Data Science

DS is an interdisciplinary field of scientific methods, processes and systems to extract unseen patterns, derive meaningful information and make business prediction. It uses complex ML algorithms to build predictive model.

Data Science Life Cycle

Step 1: Problem definition — ask relevant questions and define obj for the problem that needs to be tackeled.

Step 2: Collecting data (data preparation & data investigation) — gather & scrap the raw data necessary for project formulation.

Step 3: Preprocessing data (Data-cleaning) - fix the inconsistencies within the data set & handle the missing values.

Step 4: Data exploration (EDA - Exploratory Data Analysis)
Form hypothesis about your defined problem by visually analysing the data.

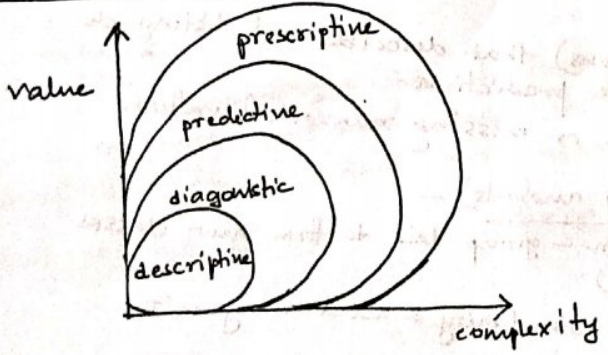
MCDM - Multi criteria Decision Making

Step 5: Feature engineering — select important feature and construct more meaningful ones using the data that we have. (PCA - Principle Component Analysis)

Step 6: Predictive modelling — train ML models, evaluate their performance and use them to make predictions.

Step 7: Data visualization — communicate the findings using plots and interactive visualisations (eg. bar plot, pie plot, etc.)

Types of Data Analytics



- **Descriptive analytics**
  It looks at data to examine, understand, and describe something that has already happened. (descriptive statistical analysis)

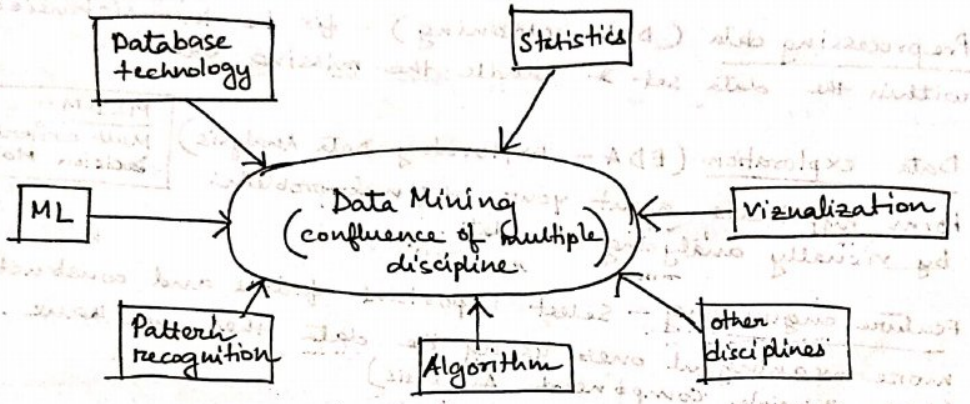
- **Diagnostic analytics**
  It goes deeper than descriptive analytics by seeking to understand the why behind what happened.

- predictive analytics – It relies on historical data, past treands and assumptions to answers questions about what will happen in the future.
- prescriptive analytics – It aims to identify specific actions that an individual/organisation should take to reach future targets or goals.

Data Mining

[Knowledge Discovery from Databases (KDD)]

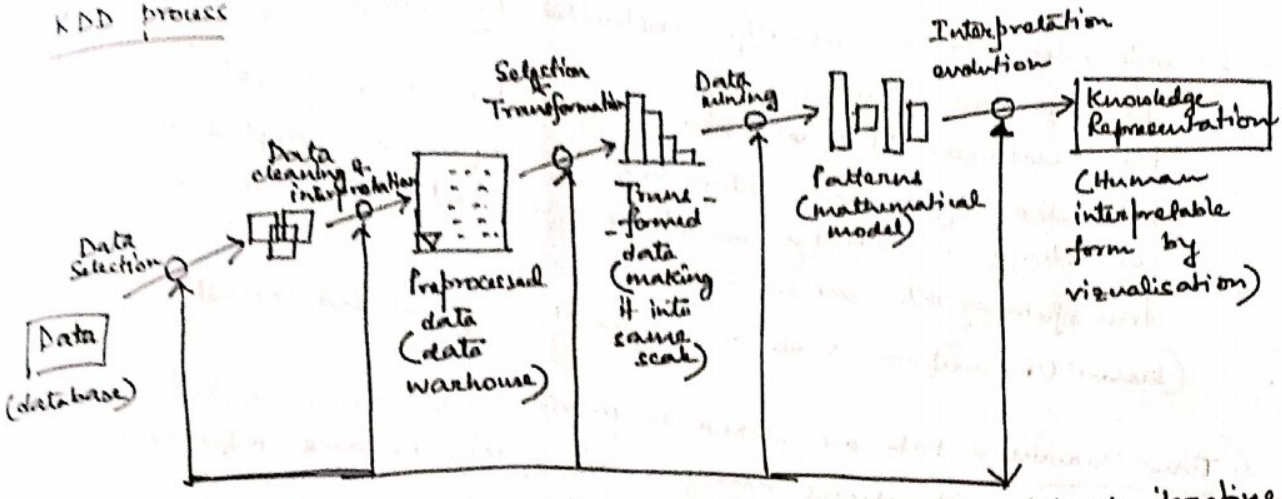
Extraction of interesting (non trivial, implicit, previously unknown and potentially useful) patters/knowledge from huge amount of data.



Data Mining Functionalities

- Multi dimensional concept description – characterization & discrimination. Generalise, summarize and contrast data characteristics.
- Frequent patterns, association, corelation (Association Rule Mining / ARM)
  eg – market basket analysis
- classification & prediction –
  (i) construct models (functions) that describe and distinguish classes/concepts for future prediction.
  (ii) predict some unknown or missing numerical values.
- cluster analysis (exploratory analysis) –
  (i) class lable is unknown – group data to form new classes (unsupervised learning)
  (ii) maximising intra class similarity & minimising inter class similarity.
- outlier analysis (anomaly detection) –
  data obj that doesn't comply with the general behaviour of the dataset.
- trend & evolution analysis –
  (i) trend & deviation (ii) sequential pattern mining
  (iii) periodicity analysis (iv) similarity based analysis.
- other pattern directed / statistical analysis

KDD process



Knowledge discovery process as depicted in figure, consists of iterative sequence of steps.

1. Data Selection — Data relevant for analysis task is retrived from db.
2. Data integration — multiple data sources may be combined.
3. Data cleaning — to remove noise & inconsistent data
4. Data transformation — where data are transformed or consolidated into forms appropriate for mining by performing summary or aggregation op.
5. Data mining — An essential process where intelligent methods are applied in order to extract data patterns.
6. Pattern evaluation — To identify the truely interesting patterns representing knowledge based on some interesting measures.
7. Knowledge Representation — where visualisation & knowledge representation techniques are used to present the mined knowledge to the user.

Data Warehouse

It is the act of organising & storing data in a way so as to make its retrival efficient and insightful. It is also called the process of transforming data into information.

According to Bill Inmon (1990) — Data warehouse is a →
- subject oriented
- integrated
- non volatile
- time variant

collection of data in supports of management's decision making.

1. Subject Oriented — A data warehouse is organised around major subjects. It provides a simple & concise view around particular subject issues by excluding data that is not useful in the decision support process.

2. integrated – It is usually constructed by integrating multiple heterogeneous sources.
Eg – Relational DBs, online transaction records, etc.
Data cleaning & data integration techniques are applied to ensure consistency in naming conventions, attribute measures & also transforming into same scale.
(Biased Un-uniform scale results in having biased results)

3. Time Variant – Data are stored to provide info from a historical perspective. It allows analysis of past, related information to present and enables forecast to future.

4. Non-volatile – Data is not updated or deleted from warehouse in real time.

Difference b/w OLTP & OLAP

Features	Online Transaction Processing (OLTP)	Online Analytical Processing (OLAP)
characteristics	Operational processing	Informational processing
Orientation	Transaction analysis	Analysis of data
User	DB professionals	Knowledge workers (Data scientist, data analyst)
Functions	Day 2 day operations	Long term informational requirement (Decision Support System) [DSS]
data	current transaction	Historical Data (Accuracy maintained over time)
Unit of work	Simple transaction queries	Complex queries
response time	requires faster response time.	Can have longer response time
access	read & write	mostly read
focus	data i/p	information out
No. of Records accessed	100s of data	millions of data (Big data)
No. of users	Thousands	Hundreds (specific requirement)
DB size	GB to higher order GB	≥ TB.

Functions of Data Warehouse Tools

(ETL – Extraction, Transformation, Loading)

① **Data Extraction** – gathers data from multiple external heterogeneous sources

② **Data Transformation** – converts data from host format to warehouse format

③ **Data Loading** – It sorts, summarises, consolidates, computes, views, checks integrity and build indices & partitions.

Besides E,T,L, 2 other terms are — (i) Data cleaning & (ii) data refreshing They are important steps in improving the data quality & subsequent data mining results.

Association Rule Mining (ARM)

Given a set of transactions, find rules that will predict the occurrance of an item based on the occurances of other items in the transaction.

Eg - Market Basket transaction

TID	Items
1	Bread, Milk (k=2)
2	Bread, Egg, Butter, Ketchup (k=4)
3	Milk, Eggs, Butter, Cookies
4	Bread, Milk, Eggs, Butter
5	Bread, Milk, Eggs, Cookies

* unsupervised

Egs of Association Rules

$\{Eggs\} \Longrightarrow \{Butter\}$

$\{Milk, Bread\} \Rightarrow \{Eggs\}$

{Milk, Bread} — Antecedent (LHS of AR)

{Eggs} — Consequent (RHS of AR)

Item Set$^{(k)}$ — A collection of 1 or more items.

Rule Evaluation Metrics

1. **Support count / frequency / count of the item set (σ)** - freq. of occurance of an item set.

$\sigma\{Milk, Bread, Eggs\} = 2$ (Milk, Bread, Eggs occuring together in DS; TID = 4 & 5)

2. **Support / Relative Support(s)** - Fraction of transactions that contain an item set. Range = [0, 1]

$s\{Milk, Bread, Eggs\} = \frac{2}{5} = 0.40$

(Total no. of transactions in DS)

3. **Frequent Item Set** - An item set, whose support is greater than or equals to a minimum support threshold (minsup thresd)

4. **Confidence** - It measures how often items in Y appear in transactions that contain X.

$P(Y|X) = \text{confidence}(X \Rightarrow Y)$ Range = [0, 1]

$= \frac{\text{support}(X \cup Y)}{\text{support}(X)}$

$\{Milk, Bread\} \Rightarrow \{Eggs\}$

$$\text{confidence} = \frac{\text{support}(\{Milk, Bread, Eggs\})}{\text{support}(\{Milk, Bread\})}$$

$$= \frac{0.4}{0.6} = \frac{2}{3} = 0.667$$

$\therefore$ 67% of customers who purchased {milk, bread}, also purchased Eggs.

ARM Task

Given a set of transactions, T, the goal of ARM is to find all the rules having

- support $\geq$ min sup threshold
- confidence $\geq$ min conf threshold (min. confidence threshold)

[Note: A major challenge in mining frequent item sets from a large dataset is the fact that such mining often generates a huge no. of item sets satisfying the minimum support threshold, especially when min support is set low. This is because, if an item set is frequent, each of its subsets is frequent as well. A long item set will contain a combinatorial no. of shorter, frequent sub item sets.]

Lift (Correlation Measure) –

Support & confidence measures are insufficient at filtering out uninteresting association rules. To tackle this weakness, a co-relation measure can be used to augment the support-confidence framework for association rules.

$$\text{lift}(A,B) = \frac{\text{Support}(A \cup B)}{\text{Support}(A) \cdot \text{support}(B)}$$

$$= \frac{\text{confidence}(A \& B)}{\text{support}(B)}$$

There are 3 cases :–

(i) lift < 1 (occurance of A is negatively co-related with that of B)

(ii) lift > 1 (+ve co-relation)

(iii) lift = 1 (confidence = support)
(A & B are independent and there is no co-relation b/w them)

~~TDB~~

Q. A TDB has 500 transactions out of which X appears in 77 transaction and out of these 77 transactions another item ~~set~~ 'Y' appears in 35 transactions. Find the confidence of the rule $x \Rightarrow y$

$$\text{confidence} = \frac{\text{Support}(X \cup Y)}{\text{support}(X)} = \frac{35/500}{77/500} = \frac{35}{77} = 0.454$$

Q. let support(x) = 0.2, & support(y) = 0.4 and support(X ∪ Y) = 0.1
Find lift (x, y) ?

$$\therefore \text{lift}(x,y) = \frac{\text{support}(X \cup Y)}{\text{support}(X)\ \text{support}(Y)} = \frac{0.1}{0.2 \times 0.4} = \frac{0.1}{0.08} = 1.25$$

Q. A TDB has 6 items, such as U, V, W, X, Y, Z. Out of 10 transactions U, V, W, X, Y, Z appears in 2, 5, 7, 3, 4 and 6 transactions respectively. let the min^m support threshold is 0.5. Find the list of rare items.

Support(u) = 0.2 (<0.5), sup(v) = 0.5, sup(w) = 07, sup(x) = 0.3 (<0.5), sup(y) = 0.4 (<0.5)
sup(z) = 0.6

∴ U, X, Y are rare items.

Apriori Algorithm

- **Apriori Principle** – If an item set is frequent, then all of its subsets must also be frequent (Reducing no. of candidate item sets)

Eg- If {A, B, C, D} is a freq. item set, then any subset like {A, B, C} or {B, D} is also a freq item set.

Apriori Principle holds due to the following property of the support measure:

$\forall X, Y : (X \subseteq Y) \Rightarrow \text{Support}(X) \geq \text{Support}(Y)$

Anti monotone property of Support.

↳ Support of an item set never exceeds the support of its subsets.

- **Apriori Algo**

1. let k = 1
2. Generate frequent itemsets of length 1 (1-itemset)
3. Repeat until no new frequent itemsets are identified
   i. Generate length (k+1) candidate itemsets from length k frequent itemsets
   ii. Prune candidate itemsets containing subsets of length k that are infrequent
   iii. Count the support of each candidate by scanning the database.
   iv. Eliminate candidates that are infrequent leaving only those that are frequent.
4. Generate strong rules (based on confidence value)

Q. For the following given transaction data set, generate rules using Apriori Algo.
Consider the values of support = 22% and confidence = 70%.

Transaction ID	Items purchased
1	I_1, I_2, I_5
2.	I_2, I_4
3.	I_2, I_3
4.	I_1, I_2, I_4
5.	I_1, I_3
6.	I_2, I_3
7.	I_1, I_3
8.	I_1, I_2, I_3, I_5
9.	I_1, I_2, I_3

Sol.

Step 1 1 ~~Freqn~~ candidate item sets $(K=1)$ or (C_1)

Item	Frequency	Relative Support %
I_1	6	6/9 = 66.67%
I_2	7	7/9 = 77.78%.
I_3	6	6/9 = 66.67%.
I_4	2	2/9 = 22.23%.
I_5	2	2/9 = 22.23%.

} all the items have support $\geq$ 22%.

Step 2 Generate pairs of item sets
2 candidate item sets $(k=2)$ or C_2

Itemset pairs	Freq	Relative support
$\{I_1, I_2\}$	4	44.45% ✓
$\{I_1, I_3\}$	4	44.45 ✓
✗ $\{I_1, I_4\}$	1	11.11 %.
$\{I_1, I_5\}$	2	22.23 %. ✓
$\{I_2, I_3\}$	4	44.45%. ✓
$\{I_2, I_4\}$	2	22.23%. ✓
$\{I_2, I_5\}$	2	22.23 %. ✓
✗ $\{I_3, I_4\}$	0	0%.
✗ $\{I_3, I_5\}$	1	11.11 %.
✗ $\{I_4, I_5\}$	0	0 %.

Step 3 8 candidate item sets ($k=3$) or C_3

Itemset	frequency	Relative support
$\{I_1, I_2, I_3\}$	2	22.23%
$\{I_1, I_2, I_5\}$	2	22.23%

All items have support $\geq$ 22%.

Step 4 4 candidate item sets ($k=4$) or C_3

Item set	frequency	Relative support
~~$\{I_1, I_2, I_3, I_5\}$~~	1	11.11%

X $\{I_1, I_2, I_3, I_5\}$ → in C_2 its subset $\{I_3, I_5\}$ got eliminated as it cut off don't clear

∴ Algo terminates
freq. item sets = $\{I_1, I_2, I_3\}$, $\{I_1, I_2, I_5\}$

Step 5 Generate the rules based on the frequent item sets obtained:

(i) Rules generated by $\{I_1, I_2, I_5\}$

$(I_1, I_2) \rightarrow I_5 = 22.23/44.45 = 50\%$ ✓
$(I_1, I_5) \rightarrow I_2 = 22.23/22.23 = 100\%$ ✓
$(I_2, I_5) \rightarrow I_1 = 22.23/22.23 = 100\%$ ✓
$I_1 \rightarrow (I_2, I_5) = 22.23/66.67 = 33.33\% = 33\%$ ✗
$I_2 \rightarrow (I_1, I_5) = 22.23/77.78 = 28.57 = 29\%$ ✗
$I_5 \rightarrow (I_1, I_2) = 22.23/22.23 = 100\%$ ✓

rules generated:
$(I_1, I_5) \rightarrow I_2$
$(I_2, I_5) \rightarrow I_1$
$I_5 \rightarrow (I_1, I_2)$

(ii) rules generated by $\{I_1, I_2, I_3\}$

$I_1 \rightarrow (I_2, I_3) = 2/6 = 33\%$
$I_2 \rightarrow (I_1, I_3) = 2/7 = 29\%$
$I_3 \rightarrow (I_1, I_2) = 2/6 = 33\%$
$(I_1, I_2) \rightarrow I_3 = 2/4 = 50\%$
$(I_1, I_3) \rightarrow I_2 = 2/4 = 50\%$
$(I_2, I_3) \rightarrow I_1 = 2/4 = 50\%$

X

for $\{I_1, I_2, I_3\}$ no rules are getting generated

∴

FP-Growth Algorithm

Q. Can we design any method that mines the complete set of freq item sets without candidate generation ?

Sol: • An interesting method in this attempt is called FP-Growth (Frequent-Pattern Growth)

• It adapts a divide & conquer strategy as follows:

(i) Firstly, it compresses the database representing frequent items into a frequent pattern tree which retains the itemset association information.

(ii) It then divides the compressed database into a set of conditional databases (projected database), each associated with one freq. item and mines each search database separately.

Q1. Generate FP-Tree for the following transaction dataset whose minimum support count = 3. Show the conditional pattern base, conditional fp-tree and frequent item set.

Transaction ID	Items
T_1	{E, K, M, N, O, Y}
T_2	{D, E, K, N, O, Y}
T_3	{A, E, K, M}
T_4	{C, K, M, U, Y}
T_5	{C, E, I, K, O}

Step 1

The first step is to scan the database to find the occurances of the item sets in the database. This step is same as the first step of Apriori Algo.

Item	Frequency
A	1 X
C	2 X
D	1 X
E	4
I	1 X
K	5
M	3
N	2 X
O	3
U	1 X
Y	3

Step 2

A frequent pattern set (L) is built which will contain all the elements whose frequency is greater than or equal to the minimum support. These elements are stored in descending order of their respective frequencies.

$L = \{K:5,\ E:4,\ M:3,\ O:3,\ Y:3\}$

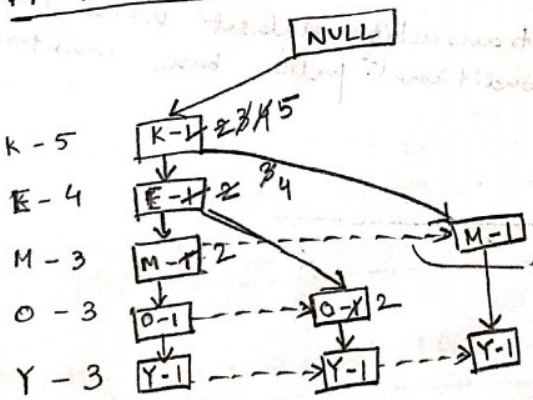
appear in data set first

Step 3
Now for each transaction, the respective ordered item set is built:

Transaction ID	Ordered item set (Descending order)
T_1	{K, E, M, O, Y}
T_2	{K, E, O, Y}
T_3	{K, E, M}
T_4	{K, M, Y}
T_5	{K, E, O}

Step 4:
Now all the ordered item sets are inserted into a Trie Data structure. To facilitate tree traversal, an item header table is built so that, each item points to its occurances in the tree via a chain of node links. In this way the problem of mining frequent patterns in db is transformed to that of mining the fp-tree.

FP-Tree Construction



Step 5:
Next step is to mine the created FP-tree. Now for each item, the conditional pattern base is computed which is path labels of all the paths which leads to any node of the given item in the FP tree.

Items	Conditional pattern base
Y	{K, E, M, O:1}, {K, E, O:1}, {K, M:1} ← last value of Y
O	{K, E, M:1}, {K, E:2}
M	{K, E:2}, {K:1}
E	{K:4}
K	—

Step 6
Now for each item, the conditional FP tree is built.

Items	Conditional FP Tree
Y	{K:3} ← check frequency & support count & find the items which satisfy it
O	{K, E:3}
M	{K:3}
E	{K:4}
K	—

Step 7
From the conditional FP Tree, the freq. pattern rules are generated by pairing the items of the conditional freq. pattern tree set to the corresponding item as shown below:

Items	Frequent pattern generated
Y	{<K, Y:3>}
O	{<K,O:3>, <E,O:3>, <K, E, O:3>}
M	{<K, M:3>}
E	{<K,E:4>}
K	—

Q. Generate FP Tree for the following transaction dataset whose minimum support count is 2. Show the conditional pattern base, conditional FP tree & freq. item set.

TID	List of items
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3

Sol.

S1

Item	Frequency
I1	6
I2	7
I3	6
I4	2
I5	2

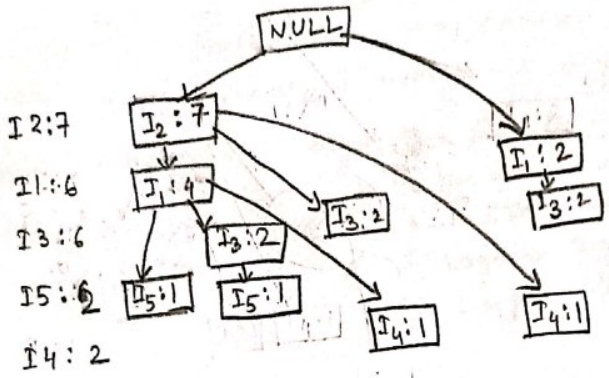
S2

L = {I2:7, I1:6, I3:6, I5:2, I4:2}

S3

Transaction id	Ordered item set
T100	$\{I_2, I_1, I_5\}$
T200	$\{I_2, I_4\}$
T300	$\{I_2, I_3\}$
T400	I_2, I_1, I_4
T500	I_1, I_3
T600	I_2, I_3
T700	I_1, I_3
T800	I_2, I_1, I_3, I_5
T900	I_2, I_1, I_3

S4



S5

Items	conditional pattern base
I_4	$\{I_2, I_1:1\}, \{I_2:1\}$
I_5	$\{I_2, I_1:1\}, \{I_2, I_1, I_3:1\}$
I_3	$\{I_2:2\}, \{I_1:2\}, \{I_2, I_1:2\}$
I_1	$\{I_2:4\}$ — R sub tree
I_2	—

S6

Items	conditional fp Tree
I_4	$\{I_2:2\}$
I_5	$\{I_2:2\}, \{I_1:2\}$
I_3	$\{I_2:4, I_1:2\}, \{I_1:2\}$ (treat subtree diff)
I_1	$\{I_2:4\}$
I_2	—

left subtree — right subtree

S7

Items	Frequent pattern generated
I_4	$\{\langle I_2, I_4:2\rangle\}$
I_5	$\{\langle I_2, I_5:2\rangle, \langle I_1, I_5:2\rangle, \langle I_2, I_1, I_5:2\rangle\}$
I_3	$\{\langle I_2, I_3:4\rangle, \langle I_1, I_3:4\rangle, \langle I_1, I_2, I_3:2\rangle\}$
I_1	$\{\langle I_1, I_2:4\rangle\}$
I_2	—

min of $\{I_2:4, I_1:2\}$ & $\{I_1:2\}$

Adv:-

- Algo needs to scan the database twice when compared to Apriori which scans the ~~transf~~ transactions for each iterations.
- No candidate item generation, hence algo takes less amount of time.
- Requires less memory in comparison to Apriori (for large db specially)
- Efficient & scalable for mining both long & short freq. patterns

Disadv:-

- building of FP tree is more complex than Apriori
- This algo may be expensive for small databases

CLASSIFICATION AND PREDICTION (Module-2)

- classification & Prediction are two forms of data analysis that can be used to extract models that describe important data class or predict future data trends. Such analysis can help to provide us with a better understanding of data at large.
- classification predicts categorical (discrete) labels, prediction models continuous valued func.

Eg: We can build a classification model to categorise bank loan applications as either safe / risky, & a prediction model predicts the expenditure of customers given their income & occupation.

Popular classification algo: ① K - nearest Neighbours (KNN)
② Decision Tree
③ Naive Bayes classifier
④ Logistic Regression
⑤ Support Vector Machine

Supervised & Unsupervised Learning

Supervised Learning

① These algo are trained using labelled data
② The goal is to train the model so that it can predict the o/p when it is given new data.
③ It can be used for those cases where we know i/p & o/p.
④ It can be categorised in classification & Regression problems.
⑤ Computational complexity less complex than unsupervised learning
⑥ These models produce more accurate result generally
⑦ Linear Regression, Logistic regression, Bayesian logic, etc.

Unsupervised Learning

① These algo are trained using unlabelled data.
② The goal is to find the hidden patterns and useful insights from the unknown dataset.
③ It can be used for those cases where we have i/p data only.
④ Classified in clustering & association problems.
⑤ Higher computational complexity
⑥ These models give less accurate result as compared to supervised learning.
⑦ Clustering, Apriori Algo, etc.

Cluster Analysis

- It is the process of grouping the data into classes / clusters so that objects within a cluster have high similarity in comparison to one another but are very dissimilar to obj in other clusters.
- Cluster analysis have been widely used in numerous applications like market research, pattern recognition, image processing, etc.
- It is also called data segmentation bcoz clustering partitions large data sets into groups acc. to their similarity.
- Clustering can also be used for outlier detection where outliers (values that are far away from any cluster) may be more interesting than common ones

- It is an example of unsupervised learning.

[learning by examples – supervised
" " observation – unsupervised]

Categorisation of major clustering methods

* (i) Partitioning methods :

Given a database of n objects, a partitioning method constructs k-partitions of the data, where each partition represents a cluster and $k \leq n$.

It classifies the data into k-groups, which satisfies the following requirements :

1. Each group must contain atleast one object.
2. Each object must belong to exactly one group.

- Centroid Based Technique – K Means –

1. K-means algo takes the i/p parameter k and partitions a set of n-objects into k-clusters so that the resulting intra cluster similarity is high for the inter-cluster similarity is low.
2. cluster similarity is measured in regard to the mean value of the objects in a cluster which is referred to as cluster's centroid.

K-Means clustering algorithm

Input : k → no. of clusters
D → dataset containing n objects $[x_1, x_2, \ldots x_n]$

Output : A set of k clusters.

Step 1 – choose k random points as initial cluster centers called centroids.

Step 2 – Assign each x_i to the closest cluster by implementing its distance to each centroid.

Step 3 – For each cluster, new centroids are computed by taking the avg of the assigned points. (updating the cluster means)

Step 4 – Keep repeating step 2 & 3 until convergence is achieved.

* Note :– K-means clustering stops creating/optimizing clusters after finding no new reassignment of data points and also after the algo reaches the defined no. of iterations.

* Q1. Apply the K-means algo for the following dataset for 2 clusters. Consider data points 1 and 2 as the initial centroid of the respective clusters. Find out the final cluster centroids. Continue the procedure for 2 iterations. Apply Euclidean dist. for as distance function.

{ 10-15 marks → max^m Iteration 3
otherwise → " " 2

Sample no.	X	Y
1	185	72
2	170	56
3	168	60
4	179	68
5	182	72
6	188	77

Iteration – I

Point	Centroid 1 (185, 72)	cluster 2 (170, 56)	cluster assignment
(185, 72)	0	21.93	c_1
(170, 56)	21.93	0	c_2
(168, 60)	20.81	4.47	c_2
(179, 68)	7.21	15	c_1
(182, 72)	3	20	c_1
(188, 77)	5.83	27.65	c_1

New centroid

$c1: \left(\frac{185+179+182+188}{4}, \frac{72+68+72+77}{4}\right)$

$= (183.5, 72.25)$

$c2: \left(\frac{170+168}{2}, \frac{56+60}{2}\right)$

$= (169, 58)$

Iteration – II

After the 2nd iteration, the assignment of points has not changed & hence the algorithm is stopped & the points are clustered.

final centroid value : $c_1 = 1, 4, 5, 6$

$c_2 = 2, 3$

	P_1	P_2	P_3	P_4	P_5
P_1	0				
P_2	9	0			
P_3	3	7	0		
P_4	6	5	9	0	
P_5	11	10	(2) min	8	0

Pair (P_3, P_5) → dist = 2

	P_1	P_2	$[P_3, P_5]$	P_4
P_1	0			
P_2	9	0		
$[P_3, P_5]$	11	10	0	
P_4	6	(5) min	9	0

Pair P_2, P_4 dist = 5

	P_1	$[P_2, P_4]$	$[P_3, P_5]$
P_1	0		
$[P_2, P_4]$	(9) min	0	
$[P_3, P_5]$	11	10	0

Pair $(P_1, (P_2, P_4))$ ← dist = 9

	$[P_1, P_2, P_4]$, $[P_3, P_5]$

$d((P_3, P_5), (P_1, P_2, P_4))$

$= \max(d((P_3, P_5), P_2), d((P_3, P_5), P_4))$

$= \max(10, 9)$

$= 9$

Average Linkage

The hierarchical method that involves looking at the distances b/w all pairs and avg. all of these distances.

Q. For the given set of points, identify clusters using avg. linkage agglomerative clustering. Round of the dist. matrix to two places of decimal.

	A	B
P_1.	1	1
P_2.	1.5	1.5
P_3.	5	5
P_4.	3	4
P_5.	4	4
P_6.	3	3.5

	P_1	P_2	P_3	P_4	P_5	P_6
P_1	0					
P_2	0.71	0				
P_3	5.66	4.95	0			
P_4	3.60	2.92	2.24	0		
P_5	4.24	3.53	1.41	1.0	0	
P_6	3.20	2.5	2.5	0.5 (min)	1.12	0

Pair (P_4, P_6) dist = 0.5

	P_1	P_2	P_3	$[P_4 P_6]$	P_5
P_1	0 (min)				
P_2	0.71	0			
P_3	5.66	4.95	0		
$[P_4 P_6]$	3.40	2.71	2.37	0	
P_5	4.24	3.53	1.41	1.06	0

Pair $(1, P_2)$ dist = 0.71

	$[P_1, P_2]$	P_3	$[P_4 P_6]$	P_5
$[P_1, P_2]$	0			
P_3	5.31	0		
$[P_4 P_6]$	3.05	2.37	0	
P_5	3.89	1.41	1.06 (min)	0

Pair $(P_5, (P_4, P_6))$ dist = 1.06

	$[P_1, P_2]$	P_3	$[P_4 P_5 P_6]$
$[P_1, P_2]$	0		
P_3	5.31	0	
$[P_4 P_5 P_6]$	3.47	1.89 (min)	0

Pair $(P_3, (P_4 P_5 P_6))$

	$[P_1, P_2]$	$[P_3 P_4 P_5 P_6]$
$[P_1, P_2]$	0	
$[P_3 P_4 P_5 P_6]$	4.39	0

4 Axioms of Distance Metrics

(i) $d(i,j) \geq 0$ (Distance is a non-negative no.)

(ii) $d(i,i) = 0$ (distance of an obj to itself is always 0)

(iii) $d(i,j) = d(j,i)$ (distance is a symmetric func.)

(iv) $d(i,j) \leq d(i,h) + d(h,i)$ (Triangular inequality)
[For 3 elements in this set, the sum of the dist for any two pairs must be greater than the dist. for the remaining part]

** weighted Euclidean dist.

$$d(i,j) = \sqrt{w_1 (x_{i_1} - x_{j_1})^2 + w_2 (x_{i_2} - x_{j_2})^2 + \cdots + w_n (x_{i_n} - x_{j_n})^2}$$

K-Medoids Clustering

(i) K-means algo is sensitive to outliers because an object with an extremely large value may substantially distort the distribution of data. Instead of taking the mean value of the obj. in the cluster as a reference point, we can pick actual objects to represent the clusters.

(ii) Each remaining object is clustered with the representative obj to which it is the most similar. The partitioning method is then performed based on the principle of minimising the sum of the dissimilarities b/w each object & its corresponding reference point.

$$E = \sum_{j=1}^{k} \sum_{p \in C_j} |p - o_j|$$

E → The sum of the absolute error for all objects in the dataset.

P → Point in space representing a given object in cluster C_j

O_j → Representative obj. of C_j (medoid)

Algo (PAM - Partitioning Around Medoids)

I/p: k → the no. of clusters

D → the dataset containing 'n' objects

O/p: A set of 'k' clusters.

Step 1: Initially select k-random points as the mediods from the given 'n' data points of the dataset.

Step 2: Associate each data point to the closest mediod.

Step 3: While the cost of the configuration decreases:

for each medoid 'm' and for each non-medoid datapoint 'o'

(i) swap 'm' and 'o', recompute the cost (sum of dist. of points to their medoid)

(ii) If the total cost of the configuration increase than the previous step, undo the swap.

Q. QB(42)

Step -1 Select 2 medoids : $c_1 = (3,4)$, $c_2 = (7,4)$

(max 3 iteration min 2 iteration)

i	X	Y	c_1	c_2	cluster assignment
X_1	2	6	3	7	c_1
X_2	3	4	0	4	c_1
X_3	3	8	4	8	c_1
X_4	4	7	4	6	c_1
X_5	6	2	5	3	c_2
X_6	6	4	3	1	c_2
X_7	7	3	5	1	c_2
X_8	7	4	4	0	c_2
X_9	8	5	6	2	c_2
X_{10}	7	6	6	2	c_2

clusters are → $c_1 = \{(2,6), (3,4), (3,8), (4,7)\}$

$c_2 = \{(6,2), (6,4), (7,3), (7,4), (8,5), (7,6)\}$

Step 2

Total cost

$(3+4+4) + (3+1+1+2+2) = 20$

dist b/w (3,4) & (2,6) | dist b/w (3,4) & (3,8)

Step 3

Next step is the selection of non medoid O' randomly. Let us assume $O' = (7,3)$

$\therefore c_1 = (3,4)$

$O' = (7,3)$

i	X	Y	c_1	O'	
X_1	2	6	3	8	c_1
X_2	3	4	0	5	c_1
X_3	3	8	4	7	c_1
X_4	4	7	4	7	c_1
X_5	6	2	5	2	c_2
X_6	6	4	3	2	c_2
X_7	7	3	5	0	c_2
X_8	7	4	4	1	c_2
X_9	8	5	6	3	c_2
X_{10}	7	6	6	3	c_2

Total cost

$= (3+4+4) + (2+2+1+3+3)$

$= 22$

Step 4

Cost of swapping medoid c_2 with O'

S = current total cost − previous total cost

$= 22 - 20 > 2 > 0$

∴ Moving to O' would not be suitable one & the prev one was better

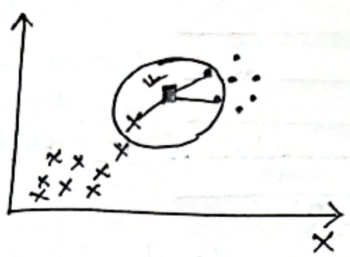
So final medoid = $(3,4)$ $(7,4)$

K-NN Technique

(K- Nearest Neighbour)

[supervised algorithm]

- K-NN is a supervised classification algorithm that classifies the new data points based on the nearest data point.



k=3 (always integer)

{ for regression – use mean
for classification – use mode

- K-NN algo is used to solve classification & regression problems. However it is mainly used for classification problem.
- K-NN is also called a lazy learning algo (lazy learner) because it doesn't perform any training when you supply the training data. Instead it just stores the data during training time and doesn't perform any calculations.
- K-NN is also called non-parametric method because it doesn't make any assumption about the underlying data distributions.

Algo

Step 1 – Load the dataset

Step 2 – choose the value of K (nearest data points)

Step 3 – for each point in the test dataset, do the following –

3.1 – Calculate the dist b/w test data and each row of training data with the help of any dist. metrics.

3.2 – Now based on the dist. value sort them in ascending order

3.3 – Next, it will choose the top K entries from the sorted list

3.4 – Label the test point based on the majority of classes present in the selected points.

Step 4 – Stop.

Q. Perform the K-NN classification algo on the following dataset and predict the class for X ($P_1 = 3$, & $P_2 = 7$)
Given K = 3

	P_1	P_2	class
(i)	7	7	F
(ii)	7	4	F
(iii)	3	4	T
(iv)	1	4	T

Sol: $D(X,i) = \sqrt{(3-7)^2 + (7-7)^2} = 4 \rightarrow (N_3)$ False

$D(X,ii) = \sqrt{(3-7)^2 + (4-7)^2} = 5$

$D(X,iii) = \sqrt{(3-3)^2 + (4-7)^2} = 3 \rightarrow (N_1)$ True

$D(X,iv) = \sqrt{(3-1)^2 + (7-4)^2} = \sqrt{4+9} = \sqrt{13} = 3.61 \rightarrow (N_2)$ True

2 True > 1 False

$\therefore$ X ($P_1 = 3$, $P_2 = 7$) belong to True class.

[Note:
(i) when dealing with 2 class problem, its better to choose an odd value of K.
(iii) Otherwise a scenario can arrise where the no. of neighbours in each class is the same.
(ii) choosing a very small value of k leads to unstable decision boundeds. For eg.- Suppose k=1 or 2 can be subjected to noise and effects of outliers]

Disadv:-
(i) associated computation cost is high as it stores all the training data
(ii) prediction may become slower if the value of N is high.
(iii) sensitive to irrelevant features.

Decision Tree Classification

- It is a supervised learning algorithm that can be used for both classification & regression. But mostly it is preferred for solving classification problems.
- It is a tree structured classifier where each internal node tests an attribute, each branch corresponds to attribute values and each leaf node assigns a classification.
- It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.

Attribute Selection Measure (ASM)

While implementing a decision tree the main issue arises that how to select the best attribute of the root node.
With the help of ASM technique, we can easily select the best attribute for the nodes of the tree.
Two popular technique :- (i) Entropy & Information Gain
(ii) Gini index.

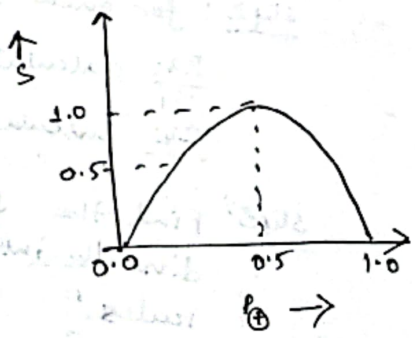
(i) Entropy & Informational Gain

Entropy- It is a metric to measure the impurity in a given attribute. Its specifies randomness in data.

$$\text{Entropy}(S) = \sum_{i=1}^{c} -p_i \log_2(p_i)$$

$$= -p_{\oplus} \log_2(p_{\oplus}) - p_{\ominus} \log_2(p_{\ominus})$$

$p_{\oplus}$ → proportion of the examples in S.



- Entropy value 0 represents data sample is pure/homogeneous
- Entropy value 1 represents that the data sample has a 50-50 split belonging to 2 categories.

GPA	Studied	Passed ← target attribute
Low	F	F
Low	T	T
Medium	F	F
Medium	T	T
High	F	T
High	T	T

Q. what is the ~~Ent~~ Entropy of the data set.

Passed = { F : 2, T : 4 } → 4+, 2−

$P_{\oplus} = \frac{4}{6}$

$P_{\ominus} = \frac{2}{6}$

$$\text{Entropy} = -P_{\oplus}\log_2 P_{\oplus} - P_{\ominus}\log_2 P_{\ominus}$$

$$= -\frac{2}{3}\log_2\left(\frac{2}{3}\right) - \frac{1}{3}\log_2\left(\frac{1}{3}\right)$$

$$= -\frac{2}{3}\log_2\left(\frac{2}{3}\right) - \frac{1}{3}\log_2\left(\frac{1}{3}\right)$$

$$= 0.918 \simeq 0.92$$

Information Gain (IG) - It is the measurement of changes in entropy after the segmentation of a data set based on an attribute. It calculates how much information a feature provides us about a class. According to the value of IG we split the node and build the decision tree. A decision tree algo always tries to maximise the value of IG and a node or attribute having the highest IG is splitted first.

$$\text{Gain}(S, A) = \text{Entropy}(S) - \sum_{v \in \text{values}(A)} \frac{|S_v|}{|S|} \text{Entropy}(S_v)$$

Q. For the following weather attributes create decision tree. Sample Dataset is given where the target attribute is "Play Tennis". (*) →

Formation of Decision Tree (steps in ID3)

Iterative Dicrotomiser

Step 1 : calculate the entropy of the entire DS.

Step 2 : for each attribute/feature :

2a: calculate entropy for all of its categorical values

2b: Calculate information ~~of~~ gain for the feature.

Step 3: Find the feature with max^m information gain. Database is divided into smaller subsets acc. to categories of decision rules.

Step 4: Repeat it until we get the desired tree.

(*) →

Sol:

Entropy of entire dataset (for target attribute) → 9 Yes, 5 No

$= [9+, 5-]$

$$\text{entropy} = -\frac{9}{14}\log_2\left(\frac{9}{14}\right) - \frac{5}{14}\log_2\left(\frac{5}{14}\right)$$

$$= 0.9403$$

Outlook → Sunny (5) [2+, 3-], Overcast (4) [4+, 0-], Rain (5) [3+, 2-]

$$S(\text{Outlook}, \text{sunny}) = -\frac{2}{5}\log_2\left(\frac{2}{5}\right) - \frac{3}{5}\log_2\left(\frac{3}{5}\right) = 0.971$$

$$S(\text{Outlook}, \text{overcast}) = 0$$

$$S(\text{Outlook}, \text{rain}) = -\frac{3}{5}\log_2\left(\frac{3}{5}\right) - \frac{2}{5}\log_2\left(\frac{2}{5}\right) = 0.971$$

$$\text{Information Gain(Outlook)} = 0.940 - \left(\frac{5}{14} \times 0.971 + \frac{4}{14} \times 0 + \frac{5}{14} \times 0.971\right)$$

$$= 0.940 - 0.694$$

$$= \boxed{0.246} \rightarrow \text{max}^{\text{m}} \rightarrow \text{root}$$

Temperature → Hot (4) [2+, 2-], Cool (4) [3+, 1-], Mild (6) [4+, 2-]

$S(\text{Temp}, \text{hot}) = 1.0$

$S(\text{Temp}, \text{cool}) = 0.8113$

$S(\text{Temp}, \text{mild}) = 0.9183$

$$IG(\text{Temp}) = 0.94 - \left(\frac{4}{14} \times 1 + \frac{4}{14} \times 0.8113 + \frac{6}{14} \times 0.9183\right)$$

$$= 0.0289$$

Humidity → High (7) [3+, 4-], Normal (7) [6+, 1-]

$S(\text{Humidity}, \text{high}) = 0.9852$

$S(\text{"}, \text{normal}) = 0.5916$

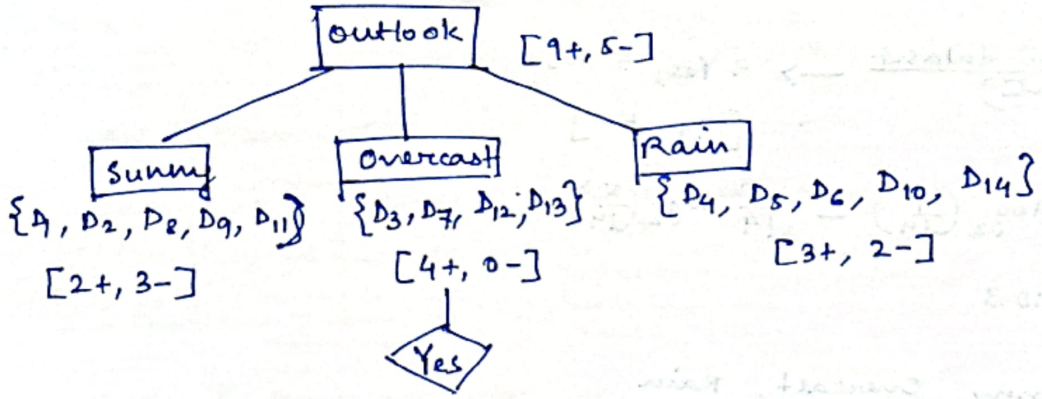
$IG(\text{humidity}) = 0.1516$

Wind → Strong (6) [3+, 3-], Weak (8) [6+, 2-]

$S(\text{wind}, \text{strong}) = 1$

$S(\text{wind}, \text{weak}) = 0.8113$

$IG(\text{wind}) = 0.0478$



Partial Decision Tree

for Sunny

S(Sunny) [2+, 3-] = 0.971

temperature

→ S_{HOT} → [0+, 2-] = 0

→ S_{mild} → [1+, 1-] = 1

→ S_{cold} → [1+, 0-] = 0

$$IG(Sunny, temperature) = 0.971 - \left(\frac{2}{5}\times 0 + \frac{2}{5}\times 1 + \frac{1}{5}\times 0\right)$$

$$= 0.571$$

humidity

→ S_{high} → [0+, 3-] = 0

→ S_{normal} → [2+, 0-] = 0

$$IG(sunny, humidity) = 0.971 - \left(\frac{3}{5}\times 0 + \frac{2}{5}\times 0\right)$$

$$= \boxed{0.971} \rightarrow max^m$$

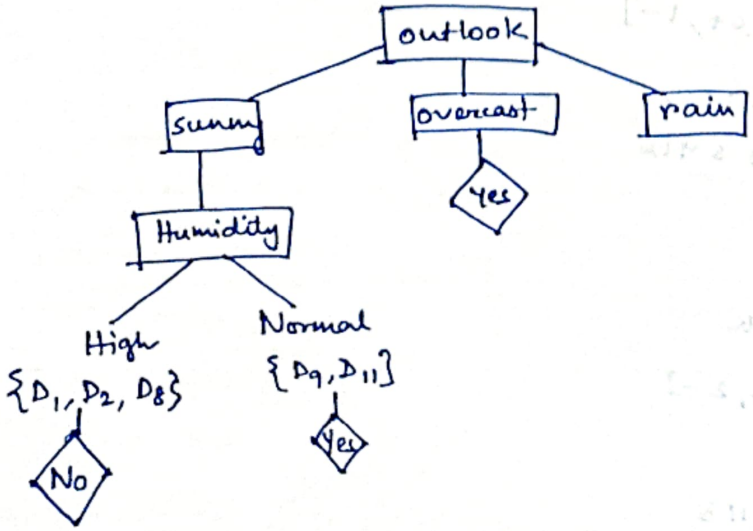
wind

→ S_{weak} → [1+, 2-] ≤ 0.918

→ S_{strong} → [1+, 1-] = 1.0

$$IG(sunny, wind) = 0.971 - \left(\frac{3}{5}\times 0.918 + \frac{2}{5}\times 1.0\right)$$

$$= 0.0192$$



Partial Decision Tree

for rain

$S(Rain) = [3+, 2-] = 0.971$

temperature

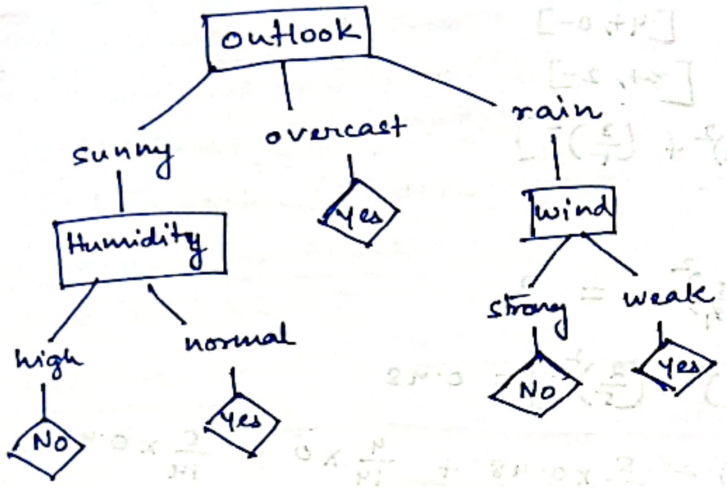
- S_{hot} $[0+, 0-] = 0$
- S_{mild} $[2+, 1-] = 0.9183$
- S_{cold} $[1+, 1-] = 1.0$

$IG(rain, temp) = 0.0192$

wind

- S_{weak}
- S_{strong}

$IG(rain, wind) = 0.971$



Decision Tree

Gini Index / Gini Impurity / Gini coefficient

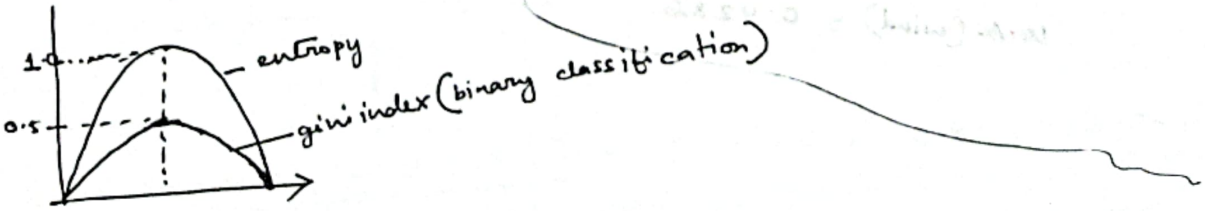
The another way of splitting the decision tree is via the Gini index. The entropy & IG method focusses on purity & impurity in a node. The Gini index measures the probability for a random instance being misclassified when chosen randomly. The lower the gini index, the lower is the ~~bottom~~ ~~lower~~ likelihood of misclassification.

$$G_I = 1 - \sum_{i=1}^{c} p_i^2$$

where p_i → proportion of data points belonging to class 'i' in the dataset.

(i) For the context of binary classes, G_I has maximum value of 0.5 & min value of 0

(ii) For multiple classes, G_I has value from 0 to 1.



Q. Decision Tree dataset

Gini index of entire dataset:

9 instances → YES, 5 instances → No

$$Gi(S) = 1 - \left[\left(\frac{9}{14}\right)^2 + \left(\frac{5}{14}\right)^2\right]$$

$$\approx 0.45919$$

Computation of Gini index

Outlook
- → sunny (5 instances) [2+, 3-]
- → overcast (4 ") [4+, 0-]
- → rain (5 ") [3+, 2-]

$$Gini(S_{sunny}) \rightarrow 1 - \left[\left(\frac{2}{5}\right)^2 + \left(\frac{3}{5}\right)^2\right]$$

$$= 0.48$$

$$Gini(S_{overcast}) \rightarrow 1 - \left(\frac{4}{4}\right)^2 = 0$$

$$Gini(S_{rain}) \rightarrow 1 - \left[\left(\frac{3}{5}\right)^2 + \left(\frac{2}{5}\right)^2\right] = 0.48$$

$$\therefore \text{weighted average (Outlook)} = \frac{5}{14} \times 0.48 + \frac{4}{14} \times 0 + \frac{5}{14} \times 0.48$$

$$= \boxed{0.34285} \text{ min.} \rightarrow \text{root}$$

Temp
- → hot
- → cool
- → mind

weighted avg (temp) = 0.4405

Humidity
- → high
- → normal

W.A. (humidity) = 0.3673

Wind
- → strong
- → weak

W.A. (wind) = 0.4286

MODULE-3

TIME SERIES ANALYSIS

- A series of observations recorded in accordance with the time of occurance is called series. The values of a variable are observed chronologically in terms of days, week, months, or year.
- The objects of time series analysis are to isolate and measure the effects of various components. Such analysis will help in understanding the past behaviour so that we may be able to predict the future tendencies.

- 4 components of time series
  (i) Secular Trend / Trend (T)
  (ii) Seasonal variations (S)
  (iii) Cyclical Fluctuations (C)
  (iv) Irregular / random movement (I)

- multiplicative model
  $y_i = S \times T \times C \times I$

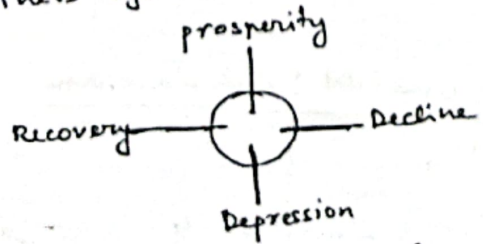
  additive model
  $y_i = S + T + C + I$

VARIOUS COMPONENTS OF TIME SERIES

(a) Secular Trend (T) → It is the smooth, regular and long term movement exhibitting the growth / decline over a period of time.
Eg- population growth, death rate

(b) Seasonal Variation (S) → It represents a type of periodic movement where the period is not longer than 1 year. This up and down movement of timeseries recurring with remarkable regularity year after years is attributable to the presence of seasonal variations.

(c) Cyclical Fluctuations (C) → It is another type of periodic movement where the period is more than a year. Such movements are regular & oscillatory in nature.
Eg- business cycle.



(d) Irregular / random movement (I) → This variations ~~irregular~~ are caused by factors of an irratic nature. This variations don't follow a particular model, & so they are not predictable.

- increasing demand of automobiles — secular trend.
- passenger traffic during 24 hours a day — seasonal fluctuations
- recession — cyclic fluctuation
- a fire in a factory delaying production for a week — irregular movement

MEASUREMENT OF TREND

There are four methods of isolating secular trend in time series.

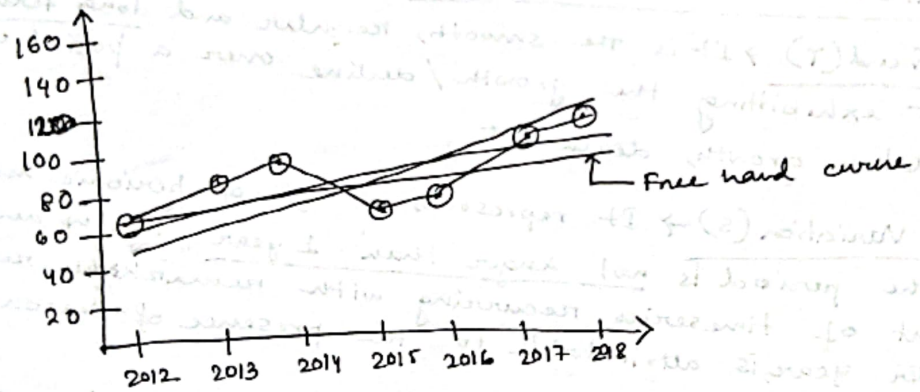
1. Free hand method.
2. Semi-average method
3. moving-average "
4. fitting mathematic curves.

1. Free hand method –

(i) The given data are plotted as points on a graph paper against time. The time-series data are shown along the vertical axis & time (t) along the horizontal axis.

(ii) A smooth free-hand curve is drawn through the scatter of the plotted points. The dist. of this line is known as trend line which gives the trend value for each time period.

Time	2012	2013	2014	2015	2016	2017	2018
values	64	82	97	71	78	112	115



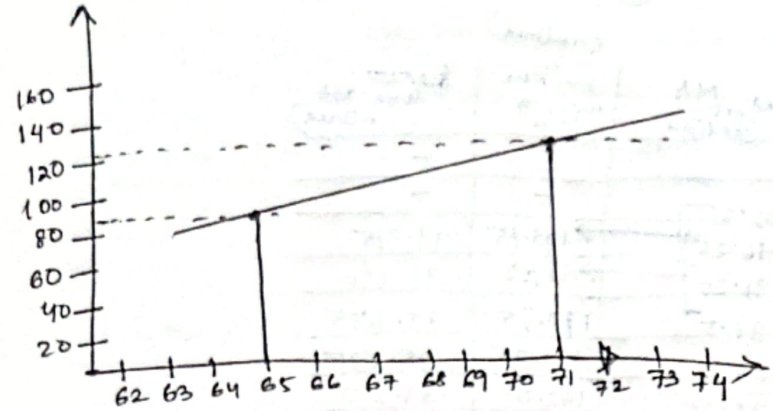
2. Semi-Average method –

The method consist in dividing the data into 2 parts and then finding an average of each part. These averages are plotted as points on a graph paper against the meet point of the time interval covered by each path. The st. line joining these 2 points gives the trend line.

year	time series	7 year semi total	semi-averages
1962	64		
63	82		
64	97		
65	71	middle → 619	88.4
66	78		88.4 + 5.5 = 93.9
67	112		93.9 + 5.5 = 99.4
68	115		⋮
69	131		
70	88	850	121.4
71	100		
72	146	74 | 120	
73	150		

n=13

odd – 1 year common



$$\frac{121.4 - 88.4}{6}$$

$$= \frac{33.0}{6}$$

$$= 5.5 \text{ slope.}$$

Method of selected points. It gets affected by outliers.

Eg -②

Even dataset

Year	production	4 year semi total	Semi avg
1971	40		
72	45	→ 167	→ 41.75
73	40		
74	42		
75	46		
76	52	→ 215	→ 53.75
77	56		
78	61		

n=8

n/2=5

3. Method of moving average

(i) A very convinient tool for ironing out fluctuations in time-series.

(ii) It tends to reduce the amount of variations in the data set by eliminating unwanted fluctuations (smoothing of time series)

(iii) MA of order 'n' = $\frac{y_1+y_2+y_3\cdots+y_n}{n}$, $\frac{y_2+y_3+\cdots y_{n+1}}{n}$, $\frac{y_3+y_4+\cdots y_{n+2}}{n}$ ....

Note: (i) If the period of moving avg is odd, the trend values correspond to the given time.

(ii) If the period is even, a two-point moving avg of the moving averages so obtained was to be found out for centering them.

Eg-① calculate the 5-year (odd) moving avg

Year	value	5 year moving total	5yr MA
2000	105	—	—
01	115	—	—
02	100	→ 490	98
03	90	→ 480	96
04	80	→ 450	90
05	95	425	85
06	85	395	79
07	75	—	—
08	60	—	—

Eg-(2)

Calculate the 4 year MA

Year	value	4yr Moving total (not centered)	4 year MA (centered)	2-item moving total (centered)	4 yr MA (centered)
2000	105			—	—
01	115	→ 410	102.5	→ 198.75	99.375
02	100	385	96.25	→ 187.5	93.750
03	90	365	91.25	178.75	89.375
04	80	350	87.5	171.25	85.625
05	95	335	83.75	162.5	81.250
06	85	315	78.75	—	—
07	75			—	—
08	60				

Merits

- The method is simple to apply & involves no difficult calculations.
- If the time series contains regular cyclical fluctuations, these fluctuations are automatically removed provided an appropriate period is choosen. Even when the fluctuations are not completely eliminated, MA process reduces the intensity
- This method is flexible in the sense that if some more observations are added to the original series, the entire calculations need not to be changed.

Demerit

- Trend values for all the given time periods can't be obtained. Some trend values at the beginning & at the end of the series have to be left out. Their no. increases with inc. in the period of MA.
- The method can't be used for forecasting future trends
- MA are useful only when the trend is linear. If the graph is curvi-linear, MA may deviate from the trend.

3(b) Weighted MA

Q. Find the trend of the following series using a 3 year weighted MA. with weight 1, 2, 1

year	values	3yr moving total	3 year MA
1	2	—	—
2	4	→ (2×1)+(4×2)+(5×1) = 15	3.75 ← 15/(1+2+1) = 15/4
3	5	→ (4+10+7) = 21	5.25
4	7	27	6.75
5	8	33	8.25
6	10	41	10.25
7	13	—	—

4) Method of fitting mathematical curves

In this method, an a

Method of least squares

Normal Eqns

[ONLINE CLASS NOTES IN LAPTOP]

UNIT-4

CLASS IMBALANCE PROBLEM

classification problem

1000 records

- 500 yes, 500 no — Balanced dataset
- 600 " , 400 "
- 700 " 300 " — imbalanced dataset
- 800 " 200 "

100 records of patient data

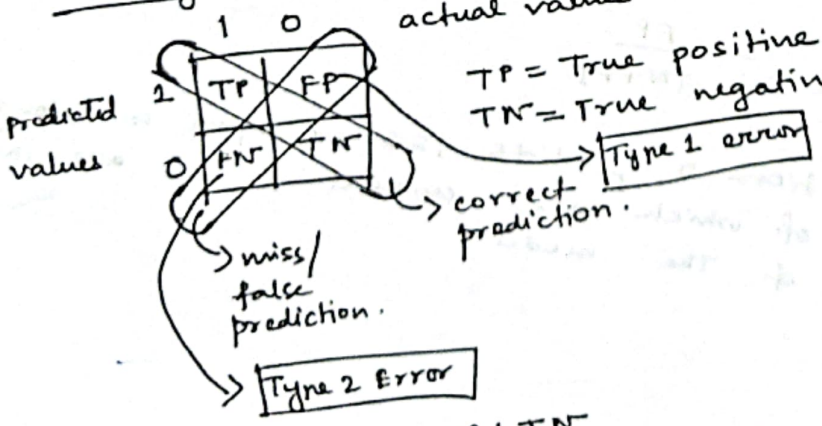
- 95% majority class (no disease)
- 5% minority class (disease)

* classifier biased towards no disease (majority class)

CONFUSION MATRIX

In ML, to measure the performance of the classification model we use the confusion matrix. It is a matrix that summarises the performance of ML model on a set of test data. It is a means of displaying the no. of accurate & inaccurate instances based on the model's predictions.

Binary classification model



$$\text{accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

Error rate = 1 − accuracy

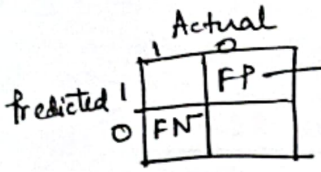
$$\text{Precision} = \frac{TP}{TP + FP}$$

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

or TPR (True Positive Rate)

or sensitivity

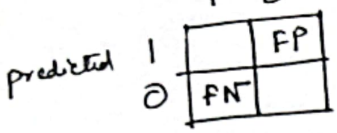
Mail Spam detection



→ importance to FP
importance to correct ~~mails~~ mails.

when FP is imp → Precision (reduce false positive)

Cancer ~~detecti~~ detection



→ medical application → importance to FN

when FN imp → use Recall.

when both FN & FP is imp → F- Beta score

$$= \frac{(1+\beta^2)\,\text{Precision} \times \text{Recall}}{\beta^2 \times (\text{Precison} + \text{Recall})}$$

if $\beta = 1 \Rightarrow$ F-1 score **

$$\therefore \frac{2 \times \text{Precison} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad \text{(Harmonic mean) HM}$$

more value better model.

- TPR (True +ve rate) $= \frac{TP}{TP+FN} = \frac{TP}{\text{Actual positive}}$
- FNR (False -ve rate) $= \frac{FN}{\text{actual +ve}} = \frac{FN}{TP+FN}$
- TNR (Tre -ve rate) $= \frac{TN}{\text{actual -ve}} = \frac{TN}{TN+FP}$
- FPR (False true rate) $= \frac{FP}{\text{actual -ve}} = \frac{FP}{TN+FP}$

Q. A model makes predictions & predicts 120 examples as belonging to minority class, 90 of which are correct & 30 are incorrect. what is the precision of the model.

Sol.

$$\text{precision} = \frac{90}{120} = 0.75$$

3
4

Q.

$n = 165$

	Predicted No	predicted Yes
Actual No	50	10
Actual Yes	5	100

Find accuracy, precision, recall.

Ans:-

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$= \frac{100 + 50}{50 + 10 + 5 + 100} = \frac{150}{165} = 0.909 \simeq 0.91$$

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

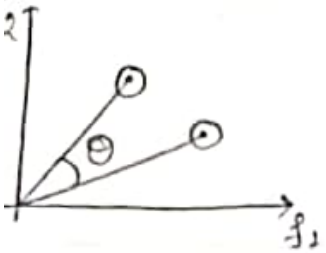
$$= \frac{100}{100 + 10} = 0.909 \simeq 0.91$$

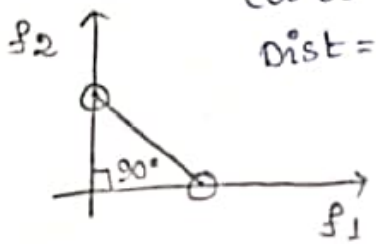
$$\text{Recall} = \frac{100}{100 + 5}$$

$$= 0.9523 \simeq 0.95.$$

Cosine similarity $\rightarrow \cos\theta$.

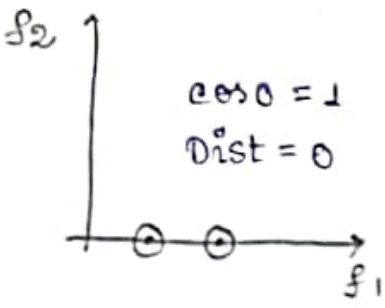
cosine distance $= 1 - \cos\theta$.





$\cos 90° = 0$ (similarity $= 0$)

Dist $= 1$.



$$\cos(x, y) = \frac{x \cdot y}{\|x\| * \|y\|}$$

- $x \cdot y$ → dot product of vector 'x' & 'y'
- $\|x\| * \|y\|$ → length of the vector x & y

Cosine similarity is a metric used to measure the similarity of two vectors. Specifically it measures the similarity in direction or orientation of vectors ignoring differences in their magnitudes. The similarity of two vectors is measured by the cosine of the angle b/w them.

Q. Find the cosine similarity b/w 2 vectors 'x' & 'y'

$x = \{3, 2, 0, 5\}$

$y = \{1, 0, 0, 0\}$

Ans:- $x \cdot y = 3\times1 + 2\times0 + 0\times0 + 5\times0 = 3$.

$\|x\| = \sqrt{3^2+2^2+0+5^2}$ $\quad\quad$ $\|y\| = \sqrt{1^2+0+0+0}$

$= \sqrt{13+25}$ $\quad\quad$ $= 1$.

$= \sqrt{38}$

$= 6.16$.

$$\cos(x,y) = \frac{3}{6.16\times1} = 0.487 \simeq 0.49.$$

Dissimmilarity of $\cos(x,y) = 1 - 0.49 = 0.51$.

Naive Bayes classifier.

Another name of mutual exhaustive set are called sample space.

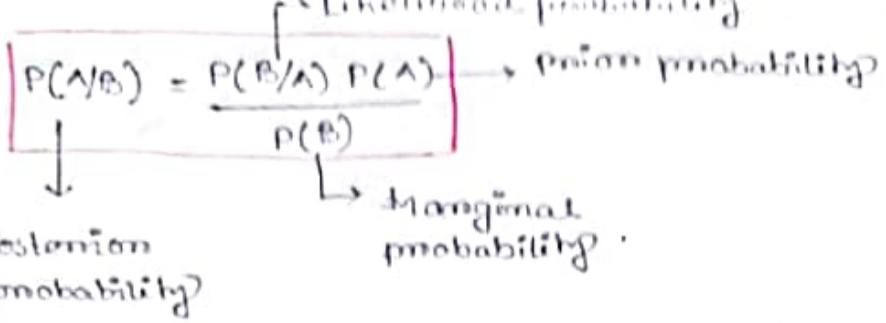
It is a supervised learning algorithm which is based on Bayes Theorem and used for solving classification problem

It is a probabilistic classifier which means it predicts on the basis of probabilistic of an object.

The algorithm is comprised of two words → Naive & Bayes

> It is called Naive because it assumes the occurence of certain feature is independent of occurence of other features.

> Bayes:- It follows the principal of Bayes theorem.



Q. Determine whether the car is stolen or not for the unseen sample X using Naive Bayes Method.

Ans Unseen Sample

X = red, sports, Domestic.

Target class Level: Stolen.

□ Prior prob :-

$P(yes) = 5/10 = 0.5$

$P(NO) = 5/10 = 0.5$

□ Likelihood for colour :-

$P(colour/yes)$ & $P(colour/NO)$

Type (colour) = Red & yellow.

$P(Red/yes) = \frac{P(Red \cap yes)}{P(yes)} = \frac{3}{5}$

$P(Red/NO) = \frac{P(Red \cap NO)}{P(NO)} = \frac{2}{5}$

$P(yellow/yes) = \frac{P(yellow \cap yes)}{P(yes)} = \frac{2}{5}$

$P(yellow/NO) = \frac{3}{5}$.

Likelihood for Type :-

$P(Type/NO)$ & $P(Type/yes)$.

$P(sports/NO) = 2/5$ $P(sports/yes) = 4/5$.

$P(suv/NO) = 3/5$ $P(suv/yes) = 1/5$.

□ Likelihood for origin.

P(origin/NO) & P(origin/yes).

$P(\text{domestic}/NO) = \frac{3}{5}$ $P(\text{domestic}/\text{yes}) = \frac{2}{5}$

$P(\text{imported}/NO) = \frac{2}{5}$ $P(\text{imported}/\text{yes}) = 3/5$.

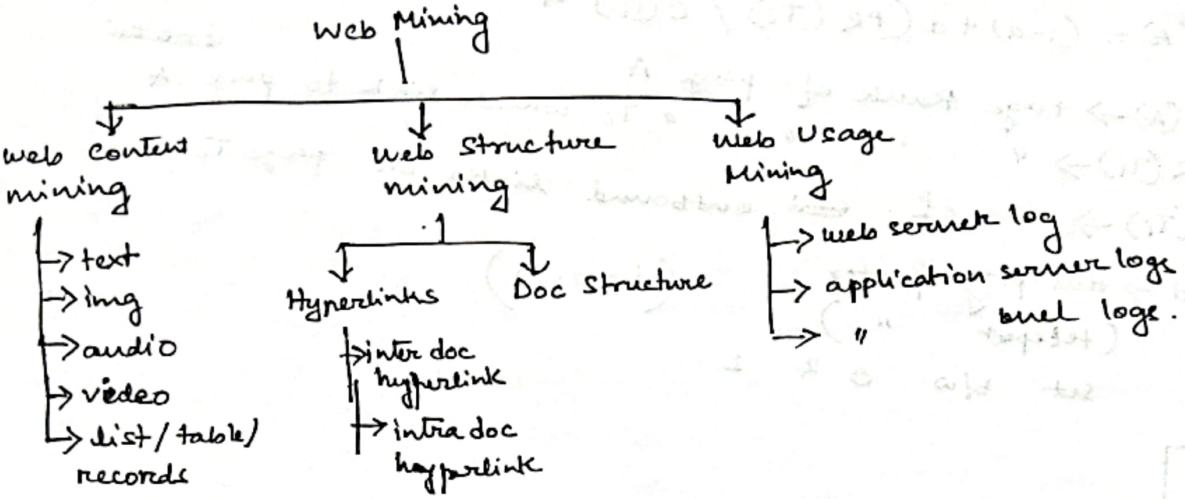
WEB MINING

{ Page rank algo
Hits algo

(i) It is the art & science of discovering patterns & insights from world wide web (www) so as to improve it.

(ii) It analyses data from the web and helps find insights that would optimize the web content & improve UX.

* challenges of mining WWW data. (Notes → classroom)



Web content mining — (i) It is the process of extracting useful info from the contents of web doc. The content comprises of text, img, audio, video or ~~structured~~ tably records, etc.

(ii) The text & application contents on the pages could be analysed for its usage by visit counts. It depends on the quality of content that attracts more users.

Web structure mining — (i) It is the process of discovering structured info (linked structures) from the web. The web walks through a system of hyperlinks using HTTP. Any page can create a hyperlink to anyother page.

(ii) web-server data includes IP addresses, page reference, access time

Q. what is meant by authoritative web-pages?
→ The web consist not only of pages but also of hyperlinks pointing ~~for~~ from one page to another. These hyperlinks contain an enormous amount of latent human annotations that can help automatically infer the notion of authority.

- There are 2 basic ~~strategic~~ strategic models for websites.

(i) Hubs → These are pages with a large no. of interesting links. They serve as a gathering point where people visits, to access a variety of info

(ii) Authority — People would gravitate towards pages that provides the most complete & authoritative info on a particular subject. These websites have the most no. of inbound links from other websites.

Page Rank Algo

(i) method of web structure mining.

(ii) It helps in measuring relative importance of a web page, within a set of similar entities. The websites with more no of links or more links from higher quality websites will be ranked higher.

(iii) It is an algo used by Google Search to rank websites in their search engine.

$$PR(A) = (1-d) + d\left(PR(T_i) / C(T_i) + PR(T_n) / C(T_n)\right)$$

$PR(A) \rightarrow$ Page Rank of page A

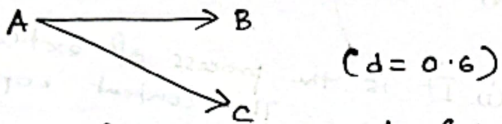
$PR(T_i) \rightarrow$ " " " T_i which link to page A

$C(T_i) \rightarrow$ no. of outbound links on page T_i

$d \rightarrow$ damping factor (teleport ") $(d = 0.85)$

set b/w 0 & 1

[Part-II Q69]



$(d = 0.6)$

initially page rank of all pages = 1 (iteration 0)

iter-1

$PR(A) = (1-0.6) + 0.6\left[0 + 0\right]$

$= 0.4$

$PR(B) = (1-0.6) + 0.6\left[\frac{0.4}{2}\right]$

$= 0.52$

$PR(C) = (1-0.6) + (0.6)\left[\frac{0.4}{2}\right]$

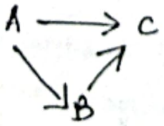
$= 0.52$

iter-2

$PR(A) = (1-0.6) + 0.6 \times 0 = 0$

$PR(B) = 0.52$

$PR(C) = $ "



$d = 0.85$

iter-0

$PR(A) = PR(B) = PR(C) = 1$

iter-1

$PR(A) = (1-0.85) + 0.85 \times 0$

$= 0.15$

$PR(B) = (1-0.85) + 0.85\left(\frac{0.15}{2}\right)$

$= 0.2137$

$PR(C) = (1-0.85) + 0.85\left(\frac{0.15}{2} + \frac{0.2137}{1}\right.$

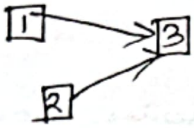
HITS Algo

HITS — Hyperlink Induced Topic Search.

(i) It is a linked analysis algo that rates web pages as being hubs / authorities.

(ii) It uses hubs & authorities to define a recursive relationship b/w web pages.

Q. Complete the hub & authority weights for the following graph.



S1 Find the adj matrix

A =	1	2	3
1	0	0	1
2	0	0	1
3	0	0	0

S2 Find the matrix transpose

A^t =	1	2	3
1	0	0	0
2	0	0	0
3	1	1	0

S3 assume initial hub weight vector is $\begin{bmatrix}1\\1\\1\end{bmatrix} = u$

compute the authority wt. vector

$$V = A^t \cdot U$$

$$= \begin{bmatrix}0 & 0 & 0\\0 & 0 & 0\\1 & 1 & 0\end{bmatrix}_{3\times3} \times \begin{bmatrix}1\\1\\1\end{bmatrix}_{3\times1}$$

$$= \begin{bmatrix}0\\0\\2\end{bmatrix}_{3\times1}$$

S4 update hub weight

$$U = AV$$

$$= \begin{bmatrix}0 & 0 & 1\\0 & 0 & 1\\0 & 0 & 0\end{bmatrix} \times \begin{bmatrix}0\\0\\2\end{bmatrix}$$

$$= \begin{bmatrix}2\\2\\0\end{bmatrix}$$

Node 1 → hub
2 → hub
3 → authority.

Regression Analysis

(i) Statistical method to model the relationship b/w a dependent (target) & independent (predictor) variables with one/more independent variables

$y = f(x)$

dependent ← y , x → independent

(ii) It predicts continuous/real values. (eg. Temperature, age, salary, etc.)

(iii) It is supervised learning technique which helps in finding the co-relation b/w variables & enables us to predict the continuous o/p variable.

(Corelation coeff = -1 to +1)

(iv) Mainly used for prediction, forecasting, time-series modelling & determining the causal effect relationship b/w variables.

(v) In regression, we plot a graph, which best fits the given data point, & using this plot the ML model makes prediction

(vi) The main factor in regression which we want to predict is called the dependent variable (target variable)

(vii) The factors which affect the dependent variables / which are used to predict the values of dependent variables are called independent variables (predictor).

(viii)

Error in Simple Regression

The regression eq. model in ML uses the slope intercept format. It identifies the values of intercept & slope by relating the values of x & y. However identifying the exact match of values for intercept & slope is not always possible. There will be some error (ε) associated with it. ε is called marginal / residual error.

(Residual is the dist b/w predicted point on ~~act~~ regression line & actual point)

(ix) A st. line is drawn as close as possible over the points on the scatter plot. Odinary Least Square (OLS) is the technique used to estimate a line that will minimise the error (ε), which the diffenence b/w the predicted & actual ~~point~~ values of y. This means summing the error of each prediction, (sum of the sq. of the errors)

$SSE = \sum_i \varepsilon_i^2$

if $y = (a + bx) + \varepsilon$

It is observed that SSE is least when 'b' takes

$$b = \frac{\Sigma (x_i - \bar{x})(y_i - \bar{y})}{\Sigma (x_i - \bar{x})^2} = \frac{\text{cov}(x, y)}{\text{var}(x)}$$

The corresponding value of 'a' is calculated as

$$a = \bar{y} - b\bar{x}$$

OLS algorithm

S1: Calculate the mean of x & y

S2: Calculate the errors of x & y

S3: Obtain the product

S4: obtain the sum of the products

S5: Sq. the diff. of x

S6: Get the sum of the sq. difference

S7: Divide o/p of S4 by o/p of S6 to calculate slope 'b'

S8: Calculate intercept 'a' using the value of 'b'.

Q. A random sample of 15 students in a class was selected & internal & external exam data is given below.

(i) Construct the regression line.

(ii) What is your estimate of the marks a student can obtain in the ext. exam if the student obtain 25 in internals.

	X	Y	$X-\bar{X}$	$Y-\bar{Y}$	$(X-\bar{X})^2$	$(X-\bar{X})(Y-\bar{Y})$
	15	49				
	23	63				
	18	58				
	23	60				
	24	58				
	22	61				
	22	60				
	19	63				
	19	60				
	16	52				
	24	62				
	11	30				
	24	59				
	16	49				
	23	68				
Σ	299	852			226.9335	429.8

$\bar{X} = 19.933$

$\bar{Y} = 56.8$

$y = a + bx$

$b = \frac{429.8}{226.9335} = 1.894$

$a = \bar{y} - b\bar{x}$

$= 56.8 - 1.894 \times 19.933$

$= 19.047$

(i) $\therefore y = 19.047 + 1.894x$

(ii) 66.397