involves a time complexity of $O(n^2)$.

Q20. What is spanning tree? Explain how to calculate least cost using minimal spanning tree with example.

Answer : (Model Paper-III, Q16(a) | June/Aug.-22, Q16(a) [OU])

Spanning Tree

For answer refer Unit-IV, Page No. 83, Q.No. 19, Topic: Spanning Tree.

Least Cost Using Minimal Spanning Tree

Least cost using minimal spanning tree can be done by Kruskal's algorithm or Prim's algorithm.

1. **Kruskal's Algorithm**

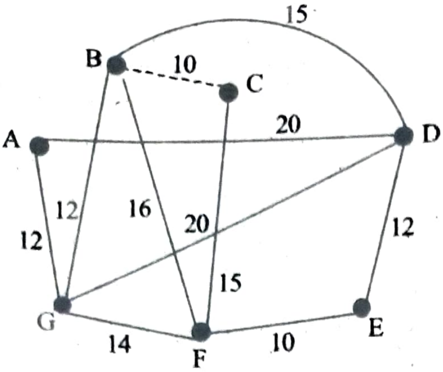
Kruskal's algorithm can be used for calculating least cost using minimal spanning tree.

Kruskal's algorithm chooses an edge in the graph with minimum weight. Successively the algorithm adds edges with minimum weight that do not form a simple circuit with those edges already chosen. Once $n - 1$ edges have been selected, stop the algorithm.

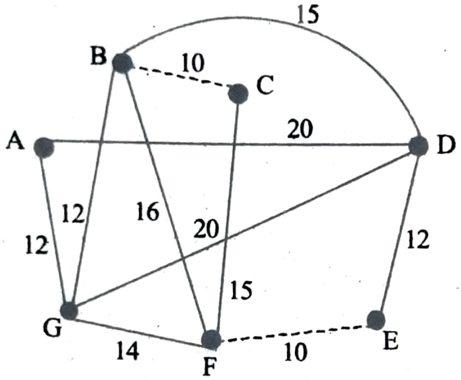
Algorithm

```
kruskal_Alg
//Input : A weighted connected graph G = (V, E)
//Output : Tree, the minimum spanning tree of G
{   Edge
    Sort E in increasing order by weight
    Tree : ={ }          //empty graph
    for i = 1 to n - 1
    {
        e : = any edge in G with smallest weight that does not form a simple circuit when added to Tree
        T = T ∪ {e}      //add e to Tree
    }
    return Tree;
}
```

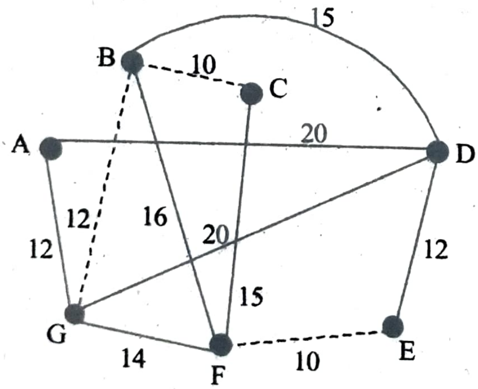
Example



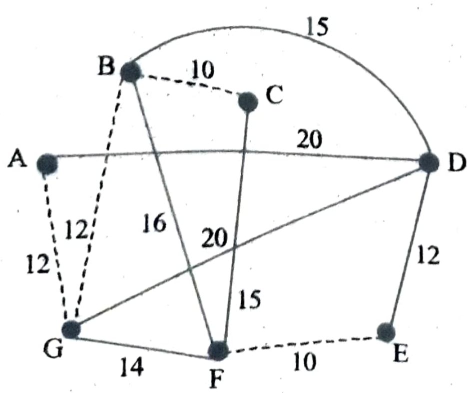
Choose e = BC



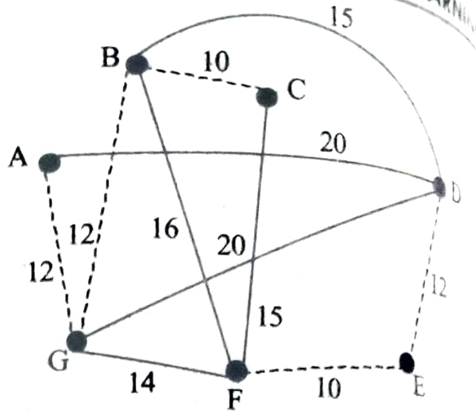
Choose e = FE



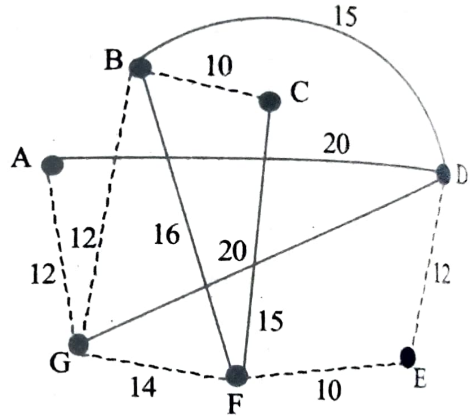
Choose e = GB



Choose e = AG



Choose e = ED



Choose e = GF

According to step 1 of Kruskal's algorithm, se... $e_1 = BC$ as it has the minimum weight. Now, impleme... step 2 of Kruskal's algorithm by recursively selec... the next edge having the least weight until $|E| = n-1$

So, the minimum spanning tree is,

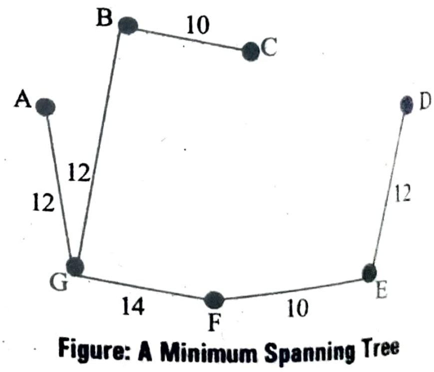


Figure: A Minimum Spanning Tree

2. Prim's Algorithm

Prim's algorithm can be used for calculating lea... cost of minimum spanning tree. This algorithm beg... by initially selecting a vertex as a tree node. It th... connects the vertex edge with another nearest ver... by comparing the edge weights of each vertex that ... connecting it. After this, the nearest vertex is considered ... and the nodes of the graph are then added to the tree (... at a time) by following the above process. This proce... continues until all the nodes of the graph are added ... the tree. Hence, the obtained tree is considered as ... minimum spanning tree.

The steps to find minimum spanning tree using prim's algorithm are as follows,

Step 1: Select a vertex v_1 of G. Let $V = \{v_1\}$ and $E = \{\ \}$.

Step 2: Select a nearest neighbor v_i of V that is adjacent to v_j, $v_j \in V$ and for which the edge (v_i, v_j) does not form a circuit with members of E. Add v_i to V and (v_i, v_j) to E respectively.

Step 3: Repeat step 2, until $|E| = n - 1$. Then V contains all n vertices of G and E contains the edges of a minimum spanning tree for G.

Pseudocode for Prim's Algorithm

Prim's algorithm for the graph's that are represented by adjacency lists is given below,

Procedure prim (G_{con})

//V_{mst} is vertices of minimum spanning tree

Initialize E_{mst} to (v_0)

//E_{mst} is the edge of minimum spanning tree

Initialize E_{mst} to ϕ

for i 1 to $|v| - 1$ do

Determine a min-weight edge $e^* = (v^*, u^*)$ from all the edge (v, u), where v is in V_{mst} and u is in $v - V_{mst}$

$V_{mst} \leftarrow V_{mst} \cup \{u^*\}$

$E_{mst} \leftarrow E_{mst} \cup \{e^*\}$

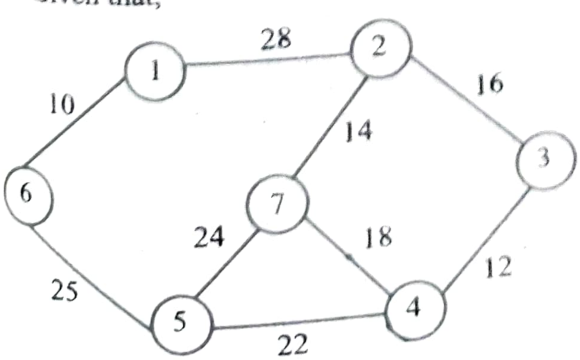
return E_{mst}

The above pseudocode is used to construct a minimum spanning tree using prim's method. The working of prim's algorithm is as follows,

This algorithm takes a weighted connected graph G_{con} as its input and returns a set of edges which form the minimum spanning tree of the graph G_{con}. Here, G_{con} is given as $G_{con} = \{V, E\}$.

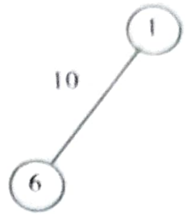
Example

Given that,

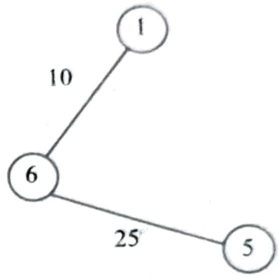


The step-by-step procedure is followed to find minimum spanning tree using prim's algorithm.

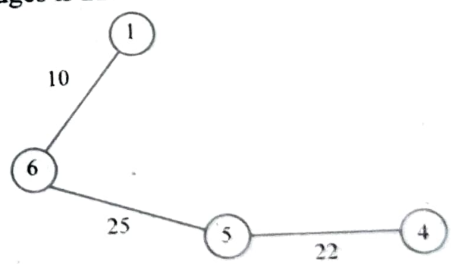
1. Choose one random vertex and the least connected edge to that particular vertex. Vertex 1 has 2 connected edges with 10 and 28. As 10 is comparatively smaller, the edge with length 10 is selected.



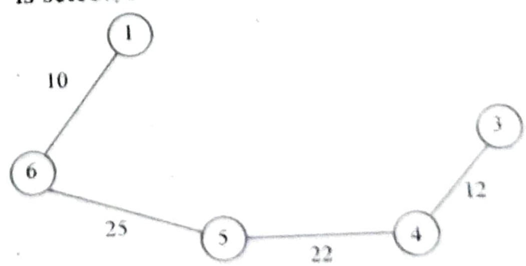
2. Check for least edges to the second vertex. Vertex 6 has only one possible edge with length 25 to move further.



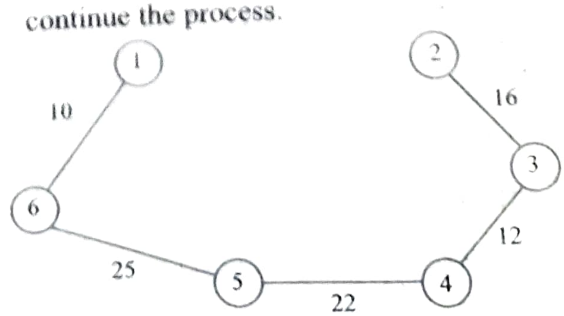
3. Continue the process till all the vertices are connected without forming a cycle. Vertex 5 has 2 connected edges 24 and 22. The least among 2 edges is 22.



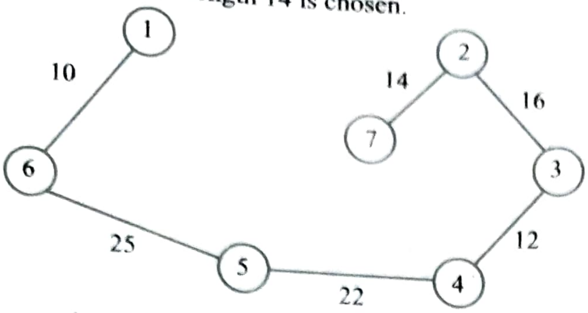
4. In between 18 and 12, 12 is the least edge. So, it is selected.



5. The vertex 3 has only one edge of length 16 to continue the process.



6. If the vertex 2 chooses edge with 28, it forms a cycle and doesn't connect all the vertices. So, edge with length 14 is chosen.



As all the vertices are connected without forming a cycle.

$$\text{Cost} = 10 + 25 + 22 + 12 + 16 + 14 = 99$$

$\therefore$ Cost of minimum spanning tree is 99.

4.2 ASSOCIATION RULES

4.2.1 Introduction, Large Itemsets

Q21. Explain about association rules.

Answer :

Model Paper-III, Q16(b)

Association Rules

Association rules refer to the probability of customer purchasing one product when he purchases some other product. It is mainly used to show the relationships among various data items. It is also called as a rule of detecting common usage of items. The discovery of association rules in market basket analysis helps the retail stores to assist in marketing, advertising, designing store layout, inventory control, sales promotion strategies etc. For instance, if customers are buying sugar how likely are they also buy tea (and what kind of tea) on the same trip to the store. By this informative strategy the retailer can get an idea of selective marketing and plan their shelf space which increases the sales.

Association rule can be mathematically represented as, consider I is a set of items i.e., $\{I_1, I_2, \ldots, I_n\}$ and a database $D = \{t_1, t_2, \ldots, t_n\}$ where $t_i = \{I_{i1}, I_{i2}, \ldots, I_{iK}\}$ and $I_{ij} \in I$, then association rule is an implication of the for $X \Rightarrow Y$. (Here, $X, Y \subset I$ and $X \cap Y = \phi$).

Applications

The various uses of association rules are as follows,

1. They are used by retail stores to help in advertising, marketing, inventory control and floor placement.

2. They are used to detect the faults in telecommunication.

3. They are used to determine the relationship among the data items.

The importance of association rules can be defined using various features. They can be defined as follows,

Example

Suppose there is a stationary database in which 5 transactions are carried out for 4 items. Assume the items as pencil, scale, eraser, sharpener. The transactional process is shown in a tabular format as follows,

		R		S	
Transaction	T/D	Pencil	Eraser	Scale	Sharpener
tr_1	1	A	A	U	U
tr_2	2	U	A	A	U
tr_3	3	U	U	U	A
tr_4	4	A	A	A	U
tr_5	5	U	A	U	U

A-Available
U-Unavailable

In the first transaction, pencil and eraser are available and scale and sharpener are unavailable for purchase. Similarly, in the second transaction, pencil and sharpener are unavailable, eraser and scale are available for purchase and so on. After processing all the transactions, it can be included that association rule is,

$$\{\text{pencil, eraser}\} \Rightarrow \{\text{scale}\}$$

This means that if customers buy a pencil and eraser, they are also likely to buy a scale. Since the probability that scale is available is more than the sharpener.

Certain measurements are used for association rule.

1. Support of association rule
2. Confidence of association rule
3. Lift of association rule
4. Conviction of association rule.

1. Support of Association Rule

Support of association rule is defined as proportions of transactions in the data set that contain a particular item set. Support is denoted by, sup.

Hence, in our example, sup(R) = sup (pencil, scale) = $\frac{2}{5}$ = 0.4, this implies that pencil and scale are present simultaneously in 40% of the transactions.

2. **Confidence of Association Rule**

Confidence is calculated as support of two item sets R and S divided by the support of R.

(or)

Confidence of association rule is nothing but probability of finding the R.H.S of association rule in the transactions with a condition that these transactions also include L.H.S of association rule.

The formula for confidence is,

$$\boxed{\frac{\sup(R \cup S)}{\sup(R)} = \text{conf}(R \Rightarrow S)}$$

$\text{conf}(R \Rightarrow S)$ – Confidence of association rule

$$\sup(R \cup S) = \frac{\sup(R) + \sup(S)}{\text{Number of transactions in which } R \text{ or } S \text{ have occured more number of times}}$$

$$= \frac{0.4 + 0.4}{3} = \frac{0.8}{3}$$

$$= 0.2 \text{ (Approximately)}$$

$$\sup(R) = \sup(\text{pencil, scale}) = 0.4$$

$$\therefore \quad \text{conf(pencil, eraser} \Rightarrow \text{scale)} = \frac{0.2}{0.4} = 0.5$$

That implies that the association rule is correct for 50% of the transactions.

3. **Lift of Association Rule**

Lift is defined as a support of union of two item sets R and S divided by the product of $\sup(R)$ and $\sup(S)$.

(or)

Lift is defined as the ratio of observed confidence to that expected by chance.

The formula for lift is,

$$\boxed{\text{Lift}(R \Rightarrow S) = \frac{\sup(R \cup S)}{\sup(R) * \sup(S)}}$$

For an example,

Lift(pencil eraser $\Rightarrow$ scale)

$$= \frac{\sup(\text{pencil, eraser} \cup \text{scale})}{\sup(\text{pencil , eraser}) * \sup\text{ (scale)}}$$

$$= \frac{0.2}{0.4 * 0.4}$$

$$= 1.25$$

4. **Conviction of Association Rule**

The conviction is calculated as subtracting support of item set S ($\sup(S)$) and confidence of the association rule ($\text{conf}(R \Rightarrow S)$) from one and dividing their results.

(or)

Ratio of the expected frequency that R occurs without S with the condition that they were independent of the observed frequency.

The formula for conviction is,

$$\frac{1-\sup(S)}{1-\operatorname{conf}(R \Rightarrow S)} = \operatorname{conv}(R \Rightarrow S)$$

$\operatorname{conv}(R \Rightarrow S)$ Conviction of association rule.

Applying this formula in our example yields,

$$\frac{1-0.4}{1-0.5} = \frac{0.6}{0.5} = 1.2$$

Q22. Explain about large itemsets along with its algorithm.

Answer :

Large or frequent itemset can be defined as an itemset whose number of occurrences is more than thresho distance(s). In this, a full set of large itemsets is represented by using 'L' and a particular large itemset in 'L' is represented by using 'l'.

The association rules can be determined by splitting the problem into two parts. The yare as follows,

1. Initially, determine the large itemsets.
2. Then, generate the rules from these itemsets.

Notations

The various notations used in the concept of association rules are as given in the below table,

S.No.	Notation	Description
1.	D	It represents the Database of various transactions.
2.	t_i	It represents a specific transaction in database, D.
3.	s	It represents the support.
4.	α	It represents the confidence
5.	X, Y	They represents two itemsets.
6.	$X \Rightarrow Y$	It represents the association rule
7.	L	It represents a full set of large itemsets.
8.	l	It represents a specific large itemset in L
9.	C	It represents a set of candidate itemsets
10.	p	It represents the count of partitions.

Here, the superscript and subscript represents the portion and item size respectively. For example, l_k represents the large itemset with size of k elements where as D^i represents the i^{th} partition of database (D).

Suppose, if the size of an itemset is m, then it consists of 2^m subsets. As the empty set is not being in consideration, eventually the count of subsets will be $2^m - 1$. The potential large itemsets are called as candidates and a full set of potential large itemsets (or candidates) refers to candidate itemset.

As soon as the first part (i.e., determining the large itemsets) is completed, the association rules can be determined by using the following algorithm.

Algorithm

Input: $D, I, L, s, \propto$

Output: R //It represents association rules which satisfies support and confidence features.

ARGEN Algorithm:

$R = \phi$;

for each $l \in L$ do:

for each $x \subset l$ such that $x \neq \phi$ do

if $\frac{\text{support } (l)}{\text{support } (x)} \geq \alpha$ then

$R = R \cup \{x \Rightarrow (l - x)\}$;

4.2.2 Apriori Algorithm

Q23. Explain Apriori algorithm in detail. July/Aug.-22, Q14 [MGU]

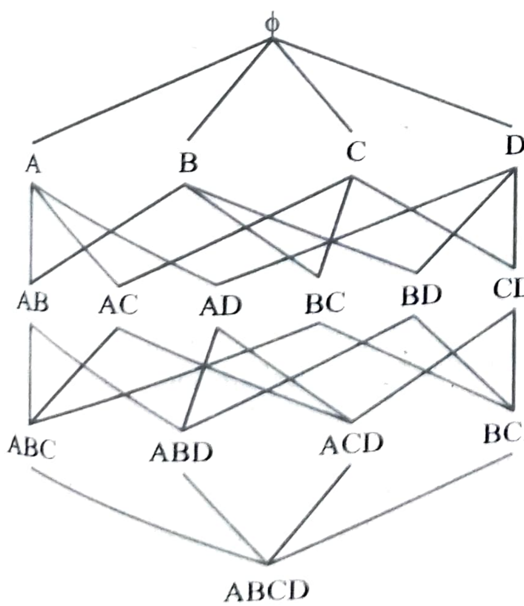
OR

Explain Apriori algorithm with suitable examples.

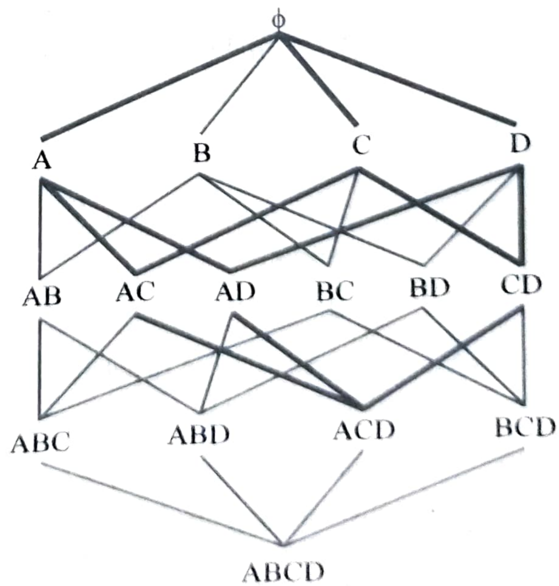
Answer : (Model Paper-I, Q16(b) | June/Aug.-22, Q16(b) [OU])

Apriori algorithm is a famous association rules algorithm that mostly deals with the commercial products. It makes use of a property called - "large itemset property" which states that - "The subset of a large item set should also be large".

Generally, the large item sets are considered as downward closed because the minimum support requirements followed by large item sets will also be followed by their subsets. In contrast, the subsets of small items are already small. Consider the following figures,



(a) Lattice of Itemsets for {A, B, C, D}



(b) Subsets of ACD

Figure: Downward Closure

In figure (a), it consists of four items, $\{A, B, C, D\}$ and its subsets represented with lines. In this case, the large item set property concludes that the subsets of the large items sets i.e., AB, AC, AD, BC, BD and CD are large.

In figure (b), the non-empty subsets of ACD are given as, $\{A, C, D, AC, AD, CD\}$. Hence, these subsets of ACD are large because the ACD is large.

There exists two different versions of apriori algorithms. They are as follows,

1. **Apriori Algorithm**

The main objective of this algorithm is to produce candidate item sets of a specific size and then examine the database to check whether they are large or not. In this algorithm, an item set is considered as candidate only when its subsets are large.

Algorithm

Input: I, D, s

Output: L

Algorithm:

$k = 0$; // k represents the scan number

$L = \phi$;

$C_1 = I$; //C_1 represents the initial candidates

repeat

$k = k + 1$;

$L_k = \phi$;

for each $I_i \in C_k$ do

$C_i = 0$ //C_i represents the count of every item set to 0.

for each $t_j \in D$ do

for each $I_i \in C_k$ do

if $I_i \in t_j$ then

$C_i = C_i + 1$;

for each $I_i \in C_k$ do

if $C_i \geq (s \times |D|)$ do

$L_k = L_k \cup I_i$;

$L = L \cup L_k$;

C_{k+1} = Apriori-Gen(L_k)

until $C_{k+1} = \phi$;

Note that, as the number of scans in database increase, the algorithm become less efficient. This is considered as the limitation of apriori algorithm.

2. **Apriori-Gen Algorithm**

The main objective of this algorithm is to produce the candidate item set for every level starting from first. Here, the candidate items are generated by joining the large itemsets (L) in the prior level i.e., L_{i-1} by itself. Thus each independent itemset must have all items except one item that is in common to be combined.

Algorithm

Input:

L_{i-1} //Large itemsets of size $i - 1$

Output:

C_i //Candidates of size i

Apriori-gen algorithm:

$C_i = \phi$;

for each $I \in L_{i-1}$ do

for each $J \neq I \in L_{i-1}$ do

if $i - 2$ of the elements in I and J are equal then

$C_k = C_k \cup \{I \cup J\}$;

Example

Transaction	Items
t_1	Blouse
t_2	Shoes, Skirt, TShirt
t_3	Jeans, TShirt
t_4	Jeans, Shoes, TShirt
t_5	Jeans, Shorts
t_6	Shoes, TShirt
t_7	Jeans, Skirt
t_8	Jeans, Shoes, Shorts, TShirt
t_9	Jeans
t_{10}	Jeans, Shoes, TShirt
t_{11}	TShirt
t_{12}	Blouse, Jeans, Shoes, Skirt, TShirt
t_{13}	Jeans, Shoes, Shorts, TShirt
t_{14}	Shoes, Skirt, TShirt
t_{15}	Jeans, TShirt
t_{16}	Skirt, TShirt
t_{17}	Blouse, Jeans, Skirt
t_{18}	Jeans, Shoes, Shorts, TShirt
t_{19}	Jeans
t_{20}	Jeans, Shoes, Shorts, TShirt

Table (1)

Scan	Candidates	Large Itemsets
1	{Blouse}, {Jeans }, {Shoes}, {Shorts}, {Skirt}, {TShirt}	{Jeans }, {Shoes}, {Shorts} {Skirt}, {Tshirt}
2	{Jeans, Shoes}, {Jeans, Shorts}, {Jeans, Skirt}, {Jeans, TShirt}, {Shoes, Shorts}, {Shoes, Skirt}, {Shoes, TShirt}, {Shorts, Skirt}, {Shorts, TShirt}, {Skirt, TShirt}	{Jeans, Shoes}, {Jeans, Shorts}, {Jeans, TShirt}, {Shoes, Shorts}, {Shoes, TShirt}, {Shorts, TShirt}, {Skirt, TShirt}
3	{Jeans, Shoes, Shorts}, {Jeans, Shoes, TShirt}, {Jeans, Shorts, TShirt}, {Jeans, Skirt, TShirt}, {Shoes, Shorts, TShirt}, {Shoes, Skirt, TShirt}, {Shorts, Skirt, TShirt}	{Jeans, Shoes, Shorts}, {Jeans, Shoes, TShirt}, {Jeans, Shorts, TShirt},{Shoes, Shorts, TShirt}
4	{Jeans, Shoes, Shorts, TShirt}	{Jeans, Shoes, Shorts, TShirt}
5	ϕ	ϕ

Table (2)

In the above example of woman's clothing store that has 10 cash register transactions during one day, as shown in Table (1). When Apriori is applied to the data, during scan one, we have six candidate itemsets, as seen in Table (2). Of these, 5 candidates are large. When Apriori-Gen is applied to these 5 candidates, we combine every one with all the other 5. Thus, we get a total of 4 + 3 + 2 + 1 = 10 candidates during scan two. Of these, 7 candidates are large. When we apply Apriori-Gen at this level, we join any set with another set that has one item in common. Thus, {Jeans, Shoes} is joined with {Jeans, Shorts} but not with {Shorts, TShirt} . {Jeans, Shoes} will be joined with any other itemset containing either Jeans or Shoes. When it is joined, the new item is added to it. There are four large itemsets after scan four. When we go to join these we must match on two of the three attributes. For example {Jeans, Shoes, Shorts} After scan four, there is only one large itemset. So we obtain no new itemsets of size five to count in the next pass. Joins with {Jeans, Shoes, TShirt} to yield new candidate {Jeans, Shoes, Shorts, TShirt}.

This algorithm concludes that, it produces a superset of large itemsets with size i, $C_i \supset L_i$ when the large itemset in the prior level is L_{i-1}.

III. Very Short Questions and Answers

Q1. Define database segmentation.

Answer :

Database segmentation can be defined as grouping of rows or records in a database. This helps in providing a general view of the data in a database.

Q2. Define similarity measure.

Answer :

Similarity measure can be defined as a measure that helps in the clustering process to move the data points with similar features into a specific group.

Q3. What is average link algorithm?

Answer :

The average link algorithm used to combine two clusters when the average distance between the data points of a cluster to data point of another cluster is less than the threshold distance.

Q4. What is minimum spanning tree?

Answer :

A minimum spanning tree is a spanning tree, that has weights with the edges and the total weight of the tree (the sum of the weights of its edges) as minimum.

Q5. Define Apriori algorithm.

Answer :

Apriori algorithm is a famous association rules algorithm that mostly deals with the commercial products. It makes use of a property called - "large itemset property" which states that - "The subset of a large item set should also be large".