Clustering

- unsupervised learning technique used to group a set of data points into clusters such that.
- data points in same group are more similar to each other than to those in other groups.
- No predefined labels are used

Need of clustering

1. Market segmentation
2. Image compression
3. document categorization
4. Anomaly detection.

clustering algo's

1. K means
2. Hierarchical.
3. Time series analysis

K-means.

1. Initialize the clusters.
   - get the value of K
   - choose initial centroids randomly or logically
2. Assign each point to the nearest centroid. based on distance calculated by

$$\text{Distance} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

(x_2: new val, x_1: centroids val.)

3. Repeat assignment for all points
4. Recalculate centroids.

$$\text{new centroid} = \left(\frac{x_1 + x_2}{2}, \frac{y_1 + y_2}{2} \right)$$

5. Repeat steps 2 to 4

Adv.
- Simple & fast for large datasets.
- Efficient when clusters are well seperated

Dis.
- Requires predefined k
- Sensitive to initial centroids
- Struggals with non spherical or ovelaping clusters.

Hierarchical clustering.
- Builds a tree like structure to show nested grouping of data points
- does not require the prepsrocessing the no. of clusters.

2 types
① Agglomerative (Bottom up) - start by each point as a cluster
② Divisive (Top down) - 1 cluster & split recursively
- leaves → individual data points
- Branches represents cluster merges.
- tree structure is known as Dendrogram.

① Agglomerative clustering.
- Bottom up. upproach.
- this algo considers each dataset as a single cluster & then starts combining the closest pare of cluster.
- close pairs are meanged
- done until only 1 cluster is remaining.

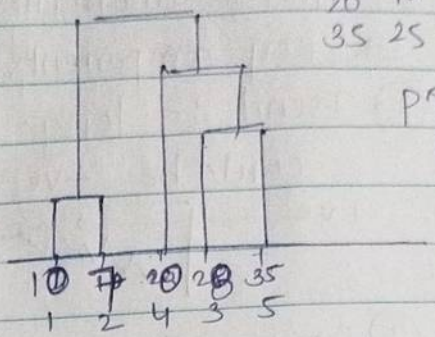
Linkage method.
① Single linkage
② centroid linkage
③ Average linkage
④ complete linkage

x/x	10	7	28	20	35
10	0	3	18	10	25
7	3	0	21	13	28
28	18	21	0	8	7
20	10	13	8	0	15
35	25	28	7	15	0

proximity matrix

eg.

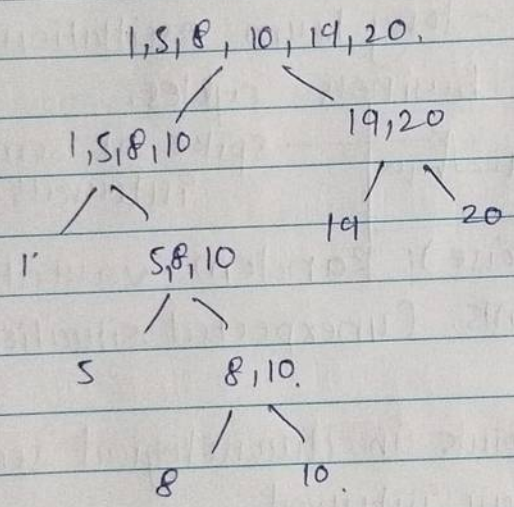
Sid.	marks
1	10
2	7
3	28
4	20
5	35



- use proximity matrix.

(2) Divisive Hierarchical clustering.
- top down strategy. , reverse of agglomerative.
- subdivides a cluster to smaller pieces.

eg.



- In py. we us skypy lib. for. clusters. (hierarchical)

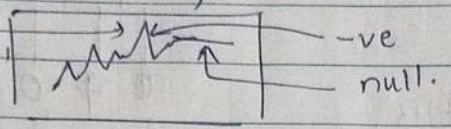
Real would application.
(1). customer segmentation → groupin on purchase behaviour
(2). Document classification. → based on topic similarity
(3). Image Segmentation → grouping pixels. of similar colors.
(4). Social Network analysis → detect comunities / hierarchical relations

Time Series Analysis :- Stat technique that deals with time ordered data points. It is used to analyze trends, patterns, seasonal variations, irregularity

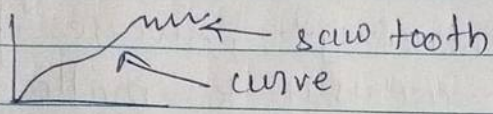
over a period of time to make future predictions or detect anomalies.

key components

① Trend :- long-term movement in the data. could be -ve, +ve or null.

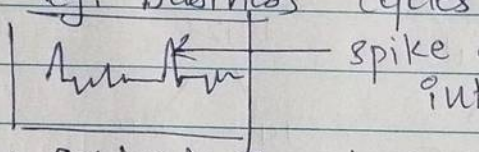


② Seasonality :- Represents short term cycles at regular interval (monthly sails spikes)
- can be curve or saw tooth.



③ cyclic patterns :- long term oscillations not of fixed period. eg. business cycles.

④.



⑤. Irregularity (Noise) : Random variations due to unforeseen events. (unexpected situations)

- sequence of data points in chronological sequence, generally gathered in regular interval.
- used on variable that changes over time

Text Analysis :- ML technique to extract valuable insights from unstructured text data.
Includes. ① Tokenization
② Stop word Removal
③. Stemming & lemmatization
④. POS Tagging.

① Tokenization
- breaking paragraph into small chunks such as words or sentences is called Tokenization.
- Token is a single entity that is the building block for a sentence or paragraph

2 types.

① Sentence Tokenization :- splits a para into list of sentences. uses sent_tokenize() method.

② Word Tokenization :- a sentence into List of words. using word_Tokenize() method.

eg. "I am Shriniwas"
"I" "am" "Shriniwas"

② Stop word removal.
- Stop words are words who dont have a lot of meaning by themself eg. is, am, the, an, are etc.
- considered as noise in the text.

#flow

Tokenized words.
↓
Remove stopwords. (choose words which are not in stop word catogary)
↓
Result (Meaningful words)

③ Stemming :- Normalization technique where list of tokenized words are converted into shortend root words. to remove redundancy
- chops of the words ending to get the root form.
- may not always give meaningful word.
eg. studies ⟶ studi
studying ⟶ study

(3). Lemmatization. root / base form
- the process of finding the lemma of a word depending on its meaning & context.
- uses grammitical ~~ifo~~. info. to find lemma.
  running, ran → run
  eg. Studies → study
  studying → study
  Grammatical info.
- uses vocabulary & morphological analysis

(4). POS tagging (Part of speach)
- tells us about the grammatical info. of words of the sentence by assigning specific tokens as tag to each words.

Part of speach	Tag.
Noun	n
Verb	v
Adjective	a
adverb	r

- helps in understanding the structure & meaning of sentence
- used in parsing, named entity recognition, lemmatization.

TF-IDF
- Term frequency - inverse document frequency.
- A numerical or statistical way of identifying. how important a word is to a document in a collection or corpus (No. of documents)

Term frequency : measure of frequency of a word (w) in a document (d)
- ratio of word's occurance in a document to total no. of words in a document.

$$TF(w,d) = \frac{\text{occurance of word } w \text{ in } d}{\text{Total no. of words in doc. } d}$$

IDF.

- measure of importance of a word.
- IDF provides weightage to each word based on its frequency in corpus D.

$$IDF(w,D) = \log_e \left(\frac{\text{No. of docs in corpus}}{\text{No. of docs containing } w} \right)$$

natural log. i.e. $\log_{10}$

TF-IDF is product of TF & IDF

- gives more weightage to words that are rare in corpus.

$$TF\text{-}IDF(w,d,D) = TF(w,d) * IDF(w,D)$$

- cant capture semantics. (dis).

Adv.

1. Simple & efficient.
2. Highlights imp words.
3. Good baseline for NLP models.
4. works well with sparse data.
5. Domain independent.

Dis.

1. ignores word order & context
2. Static vocabulary.
3. No deep understanding of sentence meaning.
4. large feature space.

Social Network analysis

- process of investigating social structures in terms of nodes & edges that connect them through the use of networks & graph theory
- used to study relationship & structure within a network
  node → entity (individuals)
  edges → relationships
- uses concept of graph theory to model social structure
- There are 4 types of edges.
  1. Symmetric $A \rightleftarrows B$ } based on Directionality.
  2. Asymmetric $A \rightarrow B$ } // has child of relationship
  3. Binary $A \xrightarrow{0} B$ } based on weight.
  4. Valued $A \xrightarrow{20} B$ }

potential connections = $\frac{n*(n-1)}{2}$ // max conn possible

Density = $\frac{\text{Actual conn}}{\text{potential conn}}$

Degree cardinality
- measure of direct ties to a node.
- indegree + outdegree.

closeness cardinality.
- measure of how close a node is to rest of the network
- inverse of sum of (dist from node 1 to others)

for node 1 eg.

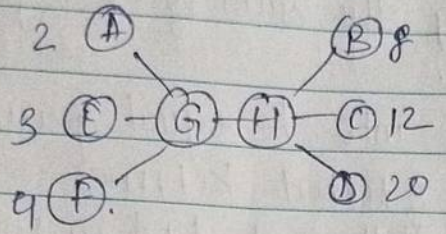
Node1	Node2	Node3	Node4	Node5	Total.
0	2	3	4	1	10

∴ closeness of node 1 is 1/10.
do same for rest of the tree/graph.

Betweenness Centrality
- measure of how often a node appears in the shortest path connecting two other nodes.

Eigen Centrality :- importance of a node in a network
- each node is assigned a no. depending upon the no. of other.

prominent scoring node connected to it.



H will have higher val. than G.

model evaluation techniques.

1. Holdout method
2. cross validation.
   - a. k-fold.
     - i) Stratified k-fold
     - ii) Repeated k-fold.
   - b. leave-one-out cross validation.
3. Sub-sampling (monte carlo cross validation)

1. Holdout method.

- divide available dataset into two distinct set Training (70-80%) & Testing (20-30%)

working.

1. split the dataset randomly
2. train module using training set
3. Evaluate model on testing set
4. Models performance based on testing set.

adv.

simple, fast, useful for large dataset.

dis.

not for small dataset, high variance, test set may not represent the whole data distribution.

(2) cross validation.

- repeatedly splitting the data & averaging the result

k-fold. (#) K fold cross validation.

- dataset is divided into K equal parts (folds)
- model is trained & tested K times.
- each time with different fold as test set.
- K-1 fold for training.

working (1 fold for testing., repeat K times.
& average the performance)

adv.

less biased estimate of performance
all data points used for training & testing

dis.

compute intensive
data leakage may happen

LOOCV (5). (#) leave one out cross validation.

- extreme version of K fold
- in each iteration 1 data point is used for testing & the rest for training

working ①. for each datapoint 1 datapoint test & rest train
②. repeat for all data points ③. average the results.

eg. 5 dp's. $[P_1, P_2, P_3, P_4, P_5]$

Situation 1 P_1 $[P_2 - P_5]$

Situation 2 P_2^{test} $[P_1, P_3 - P_5]^{train}$

Adv.

full use of data for training.
Reduced bias.

dis.

slow.
not suitable training time very high.

(3) Sub-Sampling.
- performs repeated random splits of the dataset into train & testing set.
- Each split is independent, & performance avg over all splits.

working (1). Randomly split the dataset (80, 20)
(2). Train & Test the model.
(3). Repeat N times.
(4). avg. the result.

Adv.
better than single holdout split
Easy to implement.

dis.
some points may never be tested
not all data points are equally used.

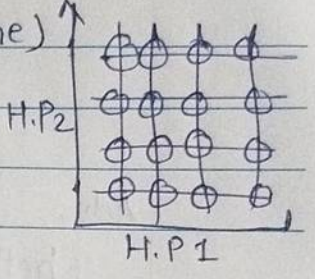
parameter tuning & optimization.
- input parameters to ML model is known as hyperparameters.
- hyperparameters define the architecture of model.
- choise of ↑ is up to us.
- automated selection process of hyperparameter is known as Hyperparameter Tuning.

3. approaches for hyperparameter tuning.
1. Manual
2. Random
3. Grid.

(1) Manual. approach (based on gut feeling seletion of Hyppara.
- based on parameters, the model is trained, & model performance measure are checked.
- this process is repeated for another set of values for same set of hyperparameters. until optimal accuracy is received.
- Not optimal (human jugment is biased)

② Random Search / approach.
- giving multiple val to the hyperparameters in one go & let the model decide which one best suits it

③. Grid approach / search of model. tuning
- expensive (computational power & time)
- most efficient.
- least posibility of missing out on an optimal solution for a model.



Confusion Matrix.

		predicted.	
~~Actual~~	~~Pos~~	Positive	Negative.
Actual {	Positive	TP	FN
	negative	FP	TN

FP → Type 1 error.
FN → Type 2 error.

Sensitivity = $\frac{TP}{(TP+FN)}$	Specificity = $\frac{TN}{(TN+FP)}$
precision = $\frac{TP}{(TP+FP)}$	-ve predictive value = $\frac{TN}{(TN+FN)}$
Accuracy = $\frac{TP+TN}{TP+TN+FP+FN}$	

- to evaluate the performance of classification models.
- used in binary classification.
- provides actual vs predicted classification. that includes. TP, TN, FP, FN to calc. sensitivity, specificity, precision etc.

ROC-AUC curve (for binary classification problems)

- extends confusion matrix.
- evaluate performance across all thresholds.
- It is a graph showing TPR Vs ~~False~~ FPR (True positive rate vs False positive rate) at various classification thresholds.
- Separates the signal from the noise

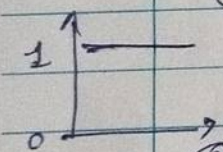
ROC (Receiver operator characteristics) - an evaluation metric for binary classification problems.

~~Are~~ AUC (Area under the curve) - measure of the ability of a classifier to distinguish between classes & its used as a summary of the ROC curve

- higher the AUC better the performance of model. to distinguish b/w +ve & -ve classes.

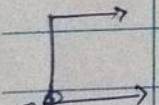
AUC values.

1. AUC = 1



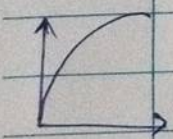
classifier is able to perfectly distinguish b/w. all +ve & -ve classes correctly.

2. AUC = 0 was 0 → 1.



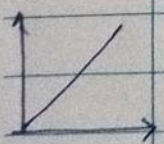
all -ve as +ve & +ve as -ve

3. ~~AUC~~ 0.5 < AUC < 1



classifier will be able to distinguish ~~b/w~~ the ~~from~~ +ve class val from -ve class value

4. AUC = 0.5



classifier NOT able to distinguish b/w. +ve & -ve class points.

$\frac{13.5}{2}$ $\frac{11}{2}$

6.75 | 5.5

half solved. bt yes correct.

Page No.
Date:

A1 2,10
A2 2,5
A3 8,4
B1 5,8 | 6.5, 6
B2 7,5
B3 6,4
C1 (1.5, 3.5)
C2 4,9

$D(A_1, A_2) = \sqrt{(2-2)^2 + (5-10)^2}$

$= 5$

$D(A_2, B_1) = \sqrt{(5-5)^2 + (5-8)^2}$

$= \sqrt{9+9}$

$= \sqrt{18}$

$= 4.24$

$D(A_2, C_1) = \sqrt{(2-1)^2 + (5-2)^2}$

$= \sqrt{1+9}$

$= 3.16$

$\frac{2+1}{2}$ $\frac{5+2}{2}$ $\frac{8}{2}$ 3.5

A1

B1, A3, B2

6.75 | 5.5

$D(A_3, A_1) = \sqrt{(8-2)^2 + (4-10)^2}$

$\sqrt{36+1}$

$\sqrt{37}$

C1, A2

(1.5, 3.5)

$D(B_1, A_3) = \sqrt{(8-5)(4-8)}$

$= \sqrt{9+16} = 3\sqrt{25}$

$D(A_3 C_1) = \sqrt{(8-1.5)(4-3.5)}$

$= \sqrt{42.5}$

$\sqrt{(7-2)^2 + (5-10)^2} = \sqrt{25+25} = \sqrt{50}$

$\sqrt{(7-6.5)^2 + (5-6)^2}$ $\sqrt{0.25+1} = \sqrt{1.25}$

$\sqrt{(7-1.5)^2 + (8-3.5)^2} = \sqrt{30.25 + 2.25}$

1.11