CBCS SCHEME

USN | 1 | C | 2 | 2 | 0 | 1 | S | [redacted] | | | 18CS641

Sixth Semester B.E. Degree Examination, June/July 2023
Data Mining and Data Warehousing

Time: 3 hrs. Max. Marks: 100

Note: Answer any FIVE full questions, choosing ONE full question from each module.

Important Note : 1. On completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages.
2. Any revealing of identification, appeal to evaluator and /or equations written eg, 42+8 = 50, will be treated as malpractice.

Module-1

1 a. Define Datawarehouse. Explain Multitier Architecture of Data Warehousing with diagram. **(10 Marks)**

b. Explain ETL process with the neat diagram. **(10 Marks)**

OR

2 a. Explain the schemas of multi-dimensional data models. **(10 Marks)**

b. Explain data cube operations with example. **(10 Marks)**

Module-2

3 a. Explain different methods of indexing OLAP data. **(10 Marks)**

b. Explain the following preprocessing techniques :
(i) Feature subset selection. (ii) Sampling. **(10 Marks)**

OR

4 a. List the different types of Dataset and explain with an example. **(10 Marks)**

b. Consider X = (0, 1, 0, 1), Y = (1, 0, 1, 0). Find cosine, correlation, Euclidean, Jaccard and SMC. **(10 Marks)**

Module-3

5 a. A database has five transactions. Let min sup = 60% and min conf = 80%.

Table Q5 (a)

TID	Items.bought
T_{100}	{M,O,N,K,E,Y}
T_{200}	{D,O,N,K,E,Y}
T_{300}	{M,A,K,E}
T_{400}	{M,U,C,K,Y}
T_{500}	{C,O,O,K,I,E}

(i) Find all frequent itemsets using Apriori Algorithm.
(ii) List all the strong association Rules. **(12 Marks)**

b. Identify and explain the alternative methods for generating frequent itemsets. **(08 Marks)**

OR

6 a. Construct FP tree by showing tree separately after reading each transaction and find the frequent itemset generation. Consider the transaction dataset :

Table : Q6 (a)

TID	Items.bought
100	{f, a, c, d, g, i, m, p}
200	{a, b, c, f, l, m, o}
300	{b, f, h, j, o}
400	{b, c, k, s, p}
500	{a, f, c, e, l, p, m, n }

Let min-support = 3 **(12 Marks)**

b. Explain the various measures of evaluating association patterns. **(08 Marks)**

Module-4

7 a. Explain the general approach for solving classification problem. (08 Marks)

b. Build a decision tree using Hun't Algorithm for the given dataset.

Tid	Home owner	Marital status	Annual Income	Defaulted Borrower
1	yes	Single	125 K	No
2	No	Married	100 K	No
3	No	Single	70 K	No
4	Yes	Married	120 K	No
5	No	Divorced	95 K	Yes
6	No	Married	60 K	No
7	yes	Divorced	220 K	No
8	No	Single	85 K	Yes
9	No	Married	75 K	No
10	No	Single	90 K	Yes

Table Q7 (b) (12 Marks)

OR

8 a. Explain K-nearest neighbor classification algorithm with example. (08 Marks)

b. State Bays theorem and explain how bayes theorem is used in the Naïve Bayesian classifier with example. (12 Marks)

Module-5

9 a. Calculate the cluster for the following 8 points (x, y). Represents into 3 clusters $A_1(2, 10)$ $A_2(2, 5)$ $A_3(8, 4)$ $B_1(5, 8)$ $B_2(7, 5)$ $B_3(6, 4)$ $C_1(1, 2)$ $C_2(4, 9)$ (10 Marks)

b. Explain the following :

(i) Density based clustering.

(ii) Graph based clustering. (10 Marks)

OR

10 a. What are the basic approaches used for generating a agglomerative hierarchical clustering? (10 Marks)

b. Explain DBSCAN algorithm with example. (10 Marks)

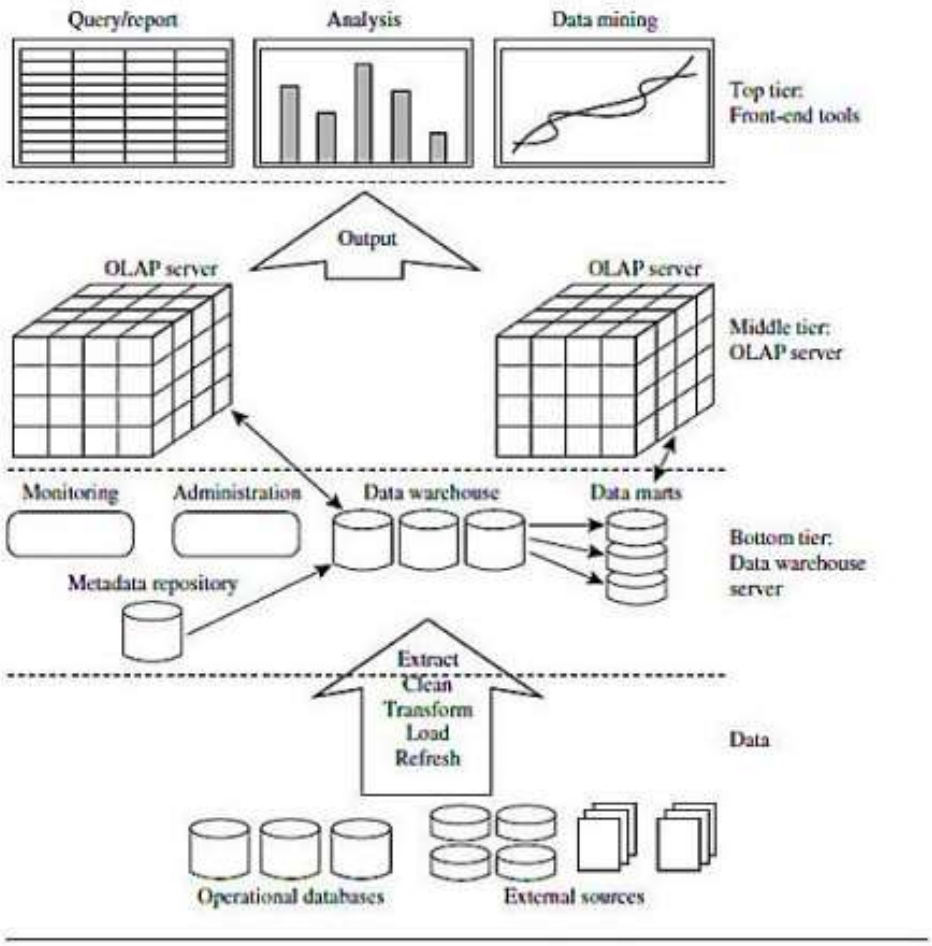
* * * * *

Module 1

Solutions:

1.a.

Data warehousing provides architectures and tools for business executives to systematically organize, understand, and use their data to make strategic decisions. A data warehouse is a subject-oriented, integrated, time-variant, and non-volatile collection of data in support of management's decision making process.



• The bottom tier is a warehouse database server that is almost always a relational database system. Back-end tools and utilities are used to feed data into the bottom tier from operational databases or other external sources (e.g., customer profile information provided by external consultants). These tools and utilities perform data extraction, cleaning, and transformation (e.g., to merge similar data from different sources into a unified format), as well as load and refresh functions to update the data warehouse. The data are extracted using application program interfaces known as gateways.

A gateway is supported by the underlying DBMS and allows client programs to generate SQL code to be executed at a server. Examples of gateways include ODBC (Open Database Connection) and OLEDB (Object

Linking and Embedding Database) by Microsoft and JDBC (Java Database Connection). This tier also contains a metadata repository, which stores information about the data warehouse and its contents.

• The middle tier is an OLAP server that is typically implemented using either (1) a relational OLAP (ROLAP) model (i.e., an extended relational DBMS that maps operations on multidimensional data to standard relational operations); or [2+6] CO1 L2 (2) a multidimensional OLAP (MOLAP) model (i.e., a special-purpose server that directly implements multidimensional data and operations).

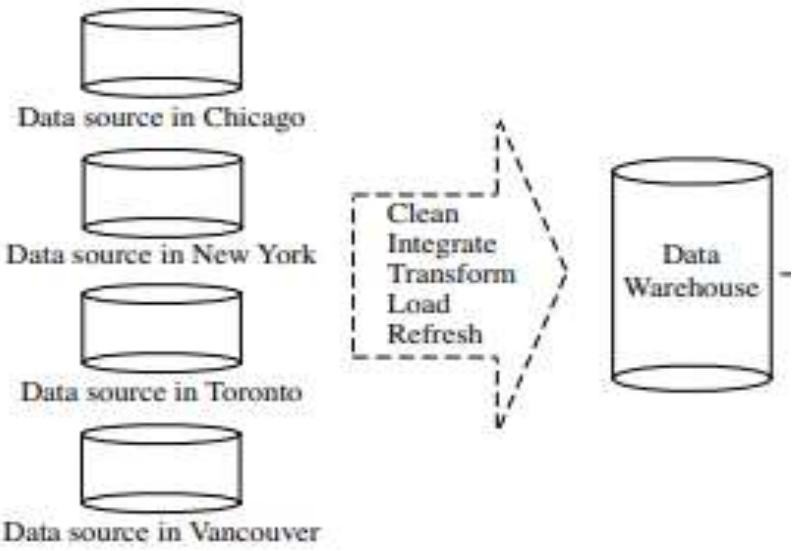
• The top tier is a front-end client layer, which contains query and reporting tools, analysis tools, and/or data mining tools (e.g., trend analysis, prediction, and so on).

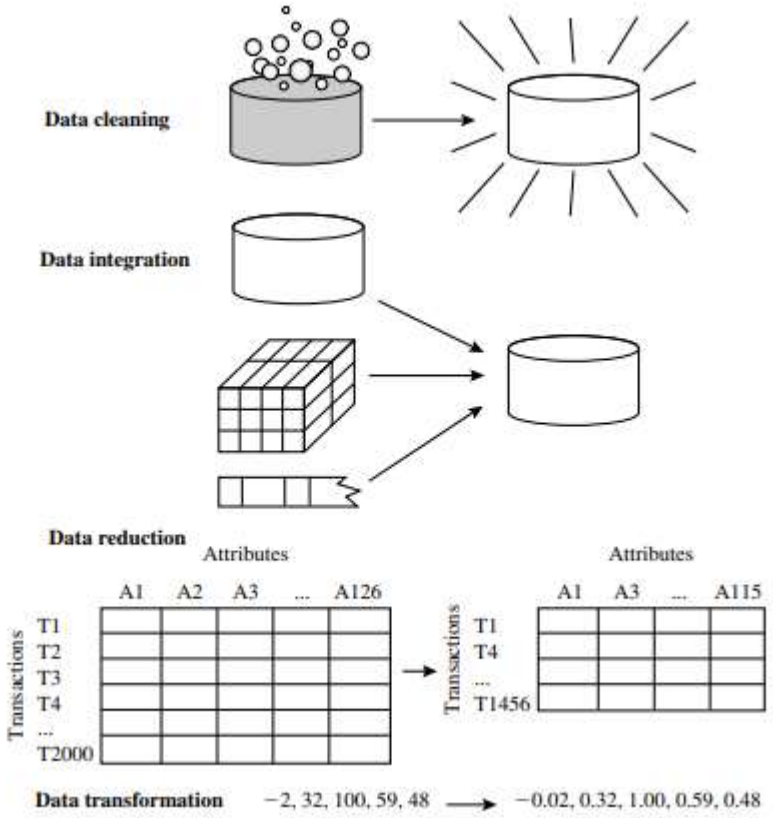
1.b.

Extraction, Transformation, and Loading

Data warehouse systems use back-end tools and utilities to populate and refresh their data. These tools and utilities include the following functions:

- Data extraction, which typically gathers data from multiple, heterogeneous, and external sources.
- Data cleaning, which detects errors in the data and rectifies them when possible.
- Data transformation, which converts data from legacy or host format to warehouse format.
- Load, which sorts, summarizes, consolidates, computes views, checks integrity, and builds indices and partitions.
- Refresh, which propagates the updates from the data sources to the warehouse.
- Besides cleaning, loading, refreshing, and metadata definition tools, data warehouse systems usually provide a good set of data warehouse management tools.



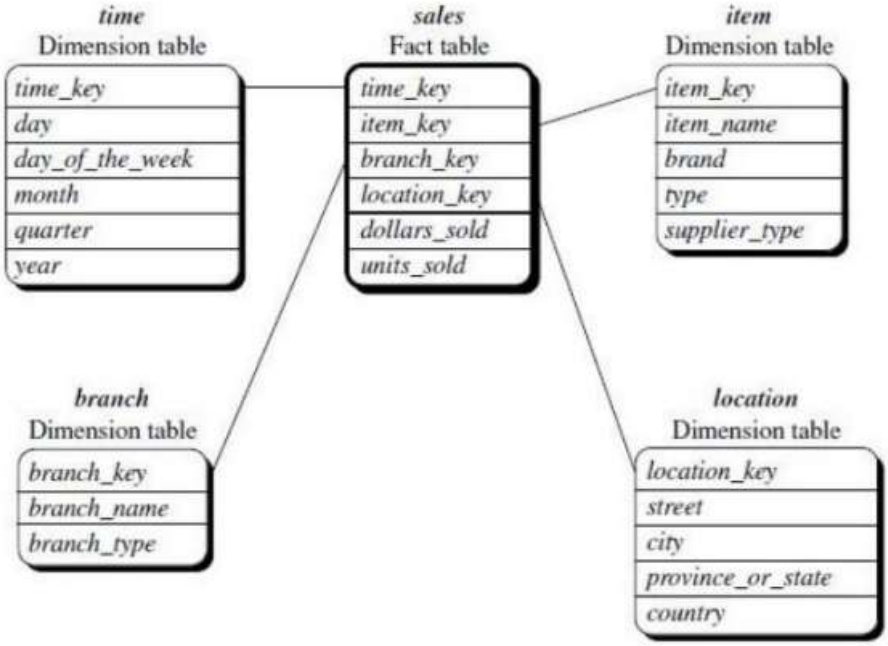


2.a .

Star schema:

• The most common modeling paradigm is the star schema, in which the data warehouse contains (1) a large central table (fact table) containing the bulk of the data, with no redundancy, and (2) a set of smaller attendant tables (dimension tables), one for each dimension.

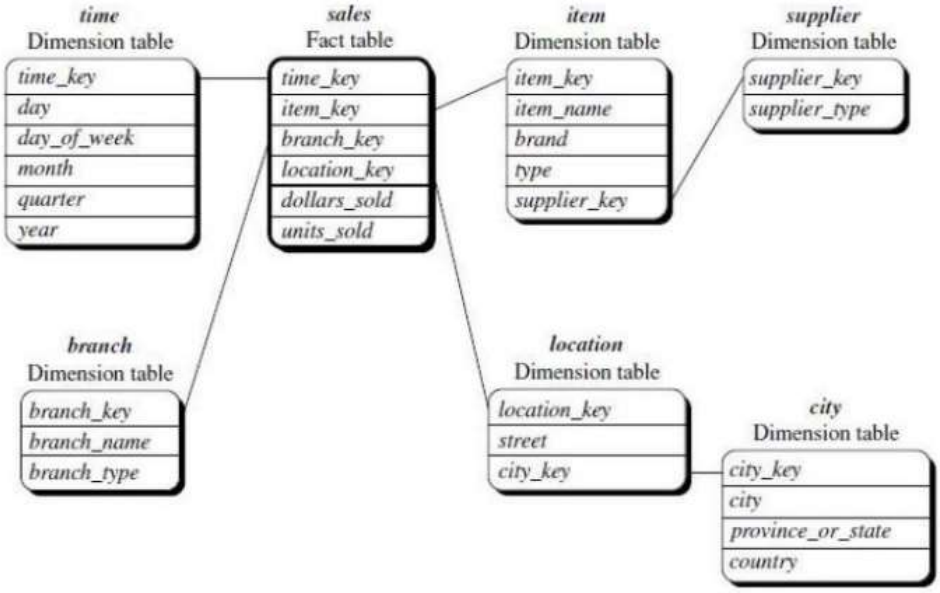
• A star schema for AllElectronics Sales are considered along four dimensions: time, item, branch, and location. The schema contains a central fact table for sales that contains keys to each of the four dimensions, along with two measures: dollars sold and units sold. To minimize the size of the fact table, dimension identifiers (e.g., time key and item key) are system-generated identifiers .



Snowflake schema:

- The snowflake schema is a variant of the star schema model, where some dimension tables are normalized, thereby further splitting the data into additional tables. The resulting schema graph forms a shape similar to a snowflake.

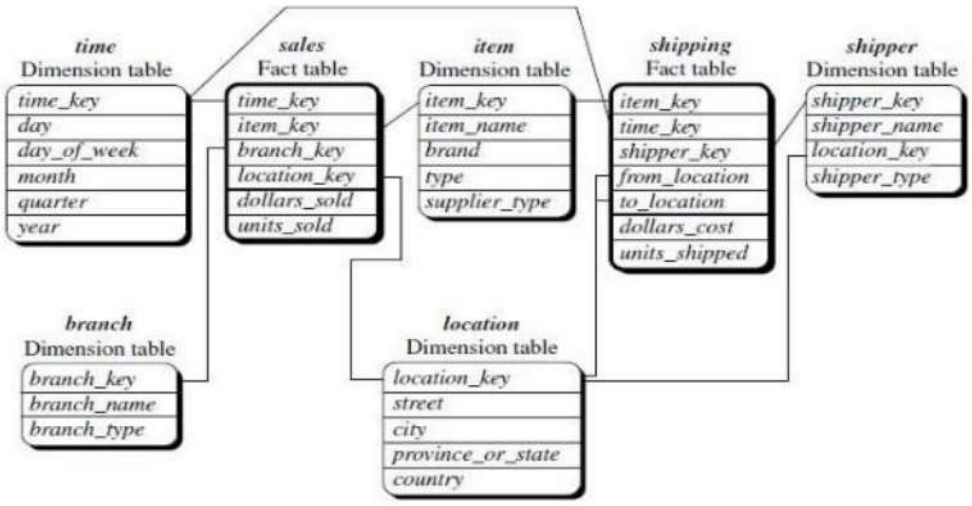
- The sales fact table is identical to that of the star schema. The main difference between the two schemas is in the definition of dimension tables. The single dimension table for item in the star schema is normalized in the snowflake schema, resulting in new item and supplier tables.



Fact constellation:

• Sophisticated applications may require multiple fact tables to share dimension tables. This kind of schema can be viewed as a collection of stars where the dimension tables are shared among the multiple fact tables, and hence is called a galaxy schema or a fact constellation.

• Fact constellation schema specifies two fact tables, sales and shipping. The sales table definition is identical to that of the star schema (Figure 4.6).



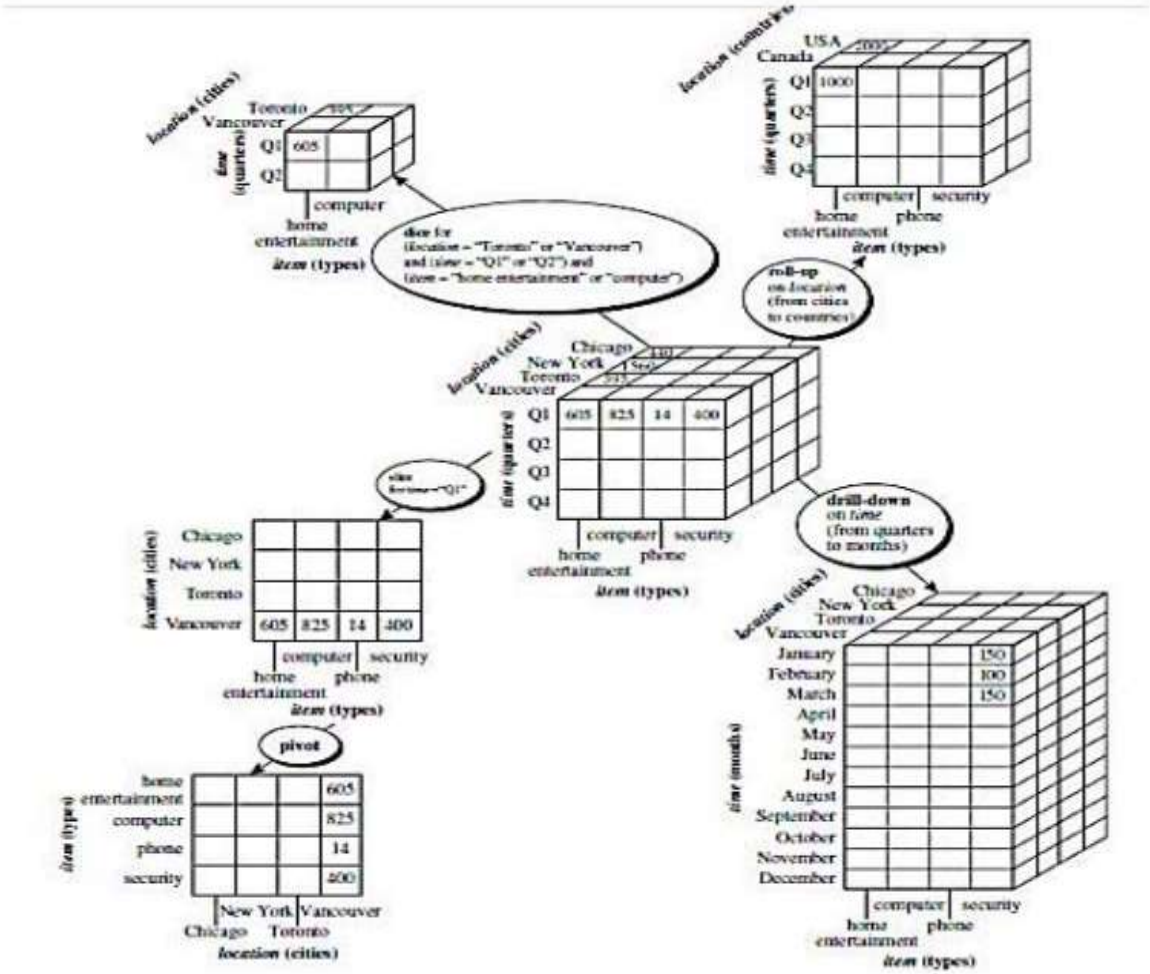
2.b Data Cube Operations:

• The roll-up operation also called as the drill-up operation performs aggregation on a data cube, either by climbing up a concept hierarchy for a dimension or by dimension reduction.

• Drill-down can be realized by either stepping down a concept hierarchy for a dimension or introducing additional dimensions.

• The slice operation performs a selection on one dimension of the given cube, resulting in a subcube and the dice operation defines a subcube by performing a selection on two or more dimensions.

• Pivot (also called rotate) is a visualization operation that rotates the data axes in view to provide an alternative data presentation.



Module 2

Solutions:

3.a .

Two ways of Indexing OLAP data are:

➔ Bitmap indexing

• In the bitmap index for a given attribute, there is a distinct bit vector, Bv, for each value v in the attribute's domain.

• If the attribute has the value v for a given row in the data table, then the bit representing that value is set to 1 in the corresponding row of the bitmap index. All other bits for that row are set to 0.

Base table

RID	item	city
R1	H	V
R2	C	V
R3	P	V
R4	S	V
R5	H	T
R6	C	T
R7	P	T
R8	S	T

item bitmap index table

RID	H	C	P	S
R1	1	0	0	0
R2	0	1	0	0
R3	0	0	1	0
R4	0	0	0	1
R5	1	0	0	0
R6	0	1	0	0
R7	0	0	1	0
R8	0	0	0	1

city bitmap index table

RID	V	T
R1	1	0
R2	1	0
R3	1	0
R4	1	0
R5	0	1
R6	0	1
R7	0	1
R8	0	1

Note: H for "home entertainment," C for "computer," P for "phone," S for "security," V for "Vancouver," T for "Toronto."

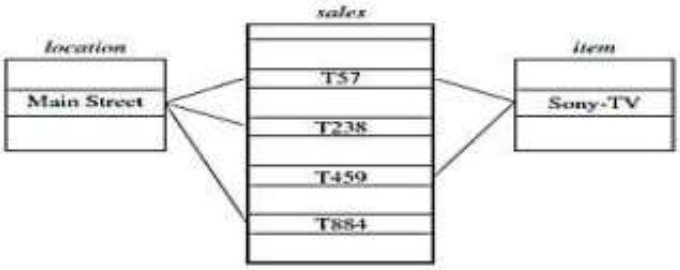
Indexing OLAP data using bitmap indices.

➔ Join Indexing

• The join indexing method gained popularity from its use in relational database query processing.

* Traditional indexing maps the value in a given column to a list of rows having that value.

• In contrast, join indexing registers the joinable rows of two relations from a relational database. For example, if two relations R(RID, A) and S(B, SID) join on the attributes A and B, then the join index record contains the pair (RID, SID), where RID and SID are record identifiers from the R and S relations, respectively.



Linkages between a *sales* fact table and *location* and *item* dimension tables.

Linkages between a *sales* fact table and *location* and *item* dimension tables.

Join index table for *location/sales*

location	sales_key
...	...
Main Street	T57
Main Street	T238
Main Street	T884
...	...

Join index table for *item/sales*

item	sales_key
...	...
Sony-TV	T57
Sony-TV	T459
...	...

Join index table linking *location* and *item* to *sales*

location	item	sales_key
...	...	...
Main Street	Sony-TV	T57
...	...	...

3.b .

(i) Feature Subset Selection

Feature Subset Selection:

- Another way to reduce dimensionality of data
- Use only a subset of the features
- Redundant and irrelevant features can reduce classification accuracy and the quality of the clusters that are found.

Redundant features

- Duplicate much or all of the information contained in one or more other attributes
- Example: purchase price of a product and the amount of sales tax paid almost same

Irrelevant features

- Contain no information that is useful for the data mining task at hand
- Example: students' ID is often irrelevant to the task of predicting students' GPA

Techniques for Feature Selection

1) Embedded approaches: Feature selection occurs naturally as part of DM algorithm. Specifically, during the operation of the DM algorithm, the algorithmitself decideswhich Attributes to use and which to ignore.

2) Filter approaches: Features are selected before the DM algorithm is run.

3) Wrapper approaches: Use DM algorithm as a black box to find best subset of attributes.

Architecture for Feature Subset Selection

The features election process is viewed as consisting of 4 parts:

1) A measure of evaluating a subset,

2) A search strategy that controls the generation of a new subset of features,

3) A stopping criterion

4) A validation procedure.

(ii) Sampling

• Sampling is the main technique employed for data reduction.

• It is often used for both the preliminary investigation of the data and the final data analysis.

• Statisticians often sample because obtaining the entire set of data of interest is too expensive or time consuming.

• Sampling is typically used in data mining because processing the entire set of data of interest is too expensive or time consuming.

➔ Simple Random Sampling

There is an equal probability of selecting any particular object.

There are 2 variations on random sampling:

i) Sampling without Replacement

As each object is selected, it is removed from the population.

ii) Sampling with Replacement

Objects are not removed from the population as they are selected for the sample. The same object can be picked up more than once.

➔ Stratified Sampling

This starts with pre-specified groups of objects. In the simplest version, equal numbers of objects are drawn from each group even though the Groups are of different sizes. In another variation, the number of objects drawn from each group is proportional to the size of that group.

➔ Progressive Sampling

If proper sample-size is difficult to determine then progressive sampling can be used. This method starts with a small sample, and Then increases the sample-size until a sample of sufficient size has been obtained. This method requires a way to evaluate the sample to judge if it is large enough.

4.a

Types of Data sets

1) Record data

→ Transaction (or Market based data)

→ Data matrix

→ Document data or Sparse data matrix

2) Graph data

→ Data with relationship among objects (World Wide Web)

→ Data with objects that are Graphs (Molecular Structures)

3) Ordered data

→ Sequential data (Temporal data)

→ Sequence data

→ Time series data

→ Spatial data

1. Transaction (Market Basket Data):

• Each transaction consists of a set of items.

• Consider a grocery store: The set of products purchased by a customer represents a transaction while the individual products represent items.

• This type of data is called market basket data.

TID	*Items*
1	Bread, Coke, Milk
2	Beer, Bread
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Coke, Diaper, Milk

2. Data Matrix:

• An m*n matrix, where there are m rows, one for each object, & n columns, one for each attribute. This matrix is called a data-matrix.

• If data objects have the same fixed set of numeric attributes, then the data objects can be thought of as points in a multi-dimensional space, where each dimension represents a distinct attribute.

3. Document Data:

• A document can be represented as a “vector”, where each term is a attribute of the vector and Value of each attribute is the no. of times corresponding term occurs in the document.

Projection of x Load	Projection of y Load	Distance	Load	Thickness
10.23	5.27	15.22	27	1.2
12.65	6.25	16.22	22	1.1
13.54	7.23	17.34	23	1.2
14.27	8.43	18.45	25	0.9

(c) Data matrix.

	team	coach	play	ball	score	game	win	lost	timeout	season
Document 1	3	0	5	0	2	6	0	2	0	2
Document 2	0	7	0	2	1	0	0	3	0	0
Document 3	0	1	0	0	1	2	2	0	3	0

(d) Document-term matrix.

GRAPH BASED DATA

Sometimes, a graph can be a convenient and powerful representation for data.

We consider 2 specific cases:

1) Data with Relationships among Objects

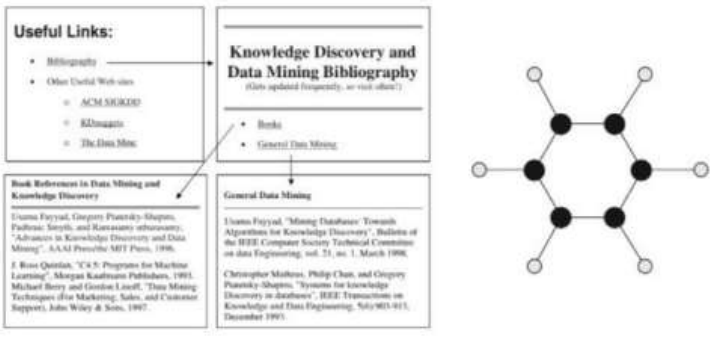
The relationships among objects frequently convey important information. In particular, the data-objects are mapped to nodes of the graph, While relationships among objects are captured by link properties such as direction & weight.

For ex, in web, the links to & from each page provide a great deal of information about therelevance of a web-page to a query, and thus, must also be taken into consideration.

2) Data with Objects that are Graphs

If the objects contain sub-objects that have relationships, then such objects are frequently represented as graphs.

For ex, the structure of chemical compounds can be represented by a graph, where nodes are atoms.



(a) Linked Web pages. (b) Benzene molecule.

Figure 2.3. Different variations of graph data.

ORDERED DATA

a. Sequential Data (Temporal Data)

• This can be thought of as an extension of record-data, where each record has a time associated with it.

• A time can also be associated with each attribute.

• For example, each record could be the purchase history of a customer, with a listing of items purchased at different times.

b. Sequence Data

• This consists of a data-set that is a sequence of individual entities, such as a sequence of words or letters.

• For example, the genetic information of plants and animals can be represented in the form of sequences of nucleotides that are known as genes.

c. Time Series Data

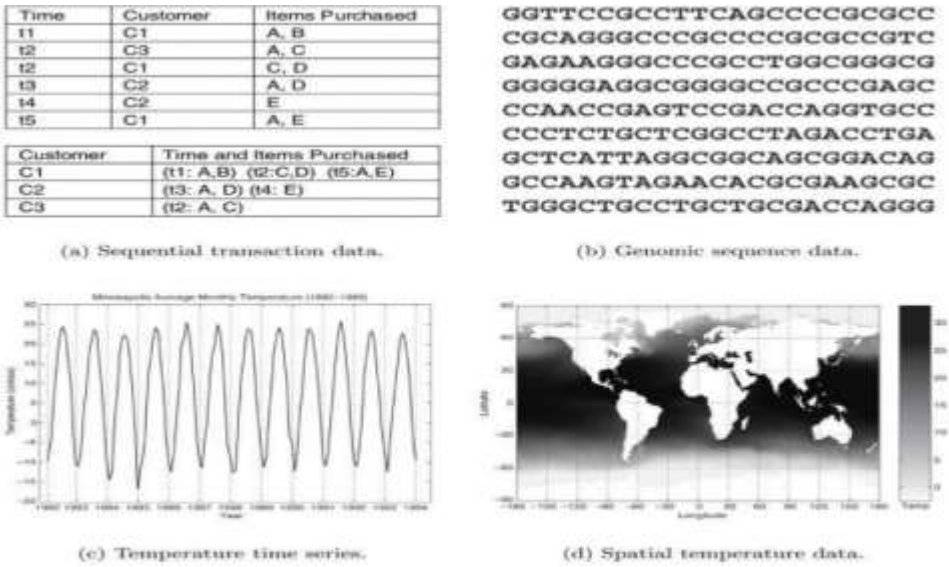
This is a special type of sequential data in which a series of measurements are taken Over time.

• For example, a financial data-set might contain objects that are time series of the daily prices of various stocks.

d. Spatial Data

•Some objects have spatial attributes, such as positions or areas.

•An example is weather-data (temperature, pressure) that is collected for a variety of geographical location.



Time	Customer	Items Purchased
t1	C1	A, B
t2	C3	A, C
t2	C1	C, D
t3	C2	A, D
t4	C2	E
t5	C1	A, E

Customer	Time and Items Purchased
C1	(t1: A,B) (t2:C,D) (t5:A,E)
C2	(t3: A, D) (t4: E)
C3	(t2: A, C)

(a) Sequential transaction data.

```
GGTTCCGCCTTCAGCCCCGCGCC
CGCAGGGCCCGCCCCGCGCCGTC
GAGAAGGGCCCGCCTGGCGGGCG
GGGGGAGGCGGGGCCGCCCGAGC
CCAACCGAGTCCGACCAGGTGCC
CCCTCTGCTCGGCCTAGACCTGA
GCTCATTAGGCGGCAGCGGACAG
GCCAAGTAGAACACGCGAAGCGC
TGGGCTGCCTGCTGCGACCAGGG
```

(b) Genomic sequence data.

(c) Temperature time series.

(d) Spatial temperature data.

Figure 2.4. Different variations of ordered data.

4.b .

https://csucidatamining.weebly.com/assign-3.html

a. x = (0, 1, 0, 1), y = (1, 0, 1, 0) cosine, correlation, Euclidean, Jaccard

cos(x, y) = 0, corr(x, y) = −1, Euclidean(x, y) = 2, Jaccard(x, y)=0

Module 3

Solutions:

5.a .

https://youtu.be/Hm_V2vt7yjk

https://www.ques10.com/p/43610/a-database-has-five-transactions-let-min-support-1/

5.b.

1. General-to-Specific versus Specific-to-General:

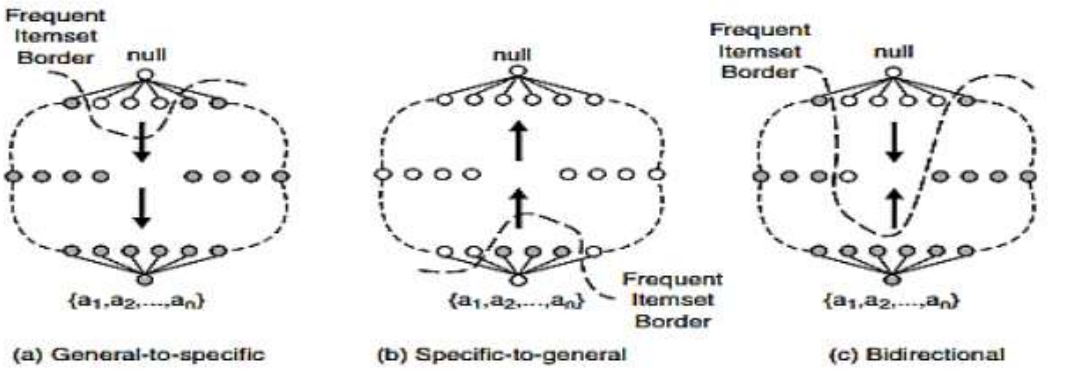
• The Apriori, algorithm uses a general-to-specific search strategy, where pairs of frequent (k- l)-itemsets are merged to obtain candidate k-itemsets.

• This general to-specific search strategy is effective, provided the maximum length of a frequent itemset is not too long.

- Alternatively, a specific to-general search strategy looks for more specific frequent itemsets first, before finding the more general frequent itemsets.

• This strategy is useful to discover maximal frequent itemsets in dense transactions.

• Another approach is to combine both general-to-specific and specific-to general search strategies. • This bidirectional approach requires more space to store the candidate itemsets, but it can help to rapidly identify the frequent itemset border

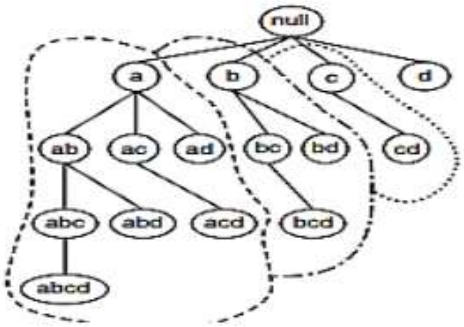


2)Equivalence Classes

• Another way to envision the traversal is to first partition the lattice into disjoint groups of nodes (or equivalence classes).

• A frequent itemset generation algorithm searches for frequent itemsets within a particular equivalence class first before moving to another equivalence class.

Equivalence classes can also be defined according to the prefix or suffix labels of an itemset.



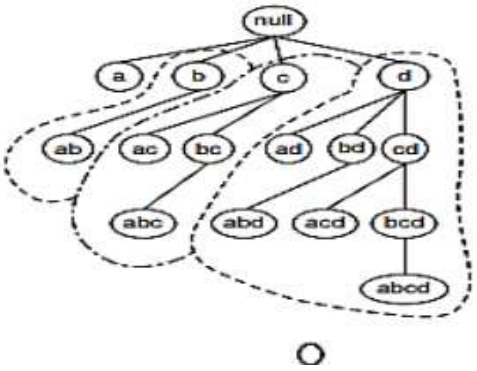


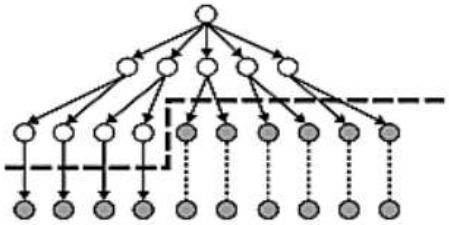
Fig. Prefix tree

Fig. suffix tree

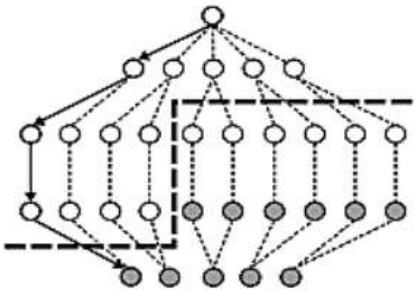
3) Breadth-First versus Depth-First:

• The Apriori, algorithm traverses the lattice in a breadth-first manner .

• It first discovers all the frequent 1-itemsets, followed by the frequent 2- itemsets, and so on, until no new frequent itemsets are generated.



(a) Breadth first



(b) Depth first

• The itemset lattice can also be traversed in a depth-first manner

• If so, the algorithm progressively expands the next level of nodes, i.e., ab, abc, and so on, until an infrequent node is reached, say, abcd.

• It then backtracks to another branch, say, abce, and continues the search from there.

• The deprth-first approach is often used by algorithms designed to find maximal frequent itemsets. • This approach allows the frequent itemset border to be detected more quickly than using a breadth-first approach.

6.a. https://youtu.be/kK6yRznGTdo

6.b.

EVALUATION OF ASSOCIATION PATTERNS :

The Association Analysis Algorithms have the potential to generate a large no of patterns, we could easily end up with thousands or even millions of patterns, many of which might not be interesting.

Scanned with CamScanner

→ "Objective interestingness measure" that uses statistics derived from data to determine whether a pattern is interesting indeed.

→ Different objective measures define different association patterns with different properties & applications.

→ The different types of Association patterns evaluation criteria are:

① Support - confidence Framework
② Interest Factor
③ Correlation Analysis
④ IS Measure

① Support - Confidence Framework :-

→ It is a data - driven approach for evaluating the quality of association patterns.

→ It is domain - independent & requires minimal input from the users, other than to specify a threshold for filtering low-quality patterns.

→ An objective measure is usually computed based on the frequency counts tabulated in a contingency table.

→ The contingency table is shown below:

	B	$\bar{B}$	
A	f_{11}	f_{10}	f_{1+}
$\bar{A}$	f_{01}	f_{00}	f_{0+}
	f_{+1}	f_{+0}	N

Limitations of Support - confidence Framework :

Existing association rule mining formulation relies on the support & confidence measures to eliminate uninteresting patterns.

→ The drawback of support was elimination of potentially interesting patterns & the drawback of confidence is more stable.

② Interest Factor :-

→ The High-confidence rules can sometimes be misleading because the confidence measure ignores the support of itemset appearing in the rule consequent.

Scanned with CamScanner

→ One way to address this problem is by applying metric known as "LIFT":

$$\text{Lift} = \frac{c(A \rightarrow B)}{s(B)}$$

which computes the ratio b/w rules confidence & the support of the itemset in rule consequent.

Limitations of interest factor :-

An example from the text mining domain, it is reasonable to assume that the association b/w a pair of words depends on the no of documents that contain both words.

③ Correlation Analysis :-

→ Correlation analysis is statistical Based technique for analyzing relationship between a pair of variables.

→ For continous variables, correlation is defined using "pearson's correlation co-efficient", for the Binary variables is given by,

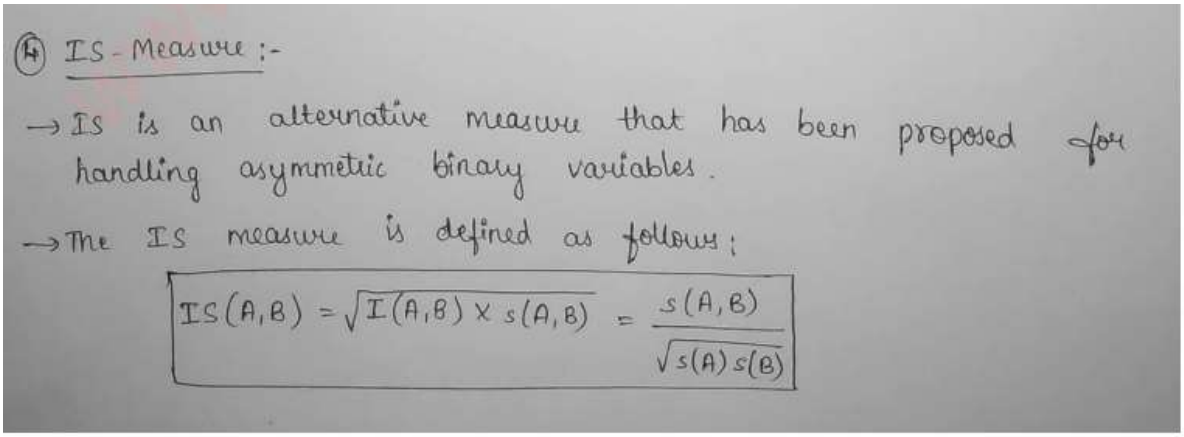
$$\phi = \frac{F_{11}F_{00} - F_{01}F_{10}}{F_{1+}F_{+1}F_{0+}F_{+0}}$$

Limitations of correlation Analysis :-

→ The drawback of using correlation can be seen from the word association.

→ Because the ϕ-coefficient gives equal importance to both co-presence & co-absence of items in a transaction.

④ IS Measure :-



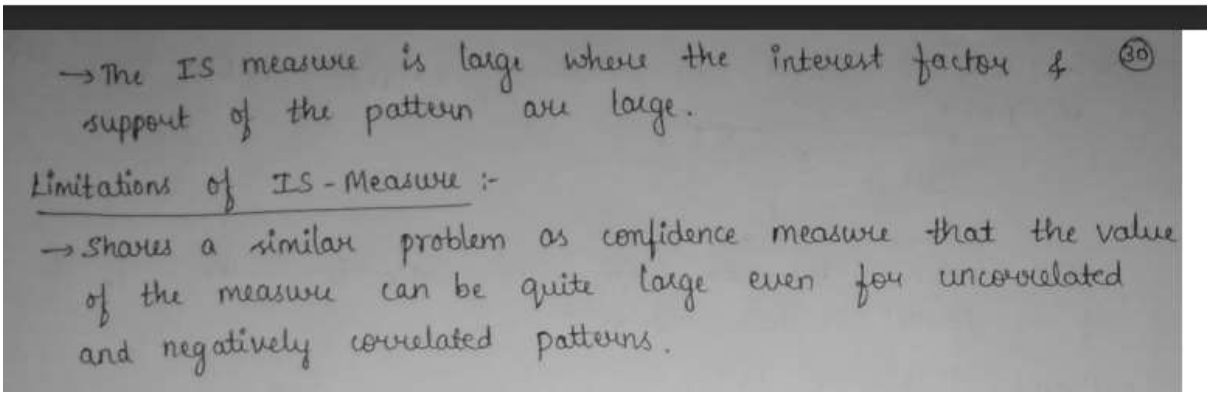
(4) IS - Measure :-

→ IS is an alternative measure that has been proposed for handling asymmetric binary variables.

→ The IS measure is defined as follows;

$$IS(A,B) = \sqrt{I(A,B) \times s(A,B)} = \frac{s(A,B)}{\sqrt{s(A)\,s(B)}}$$

Scanned with CamScanner



→ The IS measure is large where the interest factor & support of the pattern are large. (30)

Limitations of IS - Measure :-

→ Shares a similar problem as confidence measure that the value of the measure can be quite large even for uncorrelated and negatively correlated patterns.

Module 4

Solutions:

7.a.

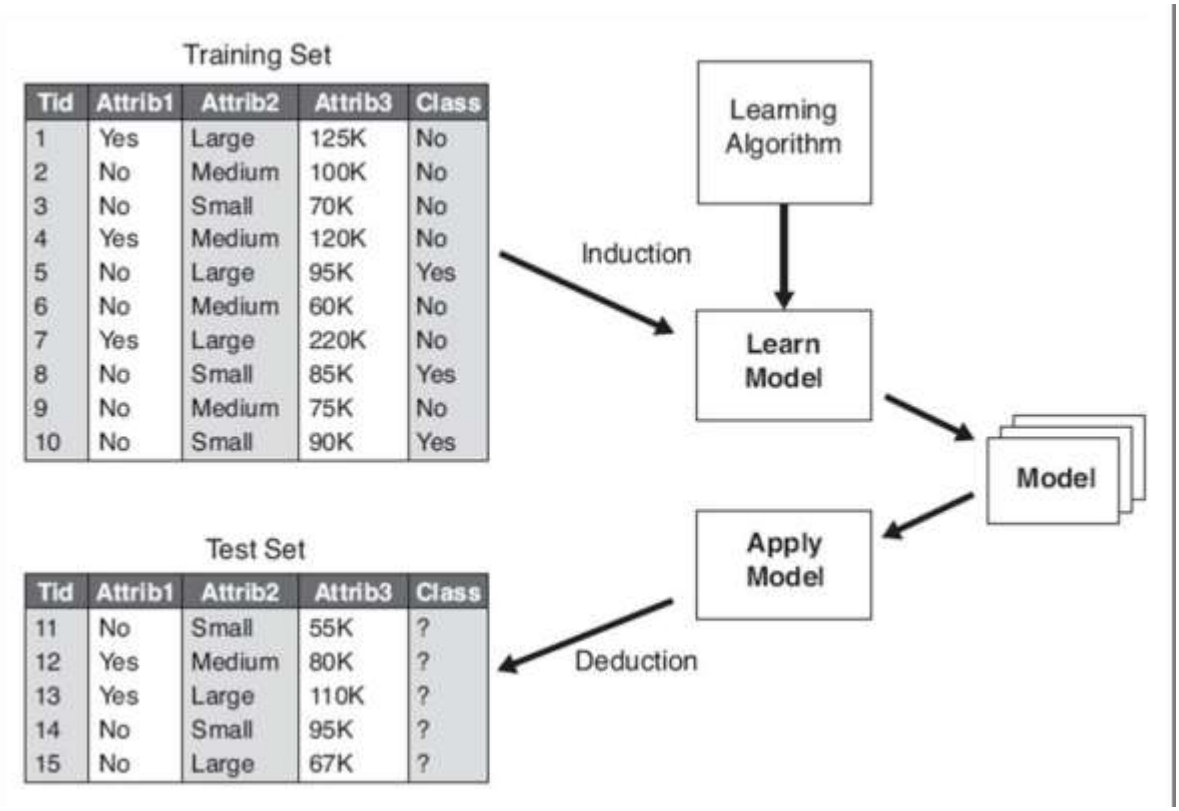
GENERAL APPROACH FOR SOLVING A CLASSIFICATION PROBLEM:

o A classification technique (or classifier) is a systematic approach to building classification models from an input data set.

o The model generated by a learning algorithm should understand the input data well and correctly predict the class labels of records it has never seen before.

o Therefore, a key objective of the learning algorithm is to build models with good generalization capability; i.e., models that accurately predict the class labels of previously unknown records.

o The training set is used to build a classification model, which is subsequently applied to the test set, which consists of records with unknown class labels.



- Evaluation of the performance of a classification model is based on the counts of test records correctly and incorrectly predicted by the model. These counts are tabulated in a table known as a confusion matrix**.**

Table: Confusion matrix for a 2-class problem.

		Predicted Class	
		Class = 1	Class = 0
Actual Class	Class = 1	f_{11}	f_{10}
	Class = 0	f_{01}	f_{00}

o Table depicts the confusion matrix for a binary classification problem. Each entry fij in this table denotes the number of records from class i predicted to be of class j. For instance, f01 is the number of records from class 0 incorrectly predicted as class 1.

o Based on the entries in the confusion matrix, the total number of correct predictions made by the model is (f11 + f00) and the total number of incorrect predictions is (f10 + f01).

Comparison of different models using performance metric:

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions} = \frac{f_{11} + f_{00}}{f_{11} + f_{10} + f_{01} + f_{0}}$$

$$Error\ rate = \frac{Number\ of\ wrong\ predictions}{Total\ number\ of\ predictions} = \frac{f_{10} + f_{01}}{f_{11} + f_{10} + f_{01} + f_{00}}$$

7.b.

Hunt's Algorithm

In Hunt's algorithm, a decision tree is grown in a recursive fashion by partitioning the training records into successively purer subsets. Let D_t be the set of training records that are associated with node t and $y = \{y_1, y_2, \ldots, y_c\}$ be the class labels. The following is a recursive definition of Hunt's algorithm.

Step 1: If all the records in D_t belong to the same class y_t, then t is a leaf node labeled as y_t.

Step 2: If D_t contains records that belong to more than one class, an **attribute test condition** is selected to partition the records into smaller subsets. A child node is created for each outcome of the test condition and the records in D_t are distributed to the children based on the outcomes. The algorithm is then recursively applied to each child node.

The decision tree for the dataset,

	binary	categorical	continuous	class
Tid	Home Owner	Marital Status	Annual Income	Defaulted Borrower
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

is build using the hunt's algorithm in the following way:

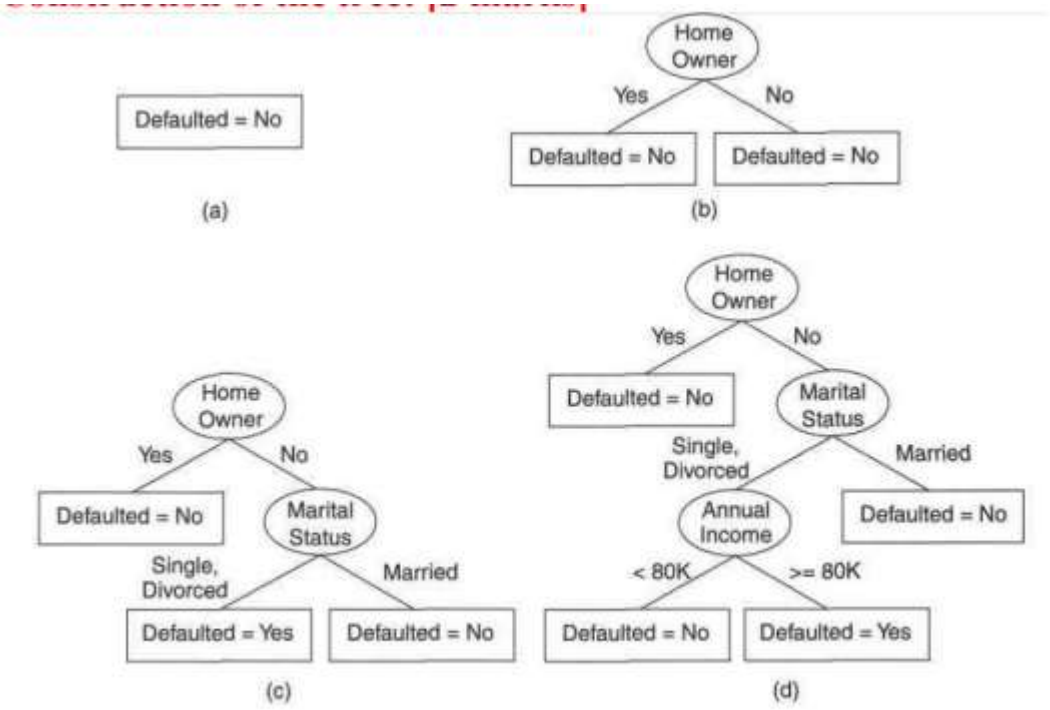


Figure 4.7. Hunt's algorithm for inducing decision trees.

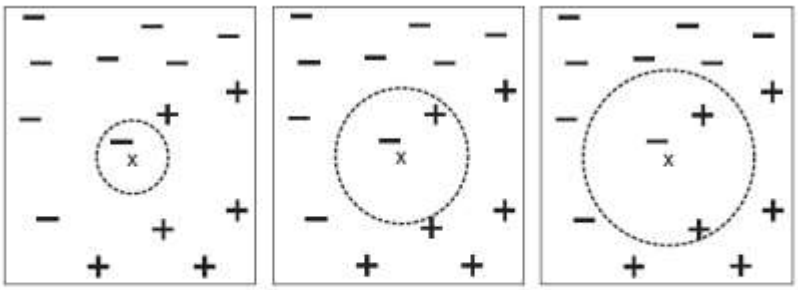
8.a.

• Nearest-neighbor classification is part of a more general technique known as instance-based learning, which uses specific training instances to make predictions without having to maintain an abstraction (or model) derived from data.

• Lazy learners such as nearest-neighbor classifiers do not require model building. • Nearest-neighbor classifiers make their predictions based on local information, whereas decision tree and rule-based classifiers attempt to find a global model that fits the entire input space.

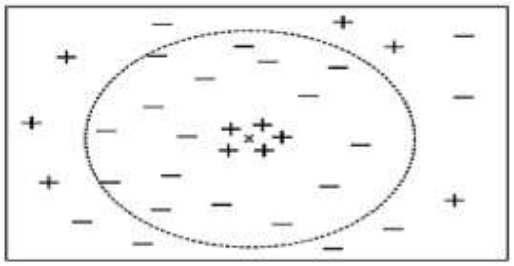
• Nearest-neighbor classifiers can produce arbitrarily shaped decision boundaries.

• Nearest-neighbor classifiers can produce wrong predictions unless the appropriate proximity measure and data preprocessing steps are taken.



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

Figure 5.7. The 1-, 2-, and 3-nearest neighbors of an instance.



k-nearest neighbor classification with large k.

Algorithm 5.2 The k-nearest neighbor classification algorithm.

1: Let k be the number of nearest neighbors and D be the set of training examples.
2: **for** each test example $z = (\mathbf{x}', y')$ **do**
3: Compute $d(\mathbf{x}', \mathbf{x})$, the distance between z and every example, $(\mathbf{x}, y) \in D$.
4: Select $D_z \subseteq D$, the set of k closest training examples to z.
5: $y' = \underset{v}{\text{argmax}} \sum_{(\mathbf{x}_i, y_i) \in D_z} I(v = y_i)$
6: **end for**

8b.

* Bayes Theorem :- It a Statistical principle for combining prior knowledge of classes with new Evidence gathered from data.

→ let X & Y be a pair of Random Variables, Their Joint probability $P(X = x, Y = y)$ refers to the probability that variable X will take on the value x & Variable Y take on value y.

The Joint & conditional probabilities for X & Y are related in following way

$$P(X, Y) = P(Y|X) \times P(X) = P(X|Y) \times P(Y)$$

Rearranging the Expression leads to formula known as "Bayes theorem"

$$P(Y|X) = \frac{P(X|Y)\,(PY)}{P(X)}$$

Nithin VCE

* Using Bayes theorem for classification :- Let X denote the attribute set & Y denote the class variable, If the class variable has non-deterministic relationship with attributes then we can treat X & Y as random variable & capture their relationship using $P(Y|X)$.

This conditional probability is also known as "posterior probability for Y", as opposed to the "prior probability $P(Y)$"

→ during the training phase, we need to learn the posterior probability $P(Y|X)$ for every combination of X & Y based on info gathered in training data.

By knowing these probabilities, a test record X' can be classified by finding class Y' that maximizes posterior probability $P(Y'|X')$.

Scanned by CamScanner

→ Lets consider the following training set

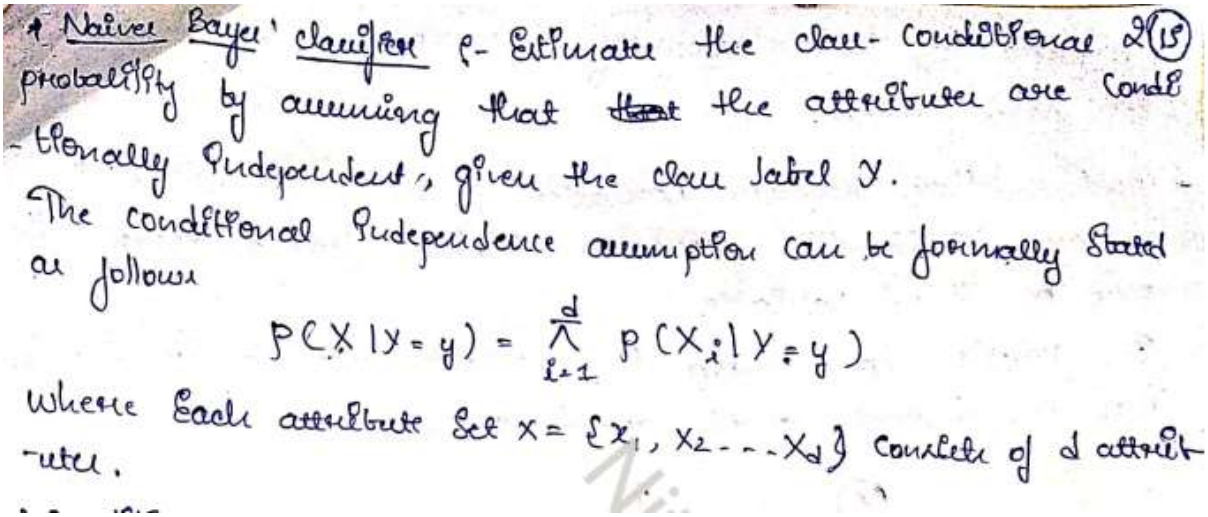
Tid	Home owner	Marital status	Annual Income	Defaulted Borrower
1	Yes	Single	125k	No
2	No	Married	100k	No
3	Yes	Single	85k	No
4	No	divorced	60k	No
5	No	Single	120k	Yes
6	No	Single	100k	No
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
10	No	Married	80k	Yes

Suppose we are given a test record with following attribute set X = (Home owner = No, M.Status = Married, A.Income = 120k)
To classify the record we need to compute posterior probabilities $P(Yes|X)$ & $P(No|X)$ based on the info available in training data.

If $P(Yes|X) > P(No|X)$ then the record is classified as "Yes" otherwise "No"

→ Bayes theorem is useful because it allows us to express posterior probability ($P(Y|X)$) in terms of prior probability $P(Y)$.

MUBIN, VCE



* Naive Bayes' classifier :- Estimate the class-conditional probability by assuming that the attributes are conditionally independent, given the class label y.

The conditional independence assumption can be formally stated as follows:

$$P(X \mid Y = y) = \prod_{i=1}^{d} P(X_i \mid Y = y)$$

Where each attribute set $X = \{X_1, X_2 \ldots X_d\}$ consists of d attributes.

Module 5.

Solutions:

9.a. https://youtu.be/KzJORp8bgqs

9.b.

Density-based

– Grid-based clustering

– Subspace clustering: CLIQUE

– Kernel-based: DENCLUE

Graph-based

– Chameleon

– Jarvis-Patrick

– Shared Nearest Neighbor (SNN)

* Density - Based clustering :- Density Based clustering Algorithm has played a vital role in finding non linear shape structure based on the density.

→ "Density - Based Spatial clustering of Application with Noise (DBSCAN)" is most widely used density Based Algorithm

Scanned by CamScanner

It uses the concept of "density reachability" & "density connectivity".

→ There are three Density - Based clustering Algorithms such as

* Grid - Based clustering
* CLIQUE
* DENCLUE

* Graph-Based Clustering :-

Graph-Based clustering Algorithms use a no of key properties & characteristics of graphs. The foll are some key Approaches used by these Algorithms

* Sparsify the proximity graph to keep only the connections of an object with its nearest neighbours.
* Define a Similarity Measure between two objects based on the no of Nearest Neighbours that they share.
* Define Core objects & build clusters around them.
* use the info in the proximity graph to provide a More Sophisticated Evaluation of whether two clusters should be Merged.

* "Sparsification" → the nxn proximity Matrix for 'n' data points can be represented as a dense graph in which Each

Scanned by CamScanner

node is connected to all others & the weight of the edge between any pair of nodes reflects their pairwise proximity

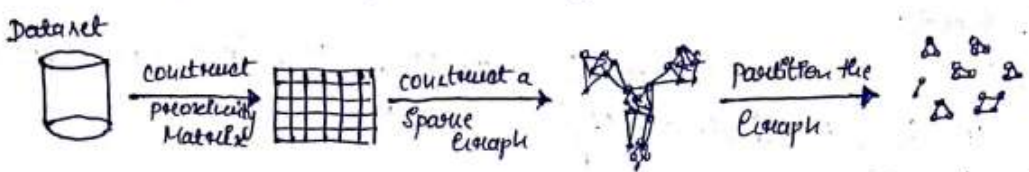


fig: Ideal process of clustering using Sparsification

Sparsification of the proximity graph should be regarded as an initial step before the use of Actual clustering Algorithms.

→ There are five Graph-Based clustering Algorithms such as

* Minimum Spanning Tree (MST) clustering
* OPOSSUM
* Chameleon
* Jarvis-patrick clustering
* SNN Density Based clustering.

10.a.

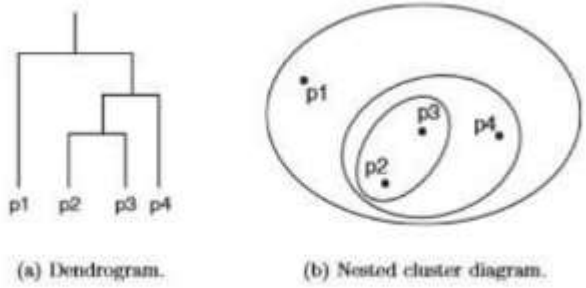
Agglomerative Hierarchical Clustering

Hierarchical clustering techniques are a second important category of clustering methods. There are two basic approaches for generating a hierarchical clustering:

Agglomerative: Start with the points as individual clusters and, at each step, merge the closest pair of clusters. This requires defining a notion of cluster proximity.

Divisive: Start with one, all-inclusive cluster and, at each step, split a cluster until only singleton clusters of individual points remain. In this case, we need to decide which cluster to split at each step and how to do the splitting.

A hierarchical clustering is often displayed graphically using a tree-like diagram called a dendrogram, which displays both the cluster-subcluster relationships and the order in which the clusters were merged (agglomerative view) or split (divisive view). For sets of two-dimensional points, such as those that we will use as examples, a hierarchical clustering can also be graphically represented using a nested cluster diagram.



(a) Dendrogram. (b) Nested cluster diagram.

Figure 8.13. A hierarchical clustering of four points shown as a dendrogram and as nested clusters.

Basic Agglomerative Hierarchical Clustering Algorithm

Many agglomerative hierarchical clustering techniques are variations on a single approach: starting with individual points as clusters, successively merge the two closest clusters until only one cluster remains. This approach is expressed more formally in Algorithm 8.3.

Algorithm 8.3 Basic agglomerative hierarchical clustering algorithm.

1: Compute the proximity matrix, if necessary.
2: **repeat**
3: Merge the closest two clusters.
4: Update the proximity matrix to reflect the proximity between the new cluster and the original clusters.
5: **until** Only one cluster remains.

Defining Proximity between Clusters

the computation of the proximity between two clusters, and it is the definition of cluster proximity that differentiates the various agglomerative hierarchical techniques.

MIN defines cluster proximity as the proximity between the closest two points that are in different clusters, or using graph terms, the shortest edge between two nodes in different subsets of nodes. This yields contiguity-based clusters as shown in Figure 8.2(c).

Alternatively, MAX takes the proximity between the farthest two points in different clusters to be the cluster proximity, or using graph terms, the longest edge between two nodes in different subsets of nodes.

Another graph-based approach, the group average technique, defines cluster proximity to be the average pairwise proximities (average length of edges) of all pairs of points from different clusters

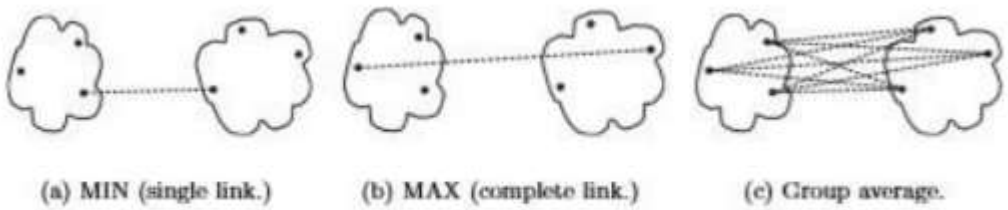


Figure 8.14. Graph-based definitions of cluster proximity

10.b.

DBSCAN

Density-based clustering locates regions of high density that are separated from one another by regions of low density. DBSCAN is a simple and effective density-based clustering algorithm that illustrates a number of important concepts that are important for any density-based clustering approach.

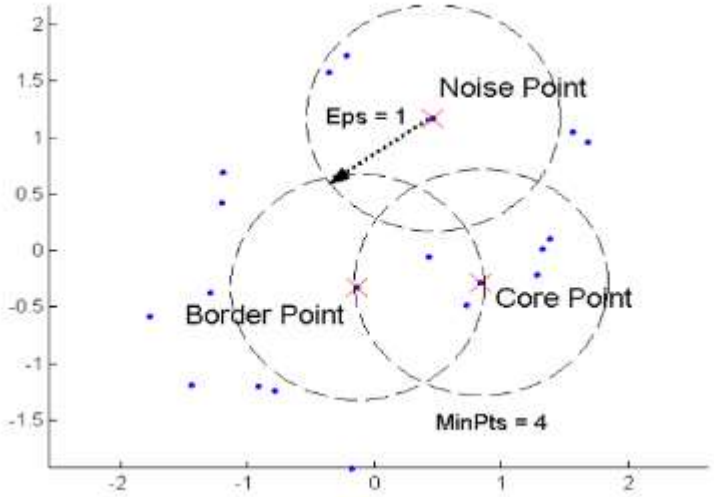
The center-based approach to density allows us to classify a point as

– A point is a core point if it has more than a specified number of points (MinPts) within Eps.

- These are points that are at the interior of a cluster

– A point on the edge of a dense region (a border point), or A border point has fewer than MinPts within Eps, but is in the neighborhood of a core point

– A noise point is any point that is not a core point or a border point.



Any two core points that are close enough within a distance Eps of one another are put in the same cluster. Likewise, any border point that is close enough to a core point is put in the same cluster as the core point. (Ties may need to be resolved if a border point is close to core points from different clusters.) Noise points are discarded.

Algorithm DBSCAN algorithm.

1. Label all points as core, border, or noise points.

2. Eliminate noise points.

3. Put an edge between all core points that are within Eps of each other.

4. Make each group of connected core points into a separate cluster.

5.Assign each border point to one of the clusters of its associated core points.

Time Complexity : $O(n^2)$

- For each point it has to be determined if it is a core point.
- can be reduced to $O(n * \log(n))$ in lower dimensional spaces by using efficient data structures.

Space Complexity : $O(n)$.

- It is only necessary to keep a small amount of data for each point i.e, the cluster label & the identification of each point as a core, border or noise point.

----------<<<ALL THE BEST>>>----------