CSA 16 – Data Warehousing and Data Mining

UNIT I

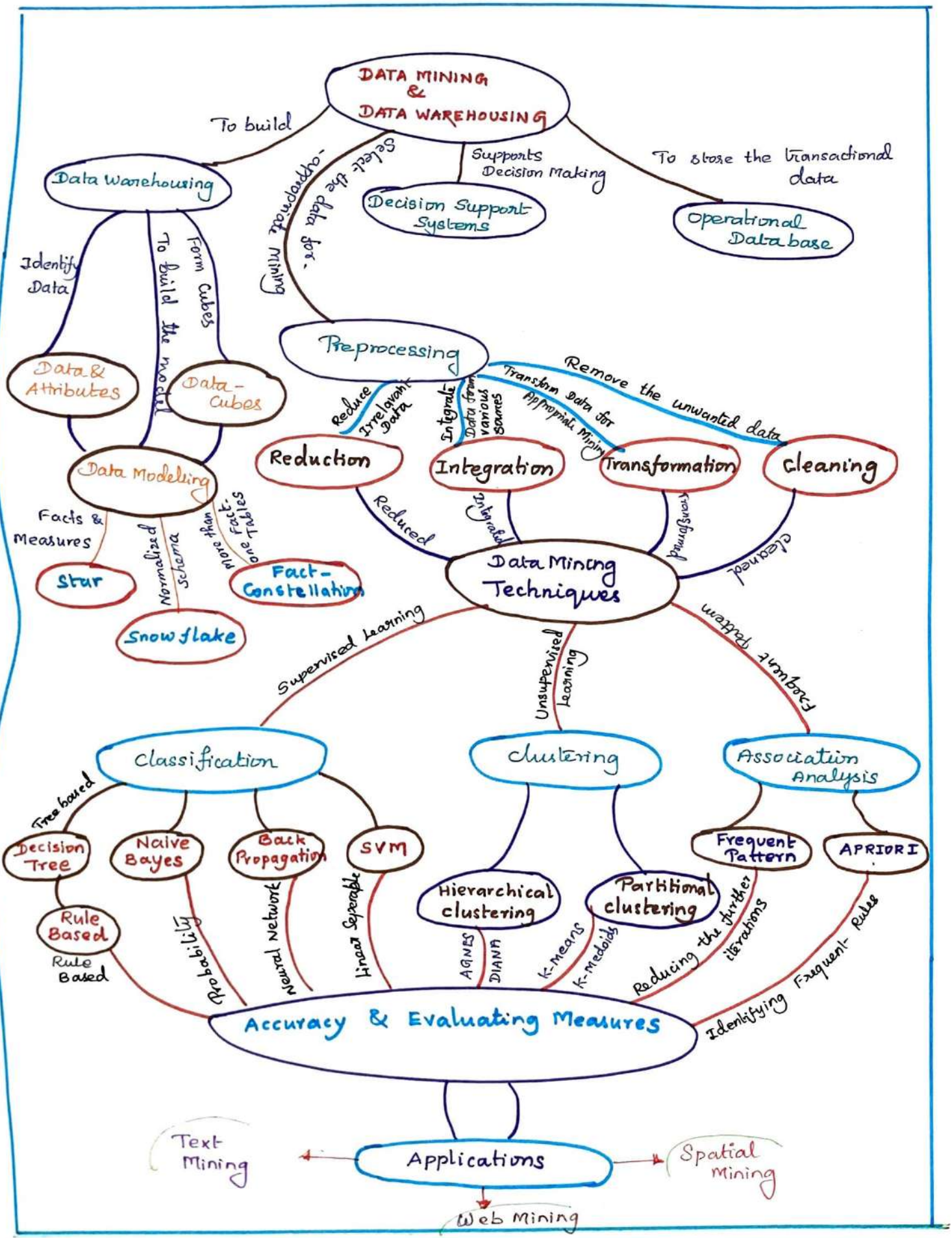
UNIT 2

UNIT 3

UNIT 4

UNIT 5

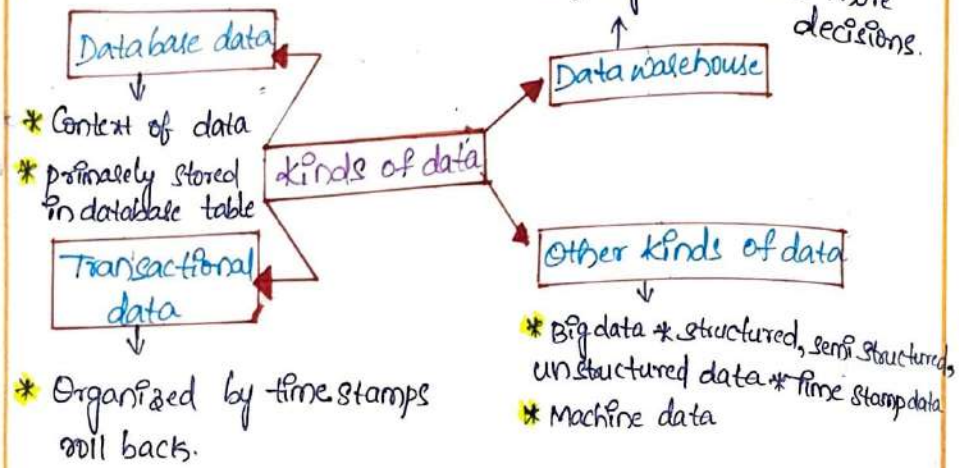
CSA16 – Data Mining & Data Warehousing



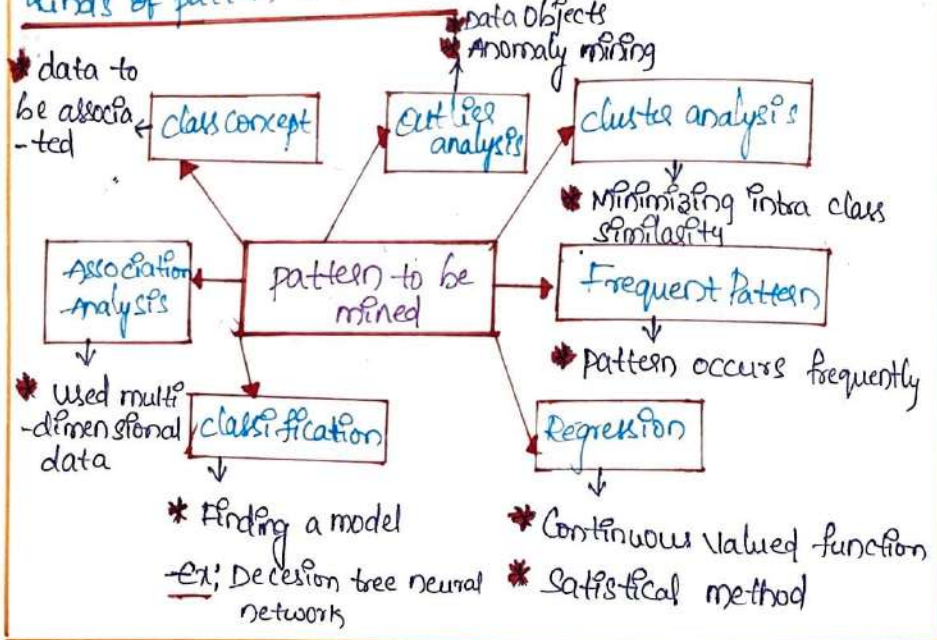
Data Mining :-

- Tools to discover knowledge from data
- Knowledge from data
- Data platform analysis
- Knowledge extraction

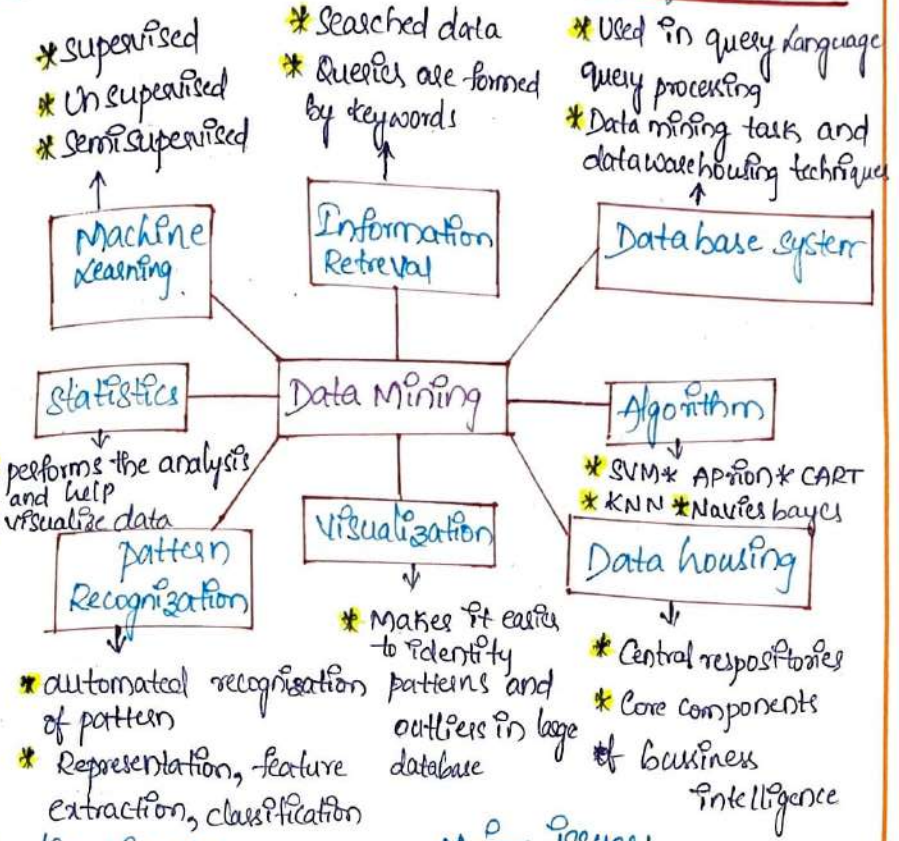
Kinds of data :-



Kinds of pattern to be mined:



Datamining adopts techniques from many domains:



Applications:

Business Intelligence :-

- Historical
- OLAP predictive analysis.

Web Search Engines :-

- Search of information on web
- web directories maintained by human editors.

Major issues:

- Enhance the power and flexibility of datamining
- Handling uncertainty, noise (or) incompleteness of data
- User interaction
- Efficiency and scalability
- Social impact of Datamining
- Privacy of datamining

GETTING TO KNOW YOUR DATA

Data objects and Attribute Types

→ Data objects → represent entity
- Ex:- Sales Data base : Customer, Store, Items
- described attributes
- Data rows → Data objects
- columns → data attribute

Attributes

- Quantitative
  - Discrete
    - finite value
    - Numerical value
  - continuous
    - float type
- Qualitative
  - Nominal
    - name of things
    - Ex: occupation, zipcode
  - Binary
    - Symmetric
      - Both outcome equally important.
      - ex: gender
    - Asymmetric
      - Both outcome not equally important.
      - ex: Medical test
  - ordinal
    - value have meaningful
    - ex: grade

Basic Statistical Descriptions of Data

Motivation - Better understand the data: central tendency, variation and speed.

Data dispersion characteristics - Median, max, min, outliers.

Numerical Dimensions - Data dispersion, Box plot.

Dispersion analysis on computed measures -

* Folding measures into numerical dimension.
* Boxplot on the transformed cube.

Data Visualization

- Mapping data onto graphical primitives.
- provide qualitative overview of large datasets.
- search for patterns.
- provide a visual proof.

Category of Visualization Method

- Pixel oriented
- Geometric progrection
- icon based
- Hierarchical
- visualizing complex data & relation.

Similarity Measures

- real value function that quantifies the similarity between two objects.
- Measure how two data objects are alike
- often falls in the range [0,1] : 0-no similarity
  1 - completly similar

Dissimilarity Measures

- Numerical measure of how different two data objects.
- Minimum dissimilarity
- range [0,1] or [0,∞]

Data Warehousing:

* provides architecture and tools for business executives to systematically organize, understand and use their data to make strategic decisions.
* DW is subject oriented, integrated, time variant and non volatile.

Subject Oriented: Data warehouse is organised around major subjects such as customer, supplier, product and sales.

Integrated: A dw is usually constructed by integrating multiple hetrogenous sources such as flat files, Relational db and online transaction records

Time variant: Historic data/information [5-10 yrs]

Non volatile: Application data found in Operational environment [permanent storage].

Difference blw Datawarehouse & Operation database

Datawarehouse	Operation database
* Datawarehouse is repository for structured, filtered data.	* Database changes frequently.
* Denormalized schema.	* Normalized schema.
* Historical data	* current transaction data.
* Online analytical processing.	* Online transaction processing.

* Decision support System: (DSS)

* It is computerized program used to support determinations, judgements and course of actions in organization.

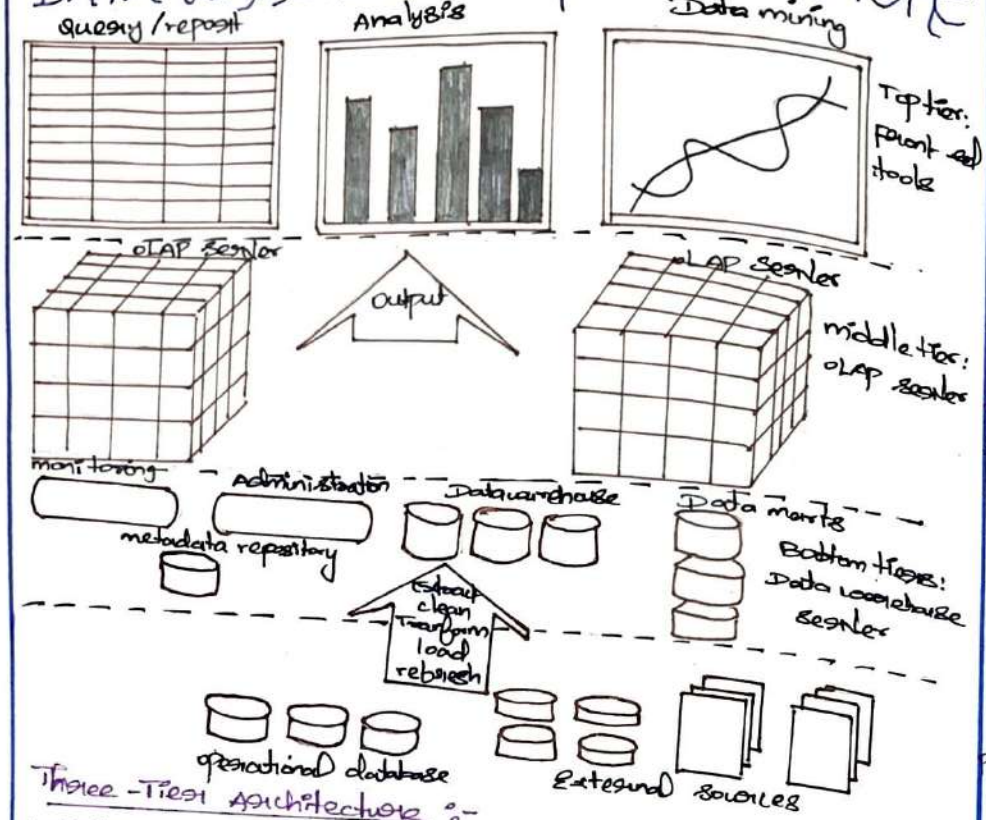
Ex:

DSS: target (or) projected revenue, predict the sales.

It is completely computerized & powered by humans.

characteristic	Operation data	DSS
Data current granularity	Real time data Atomic-detail data.	Historic data summarized data
Data model	RDBMS	non-normalized model.
Transaction volumes	high	Summary
Query complexity	simple to medium	very complex
Data volumes	Hundreds of GB	Tera (or) peta bytes
Query activity	low to medium	High

DATA WAREHOUSING ARCHITECTURE



Three-Tier Architecture :-

1. Bottom-Tier :
   → DB-Servers.
2. Back-end-Tools:
   → Extract info/..., load & clean.
3. Middle-Tier:
   → Extension of RDBMS
4. Top-Tier:
   → client layer [query tools, reporting tools.., mining tools etc.]

Types of Data warehouse :-

- Enterprise DWH → centralised DSS
- operation Data Store → OLTP support
- Data mart → Subset of DWH Eg: Sales, finance.

Extraction, Transformation & loading Extract ETL

During data extraction, raw data is copied structured or unstructured. These sources of data include but are not limited to :

→ SQL or NoSQL servers → CRM & ERP systems → Flat files.
→ Email → webpages.

Transform :

The data is transformed & consolidated for its intended analytical use case.

→ Filtering, cleansing, de-duplicating, validating, & authenticating the data.
→ conducting audits to ensure data quality and compliances
→ Removing, encrypting, or protecting data governed by industry or government regulators.
→ Formatting the data into tables or joined tables to match the schema of the target data warehouse.

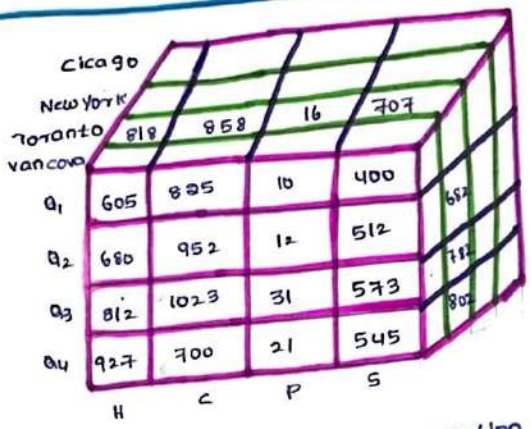
Load :

The transformed data is moved from the staging area into a target data warehouse. Initial loading of all data, followed by periodic loading of incremental data changes & , less often, full refreshes to ensure & replace data in the warehouse.

→ Comparison of OLTP & OLAP systems :

Feature	OLTP	OLAP
• characteristic	operational processing transactions	informational processing analysis
• orientation	clerk, DBA, database professional	knowledge worker (eg. manager, executive, analyst)
• user	day-to day operations	long-term informational requirements decision support
• Function	ER-based, applications-oriented	star/snowflake, sub-oriented
• DB design	current, guaranteed up-to-date	historic, accuracy maintained over time
• Data	primitive, highly detailed	summarized, consolidated
• Summarization	detailed, flat relational	summarized, multidimensional
• view	short, simple transaction	complex query
• unit of work	read/write	mostly read
• Access	data in	info. out
• Focus	index/hash on primary key	lots of scans
• operations	tens	millions
• no. of records accessed	thousands	hundreds
• no. of users	GB to high-order GB	→ TB
• DB size	high performance, high availability	high flexible, end-user autonomy
• priority	transaction throughput	query throughput, response time
• metric		

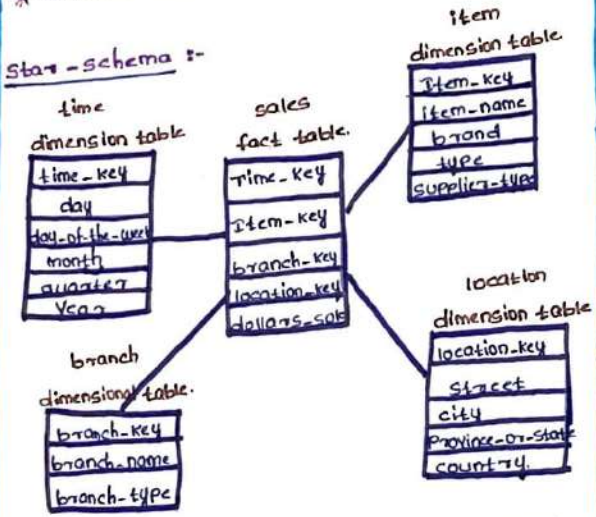
Dimentional Data Modelling



Star, snowflake and fact constellation :-

* schemas for multidimensional data model

Star-schema :-



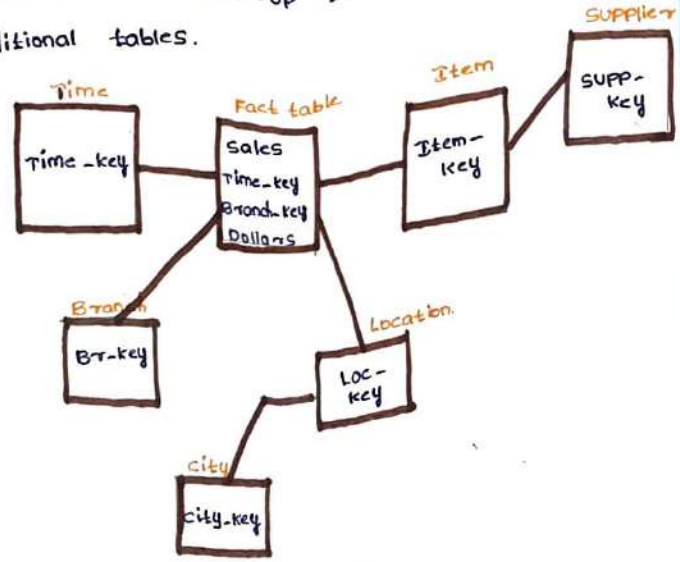
→ Bulk of data with no-redundancy.

→ A set of smaller attendant table [1-for each dimension]

* It contains both dimensional table and fact table.

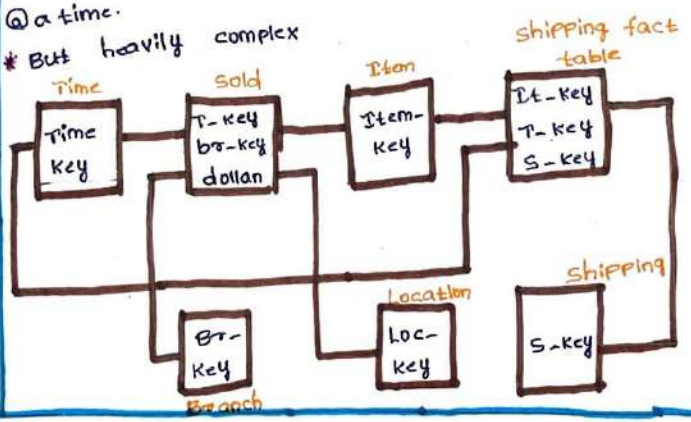
Snow-Flake Schema :- [Top-down model type]

* some of them are normalized
* Normalization split-up the data into additional tables.



Fact constellation schema :-

* It consists of dim-table that are shared by several fact tables
* It consists of more than one star-schema @ a time.
* But heavily complex



→ Also known as "Galaxy schema"

Star schema definition :-

define cube sales_star [time, item, branch, location] :
dollars_sold = sum (sales_in_dollars), units_sold = count (*)
define dimension time as (time_key, day, day_of_week, month, quarter, year)
define dimension item as (item_key, it_name, brand, type, supplier_type)
define dimension branch as (bran_key, bran_name, bran_type)
define dimension location as (loc_key, street, city, province_or_state, country)

Snowflake schema definition :-

define cube sales_snowflake [time, item, branch, location] :
dollars_sold = sum (sales_in_dollars), unit_sold = count (*)
define dimension time as (time_key, day, day_of_week, month, year)
define dimension item as (it_key, it_name, br_name, br_type, (supplier_key, supplier_type))
define dimension branch as (br_key, br_name, br_type)
define dimension location as (loc_key, street, city (city_key, city, province_or_state))

Fact constellation schema definition :-

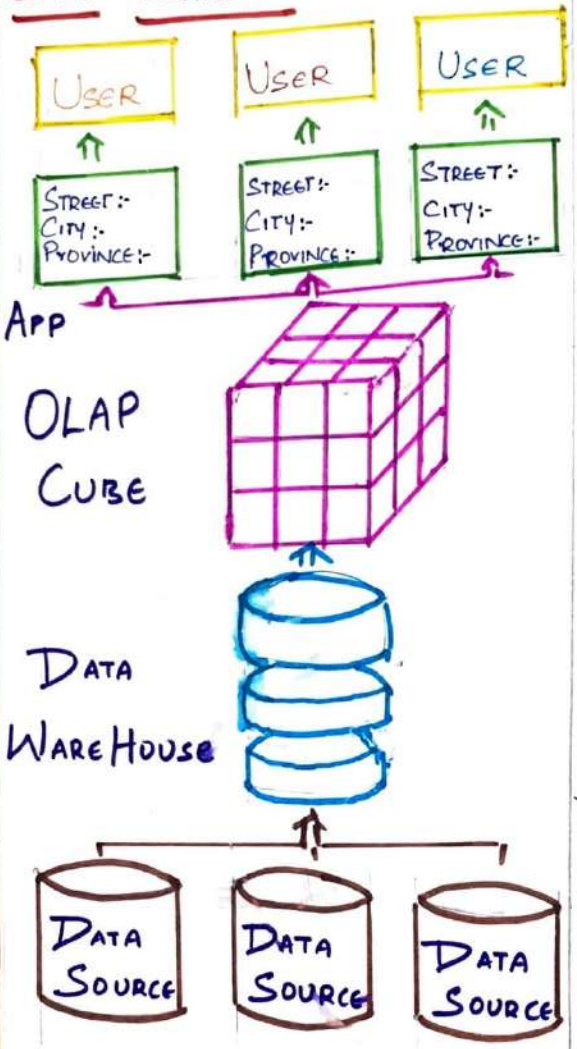
define cube sales [item, time, branch, location] :
dollars_sold = sum (sales_in_dollars), units_sold = count (*)
define dimension time as (time_key, day, day_of_week, month)
define dimension item as (item_key, item_name, brand, type)
define dimension branch as (branch_key, branch_name, br_type)
define dimension location as (loc_key, street, city, country)
define cube shipping [time, item, shipper, from_location) :
dollars_cost = sum (cost_in_dollars), = count (*)
define dimension from_location as location in cube sales
define dimension to_location as location in cube sales.

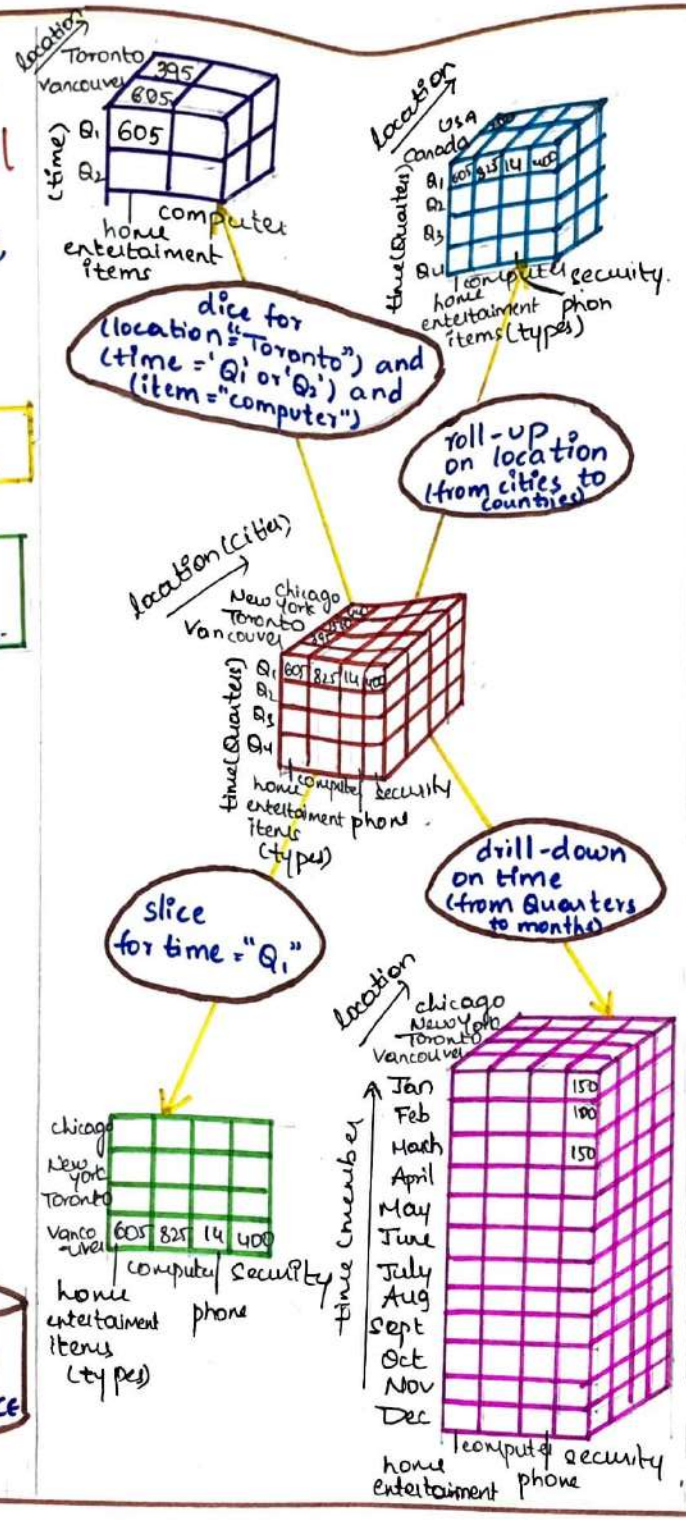
Online Analytical Processing

OLAP - Multidimensional

Information - Manage, Analysis, Interactive

OLAP Preparation





OLAP - Operations

Roll Up : Aggregation on data

ways :- (1) climbing up on a concept hierarchy for a dimension (location)

(2) Dimension Reduction

work concept :- Initial hierarchy "street < city < province < country"

Aggregated hierarchy location from "city" to "country"

Roll-Down : stepping down a hierarchy

* New dimension

Work :- hierarchy for dimension time "day < Month < Quarter < year"

Slice :- Create a new sub-cube from one-particular cube

way :- Dimension = "time", time = "Q_1"

Dice :- Three dimension

Location = "Toronto" or "Vancouver"

Time = "Q_1" or "Q_2"

Item = "Mobile" or "Tv"

Pivot :- PIVOT - Rotation Operation

TOTAL NO OF CUBOIDS :-

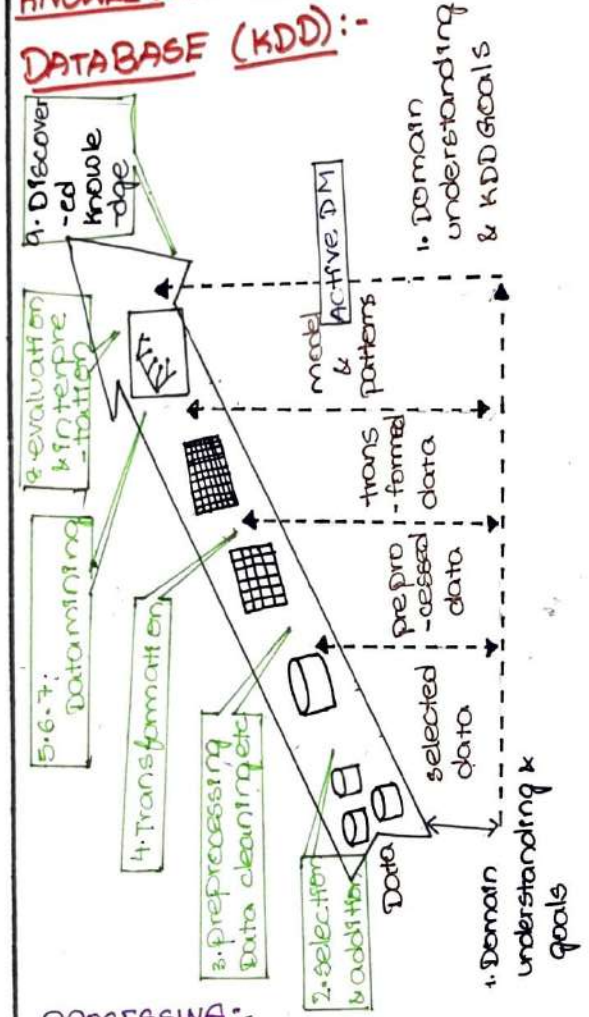
$$\prod_{i=1}^{n} (L_i + 1)$$

L_i = Levels associated with dimension

1 = Virtual top level

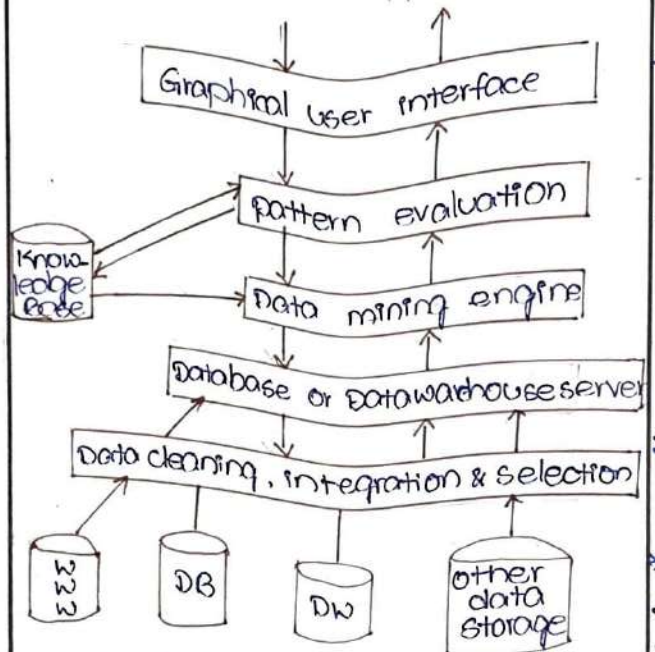
DATA CUBE COMPUTATION

KNOWLEDGE DISCOVERY IN DATABASE (KDD):-



PROCESSING:-

1. Data cleaning
2. Data integral
3. Data reduction
4. Data transfer
5. Data discreatization
6. Data mining
7. pattern evaluation
8. Representing knowledge.



DATA SOURCE:-

Database, www, text files and other documents.

organization store data in datawarehouse different process:-

* Data must be cleaned integrated & selected.
* Data of interest to be select -ed & passed to the server.

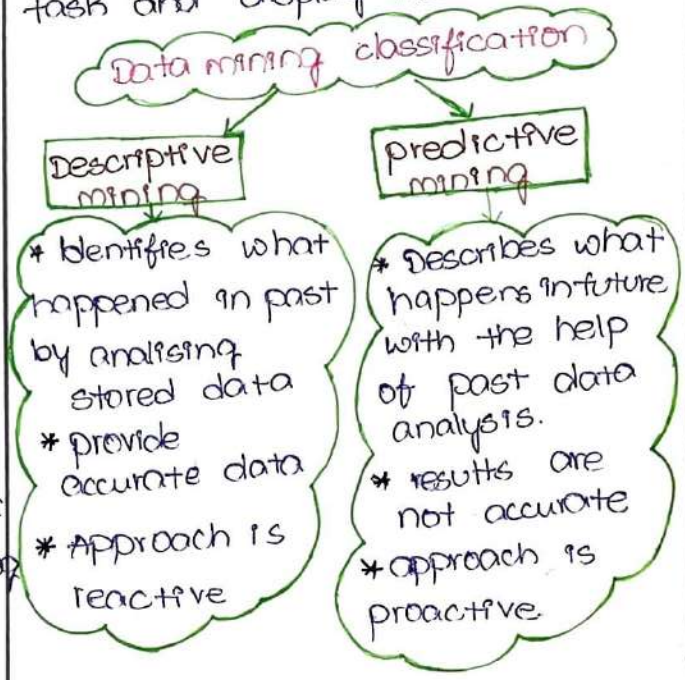
Data mining engine: contains modules like association, charac -terization, classification, clustering prediction, time-series analysis

pattern evaluation engine:-

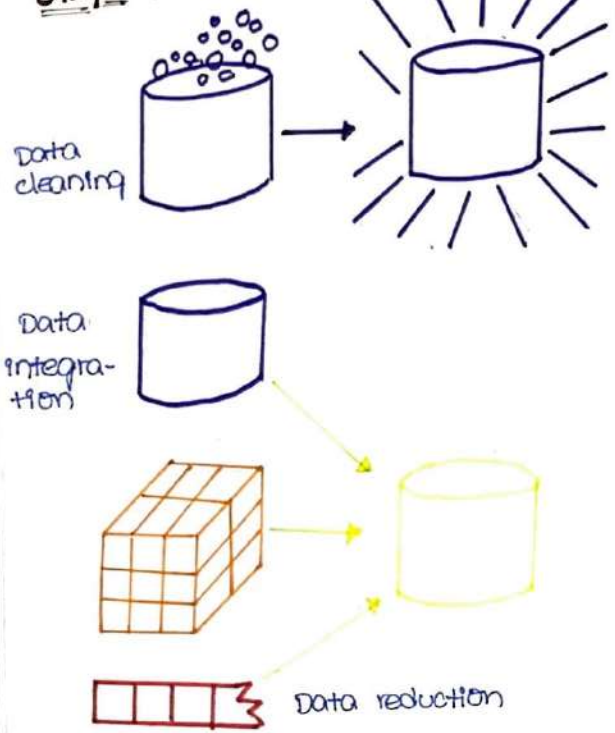
* Measure of investigation of the pattern using a threshold value.
* collaborates with data minings engine to focus the search on exciting patterns.

Graphical user interface:-

* communicates between the data mining system and the user.
* this module cooperates with the data mining system.
* user specifier a query or a task and display the results.



STEPS IN DATA PREPROCESSING:



Preprocessing:-

* Transforming raw data into understandable format.
* premilanary step for DM

* check data quality

1. **Data cleaning:-**
* remove in correct data.

2. **Data integration:-**
* combining multiple sources into a single data set.

3. **Data reduction:**
* Reduction of data volume.

4. **Data transformation** - Mining task
* convert data into required format.

DATA CLEANING:-

* Data - real world is dirty.
* incomplete - lacking attribute values eg: age=" "

NOISY:
* contain noise or error salary= "10"

INCONSISTENT:- contain discrepances, age=ba

How to handle missing data?

Ignore the tuple:
* fill the missing value manually.
* fill automatically: global constant (mean value).

How to handle noisy data?

Binning:-

1. sort data & partition into (equal frequency) bin.
2. smooth bin means, median by boundaries

Regression:-
* smooth by fitting data into regression

clustering: detect & remove outliers

DATA CLEANING AS A PROCESS:-

* DATA DISCREPENCY DETECTION.
* use metadata
* check uniqueness rule, consecutive rule
* use commercial tools:-
  Data scrubbing or data auditing.

DATA MIGRATION & INTEGRATION:
* Allows transformation to be specified.

ETL:- allows users to specify transformations through gui.

DATA INTEGRATION:- combines data from multiple sources into a coherent store.

Handling redundancy in data integration:
* redudant data occur often when integration of multiple database.
* object identification
* Derivable data.
* Redudant attributes detect using correlation analysis and covariance analysis.

Correlation analysis:

χ^2 (chi-square) test.

$$\chi^2 = \sum \frac{(\text{observed} - \text{expected})^2}{\text{expected}}$$

large χ^2 value, more lickily the variable are related.

covariance: $Cov(A,B) = E((A-\bar{A})(B-\bar{B})) = \frac{\sum_{i=1}^{n}(a_i-\bar{A})(b_i-\bar{B})}{n}$

$V_{A,B} = Cov(A,B) / \sigma A \sigma B$.

positive covariance:- if $Cov_{A,B} > 0$.

-ve covariance: if $Cov_{A,B} < 0$ A is larger than its expected value, B is small than expected value.

Independence : $Cov\ A,B = 0$

Data Reduction :- Reduced representation of the data set that smaller in — volume.

* Produces the same analytical result.

Data Reduction Strategies

Dimensialty reduction

Wavelet Transform :- To preserve relative distance between objects at different levels of resolution.

Principal Component analysis :- (PCA)

* Original data are projected onto a much smaller space, resulting in dimensionality reduction.
* Find eigen vectors of the coaminance matrix, these eigen vector defines space.

Numerosity reduction :- Regression and Log-linear

Linear :- Data modelled to fit a straight line.

Multiple regression :- Allow response to variable x to be modeled as linear function.

Histogram analysis :- Divide data into buckets and store average (sum) of each bucket perfection rule. Equal bucket range and Equal frequency.

Data Transformation :- Which maps the entire set of values of attitude to new set of values.

Smoothing :- Remove noise from data attitude. New attribute — constructed from the given ones.

Aggregation :- Summarization, data cube construction.

Nonmalization :- Fall within a smaller.

min-max nonmalization :-

$$V' = \frac{V - Vmin_A}{max_A - Min_A} (New\text{-}max_A - new\text{-}min_A) + New\text{-}min_A$$

Z Coore - Homalization :-

$$V' = V - \mu_A / \sigma_A$$

μ = Mean σ = Standard deviation

Nomalization by decimal scaling :-

$V' = V/10^j$ ∵ j = Smallest integer such that $max(|V'|) < 1$

Discretization :- Divide the range of continuous attribute into intervals.

* Interval lables can be used to replace data values.
* Reduce data size by discretization method.

Bining :- Top down split

Histogram analysis :- Top down split.

Equal-width (distance) Partitionary :-

⇒ Divide the range into N intervals of equal size.

⇒ A & B are the lowest & Highest values of attribute wide $W = (B-A)/N$.

Equal-depth (Frequency) Partitionary :-

⇒ Divide the range into N intervals, each containing approximately same number of samples.

⇒ **Good data scaling :-**

concept Hierarchy organizes, concepts hierarchically and is usually associated with each dimension in a data ware house.

CLASSIFICATION

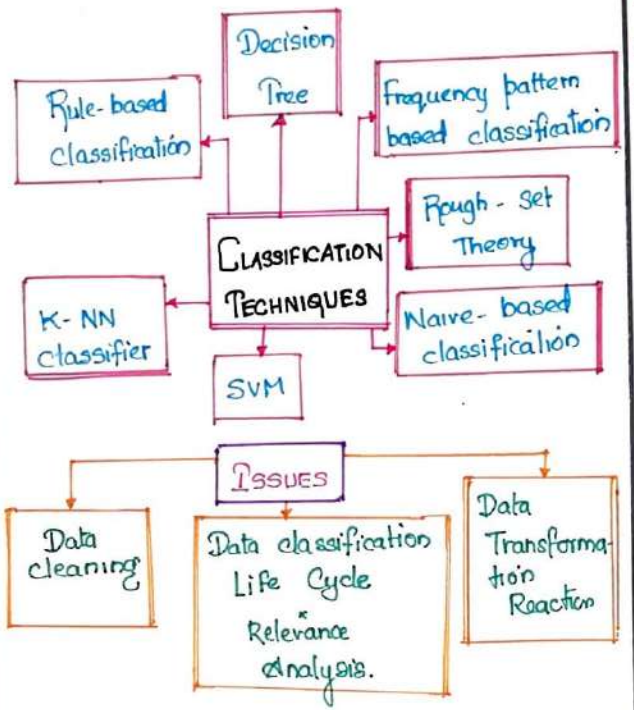
To predict the Categorial Labels
- Continuous- valued function
- Ordered value
- Regression Analysis

Building the Classifier / Model

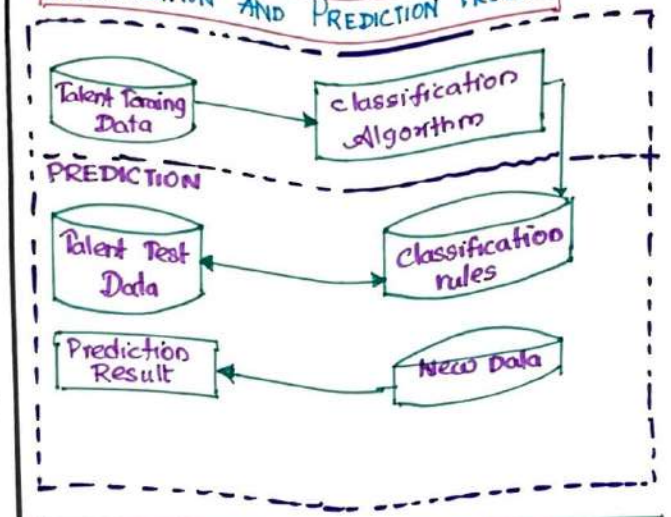
Learning Phase → Classification Algorithm → Classification Rules

Classifier for classification

Test Data → Classification Rules → Test Data & Prediction Result.



CLASSIFICATION AND PREDICTION PROCESS



COMPARISON OF CLASSIFICATION & PREDICTION

- SPEED → Value of Attributes
- ROBUSTNESS → Noisy Data
- SCALABILITY → Class prediction Label
- INTERPRETABILITY → Efficiency

$$\text{Gain Ratio}(A) = \frac{\text{Gain}(A)}{\text{Split Info}_A(D)}$$

Decision Tree Induction

- Multi - dimensional data
- Domain - knowledge
- Top- down recursive
- Divide- conquer Manner

Bayes Theorem

$$P(H/X) = P(X/H)\,P(X)$$

Rule - Based Classification

If condition THEN rules

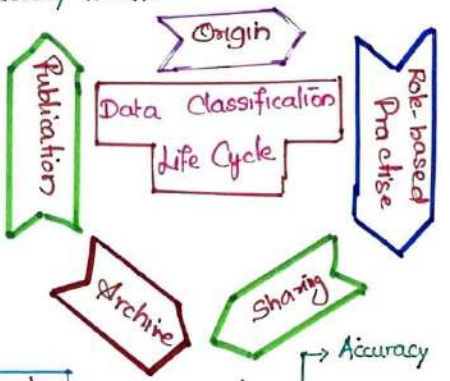
R1 : (age = youth) (Student = yes) => buys = yes .

Developing the classifier → How classification Works. → Data classification Process

Applying classifier for classification

Essential Concept of Measurement

- Measurement of Internet Traffic
- Understanding Measurement
- Per- Packet and Per- flow Measurement
- Evaluation Criteria for Perf. Measurement
- Accuracy Recall.



Tree- Pruning → Pre- pruning, Post- pruning

- Accuracy
- Error rate
- Recall
- specificity
- Precision
- F-Score

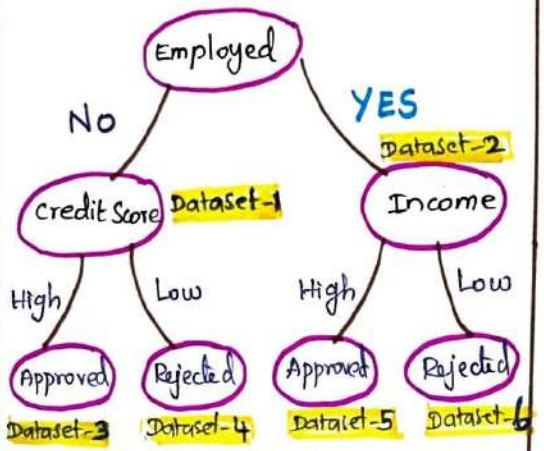
$$\text{Accuracy} = \frac{TP + TN}{P + N}$$

DECISION TREE

* Tree based classification & regression can be done using 'Decision Tree'.
* Process :
  Dataset → D.T Algorithm → classifies the data

Example :

Decides if the loan should be Approved / Rejected



→ Approve the loan {Employed with high credit score}
&
→ Approve the Loan {Employed with high income}
→ Reject the loan - {Unemployed with low credit score}
→ Reject the loan {Employed with low income}

∵ What is the source to repay the loan

Challenges :

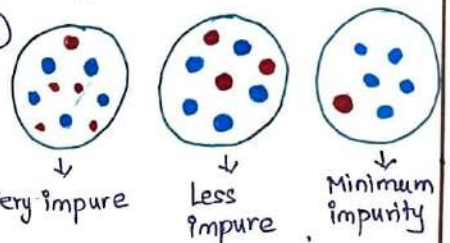
* Attribute Selection
* Popular Attribute Selection
  1. Information - Gain (IG)
  2. GINI Index

1. Information Gain :- (IG):

→ IG is calculated for a split by subtracting the weighted "entropies" of each branch from the "original entropy".
→ i.e measure the change in entropy

(i) → Entropy : → Measures the impurity [or] uncertainty in a group of observations.

→ eg)



→ Formula :

$$E = -\sum_{i=1}^{N} P_i \cdot \log_2 P_i$$

Example:

Set X = {a, a, a, b, b, b, b, b}

Total instances = 8
Instance b = 5
Instance a = 3

$$\text{Entropy} = -\left[\frac{3}{8}\log\frac{3}{8} + \frac{5}{8}\log\frac{5}{8}\right]$$
$$= 0.954$$

(ii) GINI Index :-

→ Gini Index & entropy are the criterion for calculating information gain (IG).
→ D.T. algorithm use IG to split a node. Both Gini Index & Entropy are the measures of impurity of a node.
→ Formula

$$\text{Gini} = 1 - \sum_{i=1}^{n} P^2(Ci)$$

Where,
P(Ci) → The probability [or] percentage of class Ci in a node.

→ $IG = \left(\text{Entropy of Parent node}\right) - \left(\text{Sum of weighted entropies of child node}\right)$

Decision Tree Advantages & Disadvantages :-

Advantages :-

→ They are very fast & efficient compare to KNN
→ Easy to understand, interpret & visualize
→ All type of data such as numerical & categorical are possible.

Disadvantages :-

→ Training the model take higher time
→ Inadequate for applying regression & predicting continuous values.

BAYESIAN CLASSIFICATION

- Classification technique based on "Bayes' theorem" with an assumption of independent among features.
- The presence of one-feature doesnot affect the other feature. All the features are independent of each other.
- Parametric model [Assumptions about a form of a function, to ease the learning-process]
- Bayes' Formula :-

$$P(A/B) = \frac{P(B/A) \cdot P(A)}{P(B)}$$

- Bayesian classification uses Bayes theorem to predict the occurance of any event.
- A & B → Events $P(B) \neq 0$
- $P(A/B)$ → Conditional Probability
  [Occurance of event A is given that B is true]
- $P(B/A)$ → Conditional Probability → Likelihood
  [Occurance of event B is given that A is true]
- $P(A)$ & $P(B)$ → Probabilities of observing A & B independently of each other.
  ↳ called {Marginal Probability}

Example :

Fruit = {Yellow, sweet, Long}

FRUIT	Yellow	Sweet	Long	TOTAL
Orange	350	450	0	650
Banana	400	300	350	400
Others	50	100	50	150
TOTAL	800	850	400	1200

→ To predict & classify which one has the most probability of yellow, sweet & Long.

Probability (Yellow / orange) = $\frac{P(\text{orange/yellow}) \cdot P(\text{Yellow})}{P(\text{orange})}$

$$= \frac{\frac{350}{800} \times \frac{800}{1200}}{\frac{650}{1200}}$$

$$= \frac{0.4375 \times 0.667}{0.541667}$$

$$= 0.5387 //$$

NOTE
- 800 → Yellow Fruit
- 1200 → Tot. Fruits
- 350 → Yellow orange
- 800 → Total
- 650 → Orange Fruit
- 1200 → Total

∴ Probability of Yellow / orange = 0.53

Prob.. (sweet / orange) = $\frac{P\left(\frac{\text{orange}}{\text{sweet}}\right) \cdot P(\text{sweet})}{P(\text{orange})}$

$$= \frac{\frac{450}{850} \times \frac{850}{1200}}{\frac{650}{1200}} = \frac{0.529 \times 0.7083}{0.541667}$$

$$= 0.6917 //$$

∴ Prob.. (sweet / orange) = 0.69

P (Fruit / orange) = P (Yellow / orange) × P (sweet / orange) × P (long / orange)

Note : {P (Long / orange) = 0}

$= 0.53 \times 0.69 \times 0 = \boxed{0}$

P (Fruit / Banana) = $P\left(\frac{\text{Yellow}}{\text{Banana}}\right) \times P\left(\frac{\text{sweet}}{\text{Banana}}\right) \times P\left(\frac{\text{Long}}{\text{Banana}}\right)$

$= 1 \times 0.75 \times 0.89 = \boxed{0.65}$

P (Fruit / Others) = $P\left(\frac{\text{Yellow}}{\text{others}}\right) \times P\left(\frac{\text{sweet}}{\text{others}}\right) \times P\left(\frac{\text{Long}}{\text{others}}\right)$

$= 0.33 \times 0.66 \times 0.33$

$= \boxed{0.072}$

From the above,

→ Fruit of orange is zero, ∴ eliminated

→ Fruit of Banana & Fruit of other fruits } Which is the most probability of Yellow, Sweet & Long ?

Conclusion :-

Banana is the fruit which is having yellow, sweet & Long Compare to other all given fruits.

Rule Based classification:

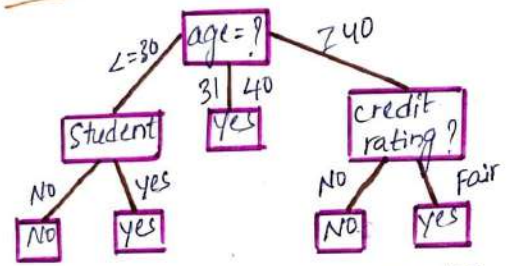
* It uses set of IF-THEN rules for classifications

*Example:

If age = "youth" and student = "yes"
then buys computer = "yes"

* Rule Extraction from decision tree

Example:-



*Chances to buy the computer

→ person with age <=30 & student
(more chances to buy a computer)

→ person with age >=40 & credit score is fair
(more chances to buy a computer)

→ person with age <=30 & not a student
(less chances to buy a computer)

→ person with age >=40 & low credit score
(less chances to buy a computer

Assessment of a rule:-

* Coverage &
* Accuracy [how correctly fit the target function / how much confidence?]

R → Rule

N-covers = No. of tuples covered by R

N-Correct = No. of tuples correctly classified by R

* If more than one rule are triggered need conflict resolution.

→ **Size ordering:** Assign the highest priority to the triggered value

→ **class based ordering:** Decreasing order of prevalance (or) Misclassification cost per class

→ **Rule based ordering:** Rules are organized into one long priority list, according to some measures of quality (or) experts.

Rule induction:

*Sequential covering Alg/method used

*Seq covering -> Extracts rules from dataset directly

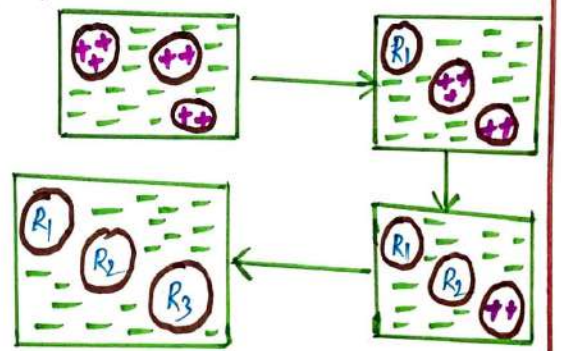
*Seq cov/- -> Mainly based on one of the Evaluation measure
i) Accuracy ii) coverage

* Rules are learned sequentially
* Rules are learned one @ a time.
* i.e sequentially covered every rule (one by one)

Algorithm:-

1) creates an Empty set of decision (rules) list.
2) A function called "learned one-rule" is used
   ↳ It extracts best rule for y class y
   → If all training ∈ class y => +ve records (accepted)
   → If all training ∉ class y => -ve records (Rejected)
3) Get only desirable values (only +ve)
4) Eliminate records (-ve)
5) New rule is added to the bottom of R

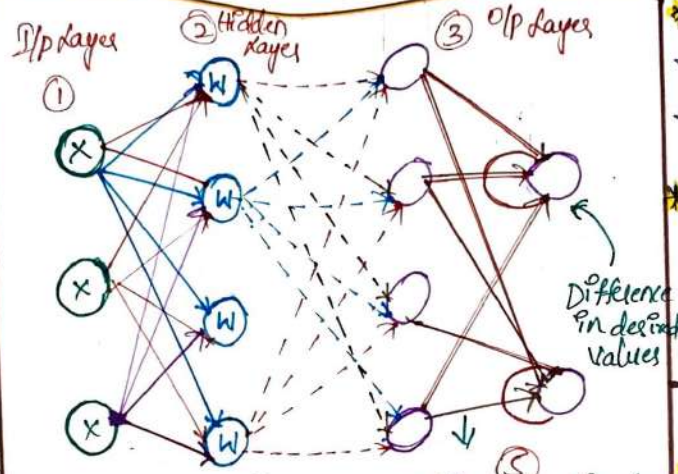
Example:-



CLASSIFICATION BY BACK-PROPAGATION

Back propagation :-

* Neurobiologistic to develop and list-Computation analogeus of neuron
* Artificial Neural N/w (ANN) uses back-propagation as a learning algorithm "to compute a gradient descent with respect to weights."
* Desired o/ps are compared to achieved system outputs & then the system are turned by adjusting connections weights
  This process to narrow the difference b/w two as much as possible
* perceptron consists of 2-types of nodes:-
  - → i/p-node : Represent I/p attributes
  - → o/p-node : Represent model o/p
* Each node connected with weight to o/p node
* Back propagation contains the following layers:-
  - → I/P layers: → Receives i/ps → x
  - → O/p layers: → Difference in desired values
  - → Hidden layers: → Caluculates the o/p & data is ready @ the o/p layers.
* The gradient-loss-function Caluculates the difference b/w the n/w-o/p & its probable o/p

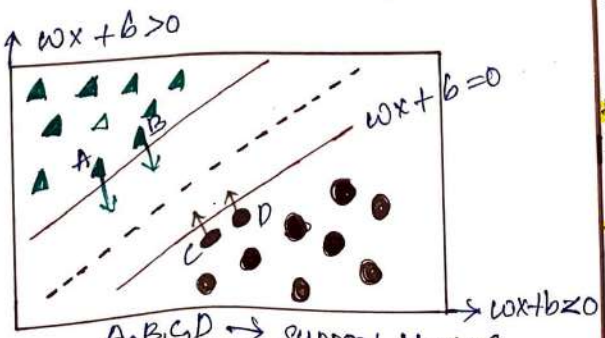


Perceptron model:

$$Y = I\left(\sum w_i x_i - t\right) \text{ (or) } Y = \text{sign}\left(\sum w_i x_i - t\right)$$

SUPPORT VECTOR MACHINE (SVM)

* SVM - supervised machine learning algorithm
* → Used for both classification & prediction
* Hyperplane - used to seperate the data
* MMC → Maximum Margin classifier helps to pick the best hyperplane



A, B, C, D → support Vectors

* $wx + b < 0$ [or] ≥ 0 [or] $= 0$

* Increase the max-Margin width, the points support vectors found well, if need
* Sometimes, deleting the support-Vectors easily change / pick the position of the optimal hyperplane.

Association Classification

* Mine data to find strong association b/w frequent pattern
* Association Rules :-

$$P_1 \wedge P_2 \wedge \dots \wedge P_n \rightarrow \text{"A class} = C\text{"}$$

(confidence, support)

* Association classification methods more accurate than other method
* Mine possible association-rules
* Classification based on multiple association rules.
  - → statistical analysis on multiple values
* Classification based on predictive association rule
  - → Generation of predictive rules but allow covered rules to retain with reduced weight.

EVALUATING ACCURACY OF A CLASSIFIER - CROSS VALIDATION. BOOTSRAP.

Measuring Accuracy:

* Model evaluation is the process through which we quantify of a system's predictions.
* The performance & usage of the classifiers model is decided in terms of accuracy measures at the end.
* Measuring accuracy using validation set of class-labelled tuples.

Confusion matrix:

True positives (TP)	False Negatives (FN)
False positives (FP)	True Negatives (TN)

* TP - predicted +ve & its true
* TN - predicted -ve & its true
* FP - predicted +ve & its false
* FN - predicted -ve & its false.

A confusion matrix - NxN - matrix, N → No. of class being predicted.

classifier Accuracy:

→ % of test_set tuples that are correctly classified.

Measures:

Accuracy:-

The proportion of the total No. of predictions that were corrected.

$$Acc = \frac{TP+TN}{TP+FP+TN+FN}$$

Error Rate:

$$ER = 1 - Accuracy$$

Sensitivity: True +ve recognition rate [or] Recall [or] True +ve rate.

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

specificity:

True -ve recognition rate

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

* A.k.A as % of -ve instances out of the total actual -ve instances.

classifier Evaluation matrix:

Precision: % of tuples classified as +ve are actually +ve.

$$Precision = \frac{TP}{TP+FP}$$

Recall:

what % positive tuples did the classifier labled as positive?

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

F1 - Measure / F-Score:-

→ Harmonic mean of precision recall.

→ Higher F1-Score is the better value.

$$F = \frac{2 \times precision \times Recall}{precision + Recall}$$

Evaluating classifier Accuracy:

Holdout: → Reserves 1/2 for training & 1/2 for testing

→ Reserves 2/3 for training & 1/3 for testing.

satisfied sampling :- Each class is represented with approximation in training a test.

cross validation:

step 1: Data split into k-subsets of equals.

step 2: Each subset is trained used for testing & remainder for training subset satisfied before cross-validation.

Evaluating classifier accuracy:

Bootstrap:

→ Works well with small data sets.

→ Samples the given training tuples uniformly with replacement

→ Each time a tuple is selected

→ To be selected again & readded to the training set.

632 bootstrap:

→ A data set with 'd' tuples is sampled 'd' times, with replacement resulting a training set of 'd' samples.

→ The data tuples that did not make it into the training set end up forming the test set.

63.2 → end with bootstrap.

36.8 → test set.

Repeat the sampling procedure k-times.

$$\overline{Acc}(M) = \frac{1}{k}\sum_{i=1}^{k} (.632 \times Acc(M_i)_{test\_set} + 0.368 \times Acc(M_i)_{train\_set})$$

CLUSTER ANALYSIS / CLUSTERING

- Clustering is the process of partitioning a set of data into Subclasses / Subsets

Application Area:-

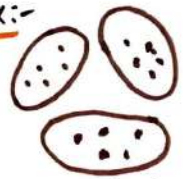
→ Image Pattern Recognition, web Search, Market - Basket Analysis etc.,

Basic clustering Methods:-

Partitioning Methods

→ A divisive data--objects into "non-overlapping Subsets" s.t each data-object is in exactly one-subset.

Ex:-



k-Means clustering is an example for partitioning method

Hierarchical Methods

Agglomerative Methods

Divisive Methods

→ A set of nested clusters organized as a hierarchical tree.

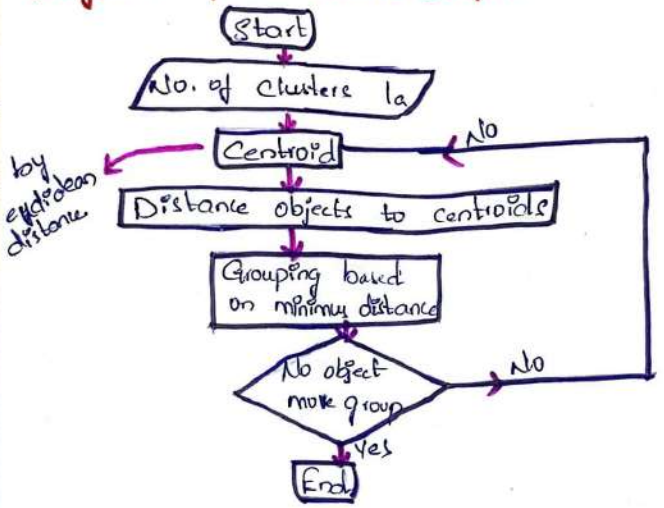
Ex:-

Chameleon is an example for Hierarchical clustering.

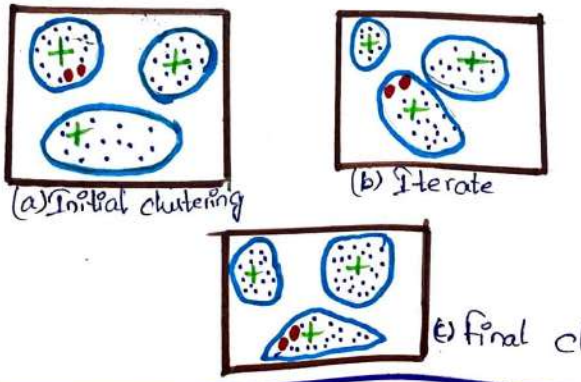
K-Means clustering:-

- Partitioning clustering approach
- Each cluster is associated with a "centroid" (i.e centrepoint)
- Each point is assigned to the cluster with the "closest Centroid".
- No. of clusters, K- must be Specified.
- closeness is measured by "Euclidean Distance"
- Initial Centroid are often chosen "randomly"

Algorithm / Flow chart steps:

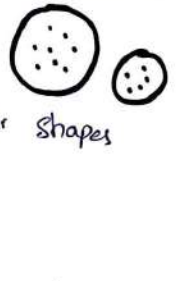


Example:-



Limitations of k-Means:-

- K-Means has problems when clusters are of differing → Sizes
  → Densities &
  → Non-globular shapes



→ if the data contains Outliers / noise.

K-Medoids

- K-Medoids also called "Partitioning Around Medoid"
- The cost in K-Medoids algorithm is given as

$$C = \sum_{c_i} \sum_{p \in c_i} |p_i - c_i|$$

Algorithm:-

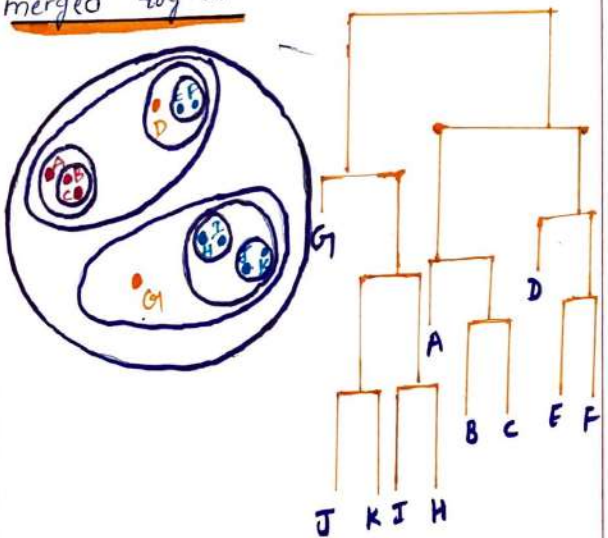
1. **Initialize**: Select k random points out of the n data points as the medoids.
2. Associate each data point to the closest medoid by using any common distance metric methods.
3. **While the cost decreases**: For each medoid m, for each data o point which is not a medoid:
   → Swap m and o, associate each data point to the closest medoid and recompute the cost.
   → If the total cost is more than that in the previous step, undo the swap.

Heirarchical clustering:-

* Grouping data into a tree of clusters
* It begins by treating every data point as a separate cluster.

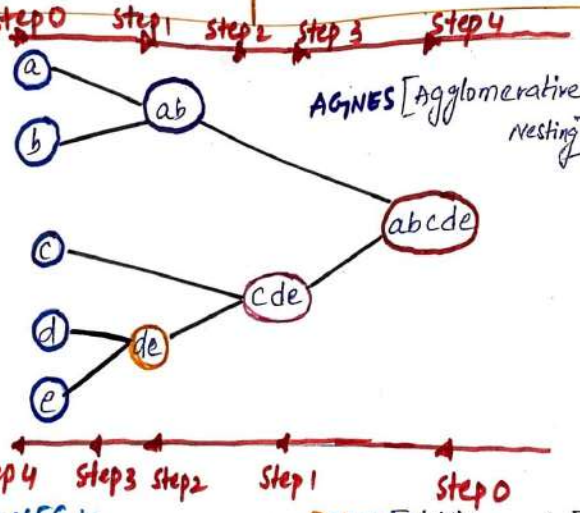
Steps:-

1) Identify the 2 clusters which can be closest together.
2) Merge the 2 maximum comparable clusters. we need to continue these steps until all the clusters are merged together.



AGGLOMERATIVE VS DIVISE

AGGLOMERATIVE	DIVISE
* Bottom up Approach	* Top down approach
* It successively merges the objects (or) group close to one another until all the groups.	* In each successive iteration a cluster is split into smaller clusters, until Eventually each object is in one cluster, or a termination condition hold
Ex:- AGNES - Agglomerative NESTing	Ex:- DIANA Divisive ANALysis.



AGNES [Agglomerative Nesting]

DIANA [Divisive Analysis]

AGNES:-

* use the "single link" method & "dissimilarity matrix".
* Merge the node that have the least dissimilarity
* Go on in a non-descending fashion
* Eventually all nodes belongs to same cluster.

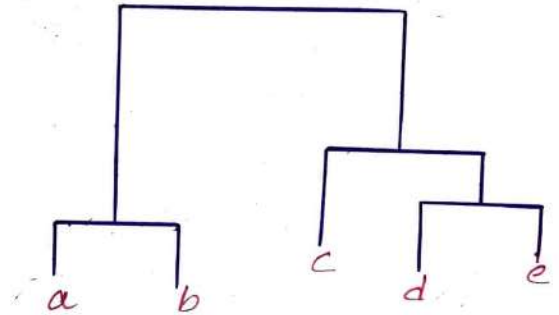
DIANA:

* Inverse order of AGNES
* Eventually each nodes forms a cluster on its own.

Steps

1) Initially all points to the dataset belong to one cluster
2) partition the cluster into two least similar clusters
3) proceed recursively to form new cluster until the desired number of cluster is obtain.

Dendogram:-



* Dendogram shows the clusters are merged
* A tree structure called a "dendogram" is commonly used to represent the process of heirarchical clustering
* Either agglomerative/divisive can be used.

FREQUENT PATTERN (FP) GROWTH ALGORITHM

FP-Growth Algorithm in Data Mining

- FP-Growth is an improved version of the Apriori algorithm which is used for frequent-pattern mining (also known as Association Rule Mining).
- Association Rule Mining can be viewed as 2-step process:
  1) Find all frequent item sets
     - Apriori Algorithm
     - FP-Growth Algorithm
  2) Generate strong association rules from the frequent itemsets.
     - These rules must satisfy the following:
       "Minimum Support" &
       "Maximum Confidence"

FP-Tree:

- FP-growth alg. represented the database in the form of a tree called "FP-Tree".
- Purpose: To mine the frequent pattern
- The root node represents null, while the lower-node reps.. item-sets.
- It is a compact data-structure that stores quantitative infn.. about freq-patterns in a database.

Algorithm / Pseudocode :-

Procedure FP-Growth * (T)

Input : A conditional FP-Tree T

Output : The complete set of all FPs corresponding to T.

Method :

1. If T only contains a single branch B
2. For each subset Y of the set of items in B
3. Output itemset YUT.base with count = Smallest count of nodes in Y;
4. else
   for each i in T.header do begin
5. Output Y = T.base U {i} with i.count;
6. if T.FP-array is defined
7. Construct a new header table for Y's-FP-Tree from T.FP-array
8. else
   construct a new header-table from T.
9. Construct Y's conditional FP-Tree Ty & Possibly if FP-array Ay;
10. If Ty ≠ ϕ
11. Call FP-growth * (Ty);
12. End.

APRIORI ALGORITHM.

Steps in Apriori :-

step-1 : Determine the support of itemsets in the transactional database & select the "minimum support" & "confidence".

step-2 :
Take all supports in the transaction with higher support value than the minimum / selected support value.

step-3 :
Find all the rules of these subsets that have higher confidence value than the threshold [or] min-confidence.

step-4 :
Sort the rules as the decreasing order to fit.

Flow chart :

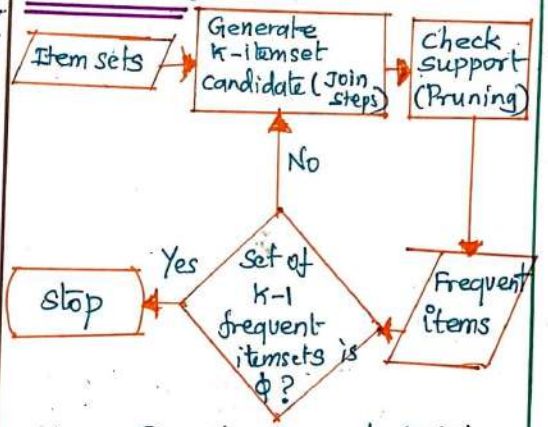
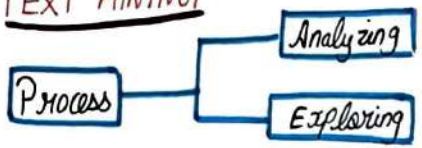


Fig. Flowchart of Apriori

TEXT MINING



Findings / Outputs => concepts, pattern, topic, etc....

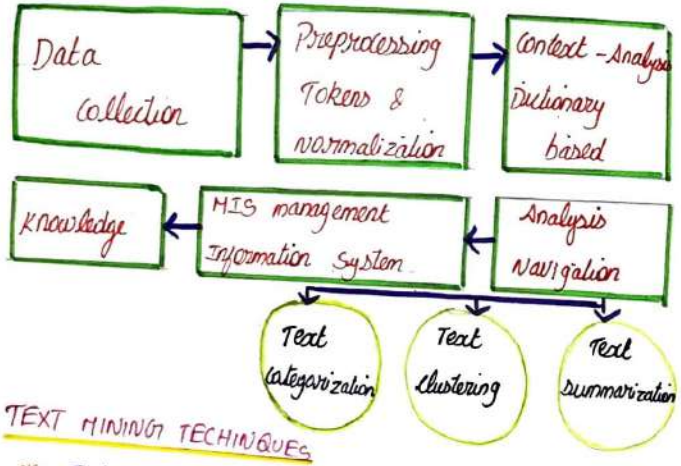
METHODS:-

STEP 1: Gathering unstructured information

STEP 2: Preprocessing - cleaning

STEP 3: Analysis - Analyze the patterns

STEP 4: Information, extraction, retrieval, categorization, clustering & summerization



TEXT MINING TECHNIQUES

* Information extraction
* Information Retrieval
* Natural Language processing
* Text clustering analysis
* Text Summerization

APPLICATION AREAS

* Digital Library
* Academic & Research field
* Life Science
* Social media
* Business Intelligence

KEY-WORD BASED ASSOCIATION

-> collects set of keywords / items that one occured frequently

-> Pre process the text -> by passing, stemming, removing the stop words

-> Implement association mining algorithm

-> Each document consider as transactions

-> view a set of keywords in document as a set of item in the transaction

AUTOMATIC DOCUMENT CLASSIFICATION

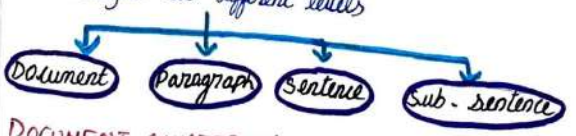
-> A.K.A categorization

-> process of managing text & unstructured information by categorization (or) clustering text

-> Enables users to organize content quickly

TEXT CLASSIFICATION

-> Analyze at different levels

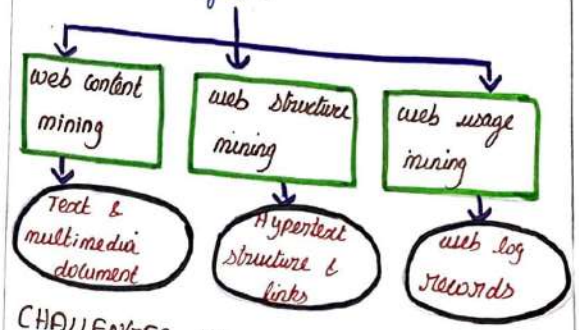


DOCUMENT CLUSTERING

-> Group the related document based on their content

WEB MINING

* Data acquired through "web crawler" [or] "web analysis".



CHALLENGES IN WEB MINING

* The complexity of web pages
* The web is a dynamic data source
* Diversity of client networks
* Relevancy of data
* The web is too broad

ADVANTAGES

-> Find useful information from the user interactions

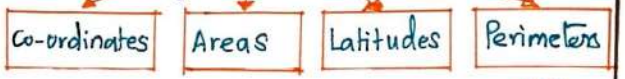
-> understanding consumers behaviour, need & buying patterns

APPLICATIONS

* Google search engine
* CRM (customer relationship mgt)
* Marketing and conversion tool
* Audience behaviour analysis
* Testing and analysis of a site
* Data analysis on website and application

MINING SPATIAL DATA

- Geographical [or] spatial information to produce business intelligence (or) other results.
- Data related to spatial description of the objects such as



etc.,

* Spatial Relations

* Finding & Analyze helps :-
  - → Earthquake points
  - → Climate / Weather predictions
  - → Google Map
  - → GPS (Global Positioning System)
  - → Trend Analysis
* Clustering Analysis Methods :-
  - → PAM → Partitioning Around Medoids [Similar to K-Means clustering]
  - → PAM divides data into groups based on medoids.
* Methods [others] :-
  - → Spatial Auto Correlation – Measure of
  - (a) dependency among points in a spatial neighborhood
  - (b) → Spatial Heterogenity – variation in events, features & relationships across a region.
* Geographic Warehouse :-
  - → Built to collect data from various resources.
* Spatial Prediction :-
  - → Identify the relationship between variables in different datasets.

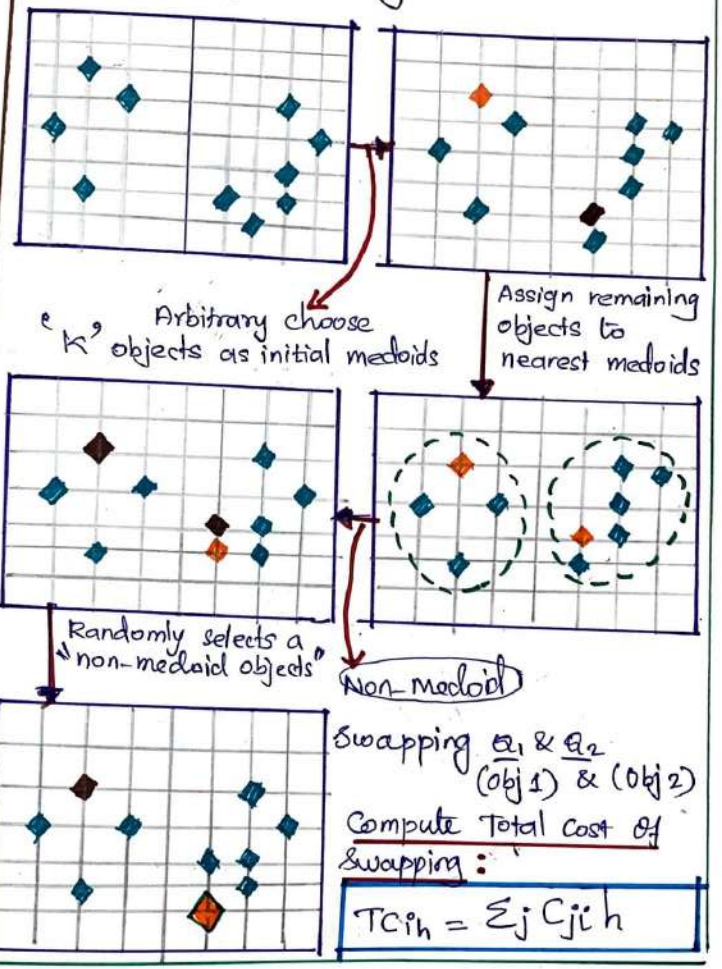
Types of Spatial Data :

1) Feature Data :
- → Follow the vector data model
- → Represents the entity of the real world. i.e., roads, trees, buildings etc.,
- → This information can be visually represented in the form of a point, line (or) polygon.

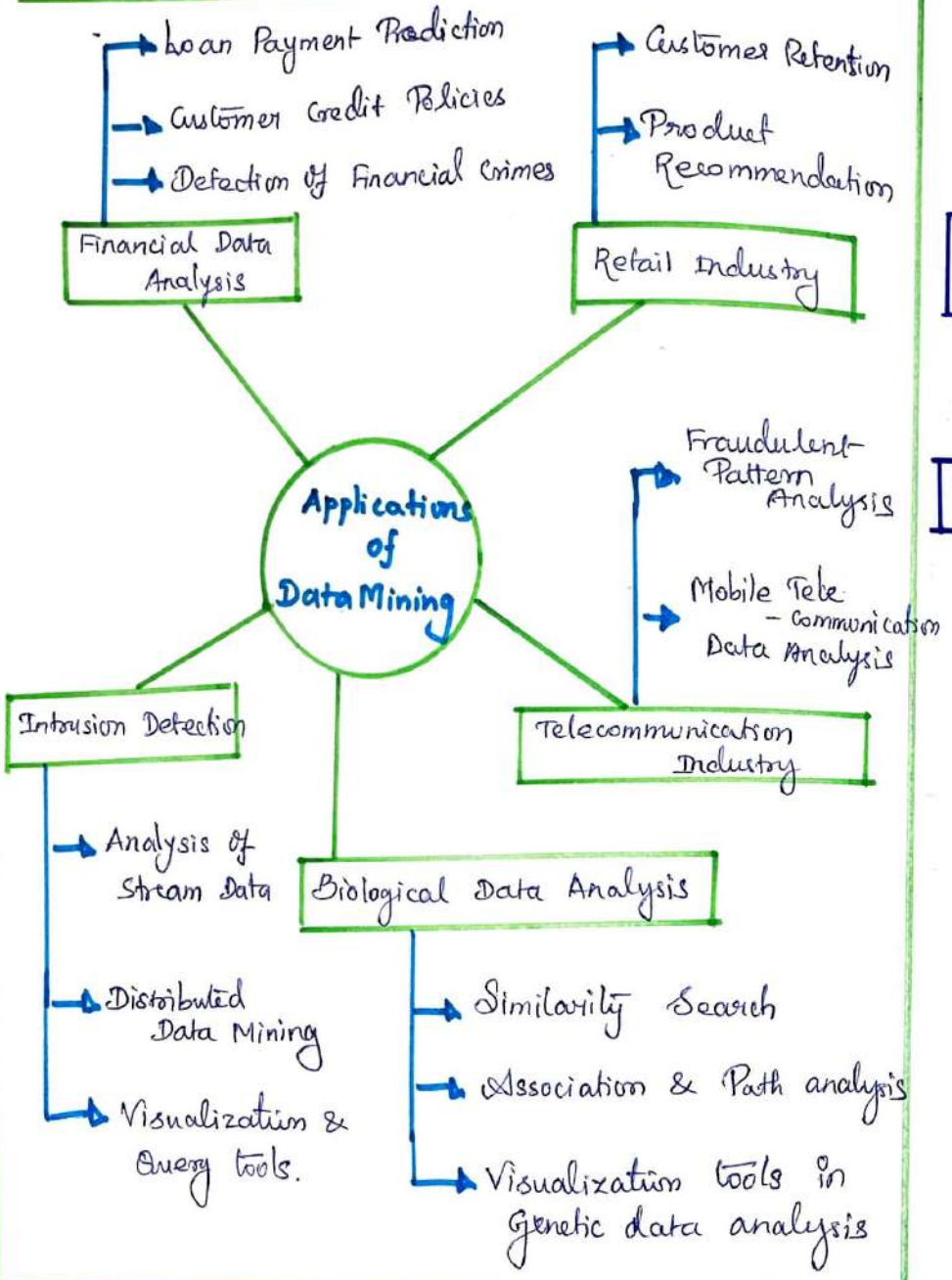
2) Coverage Data :
- → Follows the raster data model
- → Data contains the mapping of continuous data in space.
- → Represented as a satellite image, Digital surface model etc.,
- → The visual representation of coverage data is in the form of a "Grid" [or] triangulated irregular network

Partitioning Around Medoids (PAM)

* Find a sequence of objects called "Medoids" that are centrally located in clusters.
* Starts from initial set of medoids & iteratively replaces one of the medoids by one of the non-medoids
  - ↓
  - If it improves the total distance of the resulting clustering.



APPLICATIONS IN DATA MINING



TRENDS IN DATA MINING