DMBI (SEM VI)

PAGE No. 1
DATE / /

FORMULA LIST

* **Mean ($\bar{x}$)** = $\frac{1}{n}\sum_{i=1}^{n} x_i$

* **Empirical Formula** for unimodal numeric data -
  Mean - mode = 3(mean - median)

* **Midrange** = (Max value + Min value) / 2

* **Measuring dispersion of data** :-

↳ **Quartiles** → Q_1 (25th percentile) (1/4 x Median) , Q_3 (75th Percentile) (3/4 x Median)

↳ **Inter Quartile Range (IQR)** = $Q_3 - Q_1$

↳ **Five Number Summary** = (min, Q_1, median, Q_3, max)

↳ To calculate **outliers** :
$Q_1 - (1.5 \times IQR)$ & $Q_3 + (1.5 \times IQR)$

↳ **Variance** (σ^2) = $\frac{\sum_{i=1}^{n} (x_i - \mu)^2}{n}$

↳ **Standard Deviation (SD)** = $\sqrt{\text{Variance}} = \sqrt{\frac{\Sigma (x - \bar{x})^2}{n}}$

* Measuring data similarity and dissimilarity :

↳ Similarity - Value is higher when objects are more alike often falls in the range (0,1).

↳ Dissimilarity - (Need to calculate the distance between objects). Lower value when objects are more alike.
For similar objects value = 0

↳ Proximity is nothing but similarity or dissimilarity between objects.

↳ Dissimilarity data matrix :-

0			
d(2,1)	0		
d(3,1)	d(3,2)	0	
⋮	⋮	⋮	
d(n,1)	d(n,2)	⋮	⋯ 0

↳ Proximity measure for normal attributes :-

$$d(i,j) = \frac{p-m}{p}$$

{ p = Total no. of attributes
m = no. of matching attribute values }

↳ Proximity measure for binary attributes :-

⇒ For symmetric binary variables :

$$d(i,j) = \frac{r+s}{q+r+s+t}$$

	Object j: 1	0	sum
object i: 1	q	r	q+r
0	s	t	s+t
sum	q+s	r+t	p

q = 11, r = 10, s = 01, t = 00

⇒ For asymmetric Binary variables -

$$d(i,j) = \frac{r+s}{q+r+s}$$

↳ Proximity measure of numeric data -

1) Euclidean Distance -

$$d(i,j) = \sqrt{(|x_{i_1} - x_{j_1}|)^2 + |x_{i_2} - x_{j_2}|^2 + \dots |x_{ie} - x_{jp}|^2)}$$

2) Manhattan Distance -

$$d(i,j) = |x_{i_1} - x_{j_1}| + |x_{i_2} - x_{j_2}| + \dots + |x_{ip} - x_{jp}|$$

3) Cosine Similarity -

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

[$x \cdot y$: Vector dot product
$||x||$: Magnitude of vector]

NOTE : If the cosine value is closer to 1, the objects are more similar.

* HANDLE NOISY DATA -

1) Binning Method : (1) Sort the data & partition into bins
(2) Smooth data by bin means, bin median, bin boundaries, etc.

2) Equal - Depth (Freq) : Partition the data in such a way that each bin has same no. of data

3) Equal - width : Divide the data such that each bin has some width.

To calculate width of a bin :

$$\text{Width} = \frac{\text{Max value} - \text{Min value}}{\text{No. of bins}}$$

4) Clustering : Partition the data along the 2 biggest gaps in the data.

* NORMALIZATION -

1) Min - Max Normalization :

$$v' = \frac{v - \text{Min}_A}{\text{Max}_A - \text{Min}_A} (\text{new\_max}_A - \text{new\_min}_A) + \text{new\_min}_B$$

v = value given to normality
min_A & max_A : Min & Max value of given attribute range
new_max_A & new_min_A : New values given for the attribute range.

2) Z - score Normalization :

$$v' = \frac{v - \mu_A}{\sigma_A}$$

[μ_A : Mean of attribute
σ_A : S.D. of attribute]

3) Decimal Scaling :

$$v' = \frac{v}{10^j}$$ [j = calculated as per the max no. of digits of the range].

07-03-2022

CHAPTER 3 : FREQUENT MINING PATTERNS

* Applications -

↳ Market Basket Analysis (Association Analysis)
↳ Telecommunication
↳ Credit Cards / Banking Services
↳ Medical Treatments.
↳ Basketball - Game Analysis.

* ASSOCIATION RULE PROBLEM :-

① Given a database of transaction.

Transaction	Items
t1	Bread, Jelly, Peanut Butter
t2	B, PB
t3	B, Milk, PB
t4	Beer, B
t5	Beer, M

Find all the association rules :

X ⇒ Y	S	α
B ⇒ PB	60%	75%
PB ⇒ B	60%	100%
Beer ⇒ B	20%	50%
PB ⇒ Jelly	20%	33.3%
J ⇒ PB	20%	100%
J ⇒ M	0%	0%

* ASSOCIATION RULE DEFINITIONS

* Support : (Item A + Item B) / Entire Dataset
* Confidence : (Item A + Item B) / Item A

* APRIORI ALGORITHM

① Eg : A dataset D has 4 transactions. Let min support be 50%. & min confidence is 80%.
Find : All frequent item set & also generate association rule using Apriori Algorithm.

Min support count = min supp threshold * Total no. of trans
= (50/100) * 4
= 2

Database D

TID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5

Scan D →

	Itemset	Supp.
C_1	{1}	2
	{2}	3
	{3}	3
	{4}	1 → Pruning because min supp was 2 as calculated above (1 is less than 2)
	{5}	3

↓

C_2 Itemset	Sup.
{1 2}	1
{1 3}	2
{1 5}	1
{2 3}	2
{2 5}	3
{3 5}	2

Scan D ←

	Itemset
C_2	{1 2}
	{1 3}
	{1 5}
	{2 3}
	{2 5}
	{3 5}

←

Item set	Sup
{1}	2
{2}	3
{3}	3
{5}	3

→ L2

Itemset	sup
{1 3}	2
{2 3}	2
{2 5}	3
{3 5}	2

→ C3

Itemset
{2 3 5}

Scan D →

Itemset	sup
{2 3 5}	2

∴ Frequent item sets are
L = {2, 3, 5}

Now for strong association rule :
Generate non-empty subset for L
S = {2}, {3}, {5}, {2,3}, {2,5}, {3,5}
Find S → (L-S)
For eg: {2} → {3,5}

antecedant

S → (L-S)	Support (calculated at the start)	Confidence	Confidence (%)
{2} → {3,5}	2	{2,3,5} count / {2} count = 2/3	66.66
{3} → {2,5}	2	{2,3,5} count / {3} count = 2/3	66.66
{5} → {2,3}	2	2/3 (considered individual)	66.66
{2,3} → {5}	2	2/2	100
{2,5} → {3}	2	2/3	66.66
{3,5} → {2}	2	2/2	100

Minimum confidence threshold = 80%.
∴ Strong association rules are -
{2,3} → {5}
{3,5} → {2}

sup 2

(2) Consider the following database with minimum support count = 60%. Find all the frequent itemset using apriori algorithm and also generates strong association rules if minimum confidences = 50%.

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

Hint: O is bought 4 times in total, but it occurs in just 3 transactions.

Soln: Min support count = Min support threshold * Total no. of transac

$$= \frac{60 \times 5}{100}$$

$$= 3 //$$

Database D

TID	Itemset
T_1	{M, O, N, K, E, Y}
T_2	{D, O, N, K, E, Y}
T_3	{M, A, K, E}
T_4	{M, U, C, K, Y}
T_5	{C, O, O, K, I E}

Scan D →

C_1 Itemset	Supp.
{A}	1
{C}	2
{D}	1
{DE}	4
{I}	1
{K}	5
{M}	3
{N}	2
{O}	3
{U}	1
{Y}	3

Itemset	Supp.
{E}	4
{K}	5
{M}	3
{O}	3
{Y}	3

→ C2 (Scan D) →

Itemset	Supp.
{E, K}	4
{E M}	2
{E O}	3
{E Y}	2
{K M}	3
{K O}	3
{K Y}	3
{M O}	1
{M Y}	2
{O Y}	2

→ L2 (Scan D) →

Itemset	Supp.
{E K}	4
{E O}	3
{K M}	3
{K O}	3
{K Y}	3

↓

C3 (Scan D) →

Itemset	Supp.
{E K O}	3

→ L3

Itemset
{E K O}

∴ Frequent item sets are
L = {E K O}

Now for strong association rule :
Generate non-empty subset for L
S = {E}, {K}, {O}, {E K}, {O K}, {O, E}
Find S → (L-S)

S → (L-S)	Support	Confidence	Confidence (%)
{E} → {O K}	3	3/4	75%.
{O} → {E K}	3	3/3	100%.
{K} → {E O}	3	3/5	60%.
{O K} → {E}	3	3/3	100%.
{E K} → {O}	3	3/4	75%.
{E O} → {K}	3	3/3	100%.

Minimum confidence threshold = 50%.

∴ Strong association rules are -

{O K} → E

{E O} → {K}

{O} → {E K}

* APRIORI ADVANTAGES / DISADVANTAGES :

Advantages

- Use large itemset property
- Easy to implement
-

Disadvantages

- Requires many database scan

* FP GROWTH

- Improves apriori to a big extent.
- Frequent itemset mining is possible without candidate generation.
- Only 'two' scans are required.

STEPS :-

① Build a compact data structure called the FP-tree
- Built using 2 passes over the data-set.

② Extracts frequent item sets directly from the FP tree.

Eg: ~~Database~~ Find the frequent item set. (support count = 2)

① Database D

TID	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

Itemset	Support
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

Descending order

~~Itemset~~ ~~TID~~ Items

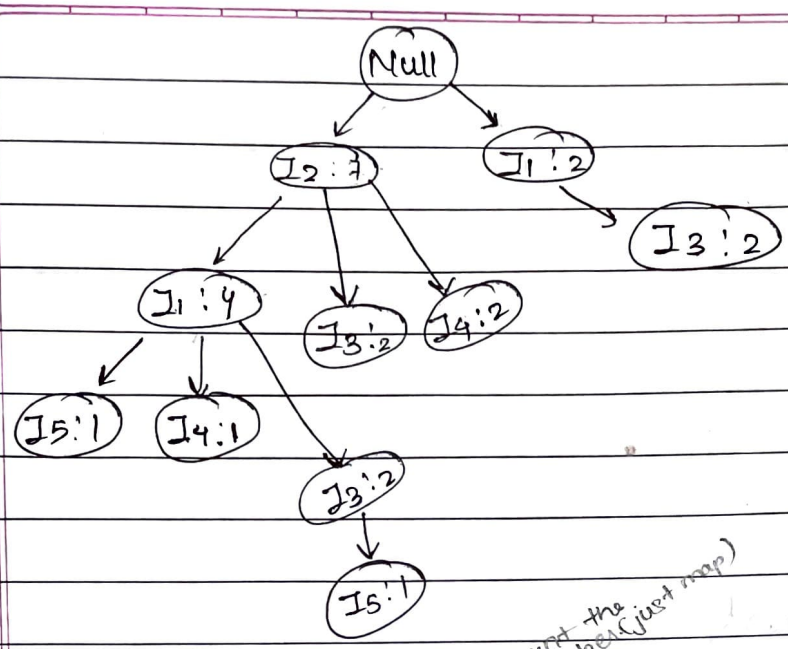
T200 I2, I4
T100 I1, I2, I3
T300 I2, I3
T400 I1, I2, I4
T500 I1, I3

Items	Support
{I2}	7
{I1}	6
{I3}	6
{I4}	2
{I5}	2

(this order should be followed while doing)

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

[∵ I2 > I1 > I5 in this arranged]



count the branches (just map) → take common count → make combinations with them

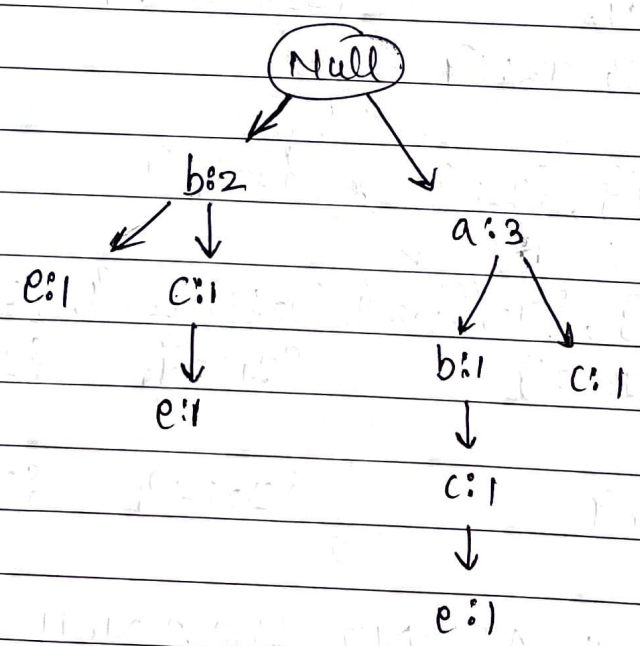
Item	Conditional Pattern Base	Conditional FP-Tree	Frequent Pattern Generated
I_5	$\{I_2, I_1 : 1\}$, $\{I_2, I_1, I_3 : 1\}$	$\{I_2 : 2, I_1 : 2\}$	$\{I_2, I_5 : 2\}$, $\{I_1, I_5 : 2\}$, $\{I_2, I_1, I_5 : 2\}$
I_4	$\{I_2, I_1 : 1\}$, $\{I_2 : 1\}$	$\{I_2 : 2\}$	$\{I_2, I_4 : 2\}$
I_3	$\{I_2 : 2\}$ (LST), $\{I_2, I_1 : 2\}$, $\{I_1 : 2\}$ (RST)	$\{I_2 : 4, I_1 : 2\}$, $\{I_1 : 2\}$	$\{I_2, I_3 : 4\}$, $\{I_1, I_3 : 2\}$ (two times because 2 times), $\{I_2, I_1, I_3 : 2\}$, ~~$\{I_1, I_3\}$~~
I_1	$\{I_2 : 4\}$	$\{I_2 : 4\}$	$\{I_2, I_1 : 4\}$
I_2	IGNORE AS NO BRANCH		

(2) Draw FP tree

T1 : b, e
T2 : a, b, c, e
T3 : b, c, e
T4 : a, c
T5 : a

Soln:

Items	Support
a	3
b	3
c	3
e	3



③ Draw the FP tree. Support count = 3.

T_1 : f, a, c, d, g, i, m, p

T_2 : a, b, c, f, l, m, o

T_3 : b, f, h, j, o

T_4 : b, c, k, s, p

T_5 : a, f, c, e, l, p, m, n

Soln:

Items	support
a	3
b	3
c	4
d	1
e	1
f	4
g	1
h	1
i	1
j	1
k	1
l	2
m	3
n	1
o	2
p	3
s	1

~~Descending order~~

Items	support
a	3
b	3
c	4
f	4
m	3
p	3

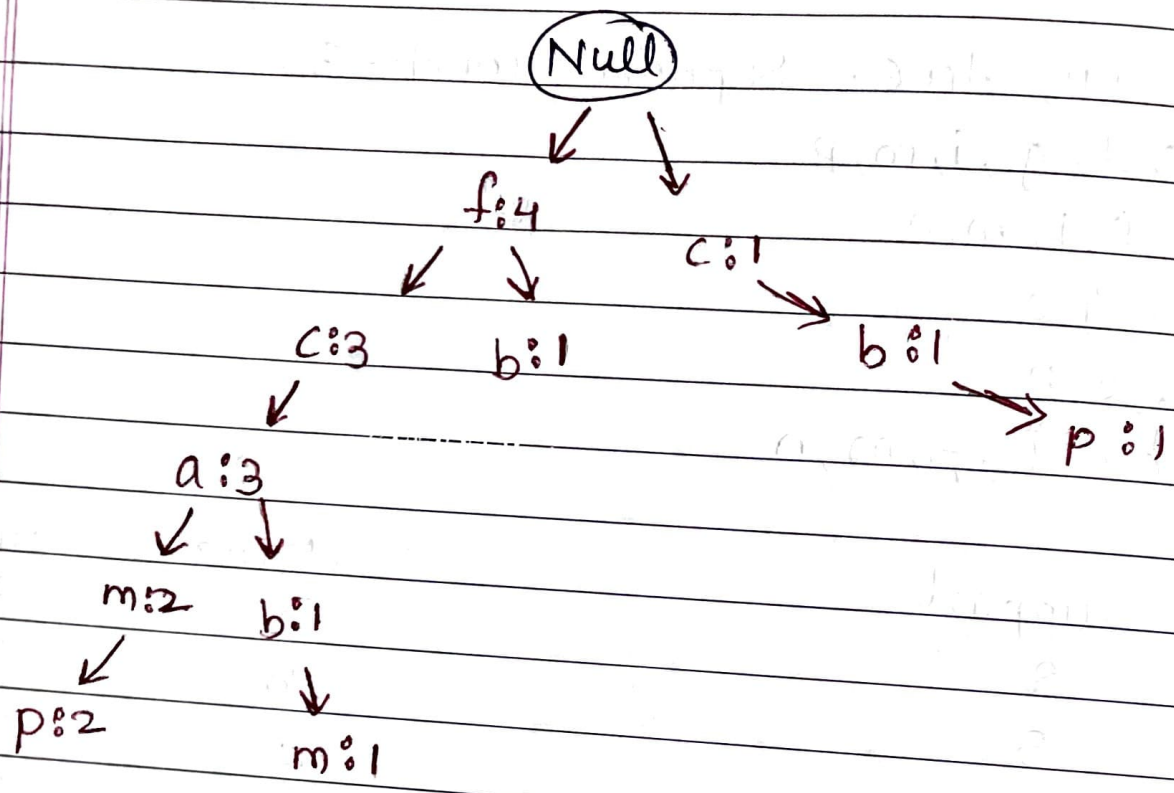
↓ Descending order

~~Items~~ ~~support count~~

Items	support
f	4
c	4
a	3
b	3
m	3
p	3

↓

TID	Items
T_1	f, c, a, m, p
T_2	f, c, a, b, m
T_3	f, b,
T_4	c, b, p
T_5	f, c, a, m, p


Null
f:4
c:1
c:3
b:1
b:1
p:1
a:3
m:2
b:1
p:2
m:1

21-03-2022

CHAPTER 4 : CLASSIFICATION TECHNIQUES

* **Classification :** (Definition)
  ↳ Process of finding a model that describes data classes.
  ↳ For the purpose of being able to use the model.
  ↳ To predict the class of objects whose class label is unknown.

* **SUPERVISED CLASSIFICATION :** (Type 1)
  ↳ Set of possible classes is known in advance.

* **UNSUPERVISED CLASSIFICATION :** (Type 2)
  ↳ Set of possible classes is not known. After classification we can try to assign a name to that class. Unsupervised classification is called clustering.

NOTE : Classification consists of assigning a class label to a set of unclassified class.

* **WHY CLASSIFICATION? A MOTIVATING APPLICATION**

Eg :

- Credit Approval
  - A bank wants to classify its customers based on whether they are expected to pay back their approved loans.
  - The history of past customers is used to train the classifier.
  - The classifier provides rules, which identify potentially reliable future customers.
  - Classification rule :
    - If age = "31...40" & income = high & credit_rating = excellent.

- Future customers
  - Paul : age = 35, income = high => Excellent credit rating
  - John : age = 20, income = medium => Fair credit rating

* TYPICAL APPLICATIONS OF CLASSIFICATION :

- Credit approval
- Target Marketing
- Medical Diagnosis
- Fraud Detection

* CLASSIFICATION - A TWO STEP PROCESS

① MODEL CONSTRUCTION :
- Describing a set of predetermined classes.
- The model is represented as classification rules, decision trees, or mathematical formulae.

② MODEL USAGE :
- For classifying future or unknown objects.
- Test sample is compared with the classified result from the model.

Classification Model can be in the following forms :
↳ IF-THEN Rules
↳ A decision Tree
↳ Neural Network .... (Not in syllabus)

CLASSIFIER TECHNIQUES :

* BAYESIAN

- A Bayes. classifier is a simple probabilistic classifier.
- A naive Bayes classifier will be studied.

NAÏVE BAYES CLASSIFICATION –

(1) Eg : Predict the class label for an unknown sample "X" using Naive Bayesian classification.

'X' = (Outlook = Sunny, Temperature = Cool, Humidity = High, Wind = Strong).

Play Tennis : Training Examples (Target attribute)

Day	Outlook	Temperature	Humidity	Wind	Play Tennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	~~Weak~~ Normal	Weak	Yes
D10	Rain	Mild	~~Weak~~ Normal	Weak	Yes
D11	Sunny	Mild	~~Weak~~ Strong Normal	Strong	Yes
D12	Overcast	Mild	~~Strong~~ High	Strong	Yes
D13	Overcast	Hot	~~Weak~~ Normal	Weak	Yes
D14	Rain	Mild	~~Strong~~ High	Strong	No

Soln: Learning Phase :

P(Play = Yes) = 9/14 P(Play = No) = 5/14

Outlook	Play = Yes	Play = No
Sunny	2/9	3/5
Overcast	4/9	0/5
Rain	3/9	2/5

Temp	Play = Yes	Play = No
Hot	2/9	2/5
Mild	4/9	2/5
Cool	3/9	1/5

Humidity	Play = Yes	Play = No
High	3/9	4/5
Normal	6/9	1/5

Wind	Play = Yes	Play = No
Strong	3/9	3/5
Weak	6/9	2/5

Test Phase :

- Given a new instance,
  x' = (Outlook = Sunny, Temperature = Cool, Humidity = High, Wind = Strong)

- MAP RULE

↳ P(x' | Yes) = P(Outlook = Sunny / yes) * P(Temp = cool / Yes) * P(Humidity = High / yes) * P(Wind = Strong / Yes) * P(Yes)

= 2/9 * 3/9 * 3/9 * 3/9 * 9/14

= 0.0053 //

↳ $P(x'/No) = P(Outlook = Sunny/No) * P(Temp = cool/No) * P(Humidity = High/No) * P(Wind = Strong/No) * P(No)$

$= 3/5 * 1/5 * 4/5 * 3/5 * 5/14$

$= 0.0206$ //

V. Imp:

⟹ Given the fact $P(x/yes) < P(x/No)$, we label x to be "Play Tennis = No." (.... Last statement has 2 Marks).

(2) Naïve Bayesian classification eg:

- Predict a class label of an unknown sample using Naïve Bayesian classification on the following training dataset from all electronics customer database.
- The unknown sample is -

$X' = \{age = "<=30", Income = "Median," Student = "Yes", Credit\ rating = "fair"\}$

Age	Income	Student	Credit_Rating	Buys_computer
<=30	High	No	Fair	No
<=30	High	No	Excellent	No
31...40	High	No	Fair	Yes
>40	Medium	No	Fair	Yes
>40	Low	Yes	Fair	Yes
>40	Low	Yes	Excellent	No
31...40	Low	Yes	Excellent	Yes
<=30	Medium	No	Fair	No
<=30	Low	Yes	Fair	Yes
>40	Medium	Yes	Fair	Yes

<=30	Medium	Yes	Excellent	Yes
31...40	Medium	No	Excellent	Yes
31...40	High	Yes	Fair	Yes
>40	Medium	No	Excellent	No

Soln: Learning Phase:

P(Buys_computer = Yes) = 9/14 P(Buys_computer) = 5/14

Age	Play = Yes	Play = No
<=30	2/9	3/5
31...40	4/9	0/5
>40	3/9	2/5

Income	Play = Yes	Play = No
High	2/9	2/5
Medium	4/9	2/5
Low	3/9	1/5

Student	Play = Yes	Play = No
Yes	6/9	1/5
No	4/9	4/5

Credit_Rating	Play = Yes	Play = No
Excellent	3/9	3/5
Fair	6/9	2/5

Test Phase:

- Given a new instance,
  $x' = \{$ age = "< 30", Income = "Medium", Student = "Yes",
  Credit_Rating = "Fair" $\}$

- MAP RULE

↳ $P(x'|Yes) = P(Age = "<30"/Yes) * P(Income = "Medium"/Yes) * P(Student = Yes) * P(Credit\_Rating/Yes) * P(Yes)$

$= 2/9 * 4/9 * 6/9 * 3/9 * 9/14$

$= 0.028$ //

↳ $P(X' | No) = P(Age = <30 / No) * P(Income = Medium / No) * P(Student / No) * P(credit\_Rating / No) * P(No)$

$= 3/5 * 2/5 * 1/5 * 2/5 * 5/14$

$= 0.007$ //

Since $0.028 > 0.007$,

∴ The Naïve Bayesian classifier predicts Buyers computer = "Yes" for sample X.

(Very lengthy don't attempt in exam)

* **DECISION TREE** – (Classification)

- Builds classification models in the form of a tree structure.
- It breaks down data into smaller subsets.

Eg:

① Using given training data set, create classification model using decision tree.

(Table same as Eg 1 of Naïve Bayes classification on pg 19 in the N.B.)

Soln :- $E(S) = \sum_{i=1}^{c} -P_i \log_2 P_i$ → No. of values.

(E → Entropy)

Play Golf	
Yes	No
9/14	5/14

Entropy (Play Golf) = Entropy (5, 9) ↵ 5/14, 9/14

= Entropy (0.36, 0.64)

$= -(0.36 \log_2 0.36) - (0.64 \log_2 0.64)$

= 0.94

⇒ $E(T, X) = \sum_{c \in X} P(c) E(c)$ (Target)

		Play Golf		
		Yes	No	
OUTLOOK	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5

↓

E (PlayGolf, Outlook) = P(sunny) * E(3, 2) + P(Overcast) * E(4, 0) + P(Rainy) * E(2, 3)

(Probability; From the table.)

$= (5/14) * 0.971 + (4/14) * 0.0 + (5/14) * 0.971$ (Put in the above entropy formula.)

= 0.693

⇒

		Play Golf		
		Yes	No	
TEMPERATURE	Hot	2	2	4
	Mild	4	2	6
	Cold	3	1	4

$E(\text{PlayGolf}, \text{Temp}) = P(\text{Hot}) * E(2/4, 2/4) + P(\text{Mild}) * E(4/6, 2/6) + P(\text{cold}) * E(3/4, 1/4)$

$= 4/14 * 1 + 6/14 * 0.92 + 4/14 * 0.81$

$= 0.91$

→ Similarly for $E(4/6, 2/6)$ & $=$ (5/7, ...)

[E = Entropy(0.5, 0.5)
$= -(0.5 \log_2 0.5) - (0.5 \log_2 0.5)$
$= 1$]

Similarly,
E (Humidity) = 0.788
E (Wind) = 0.892

⇒ $\boxed{\text{Gain}(T, X) = \text{Entropy}(T) - \text{Entropy}(T, X)}$ → FORMULA OF INFORMATION GAIN.

- $G(\text{PlayGolf}, \text{Outlook}) = E(\text{PlayGolf}) - E(\text{PlayGolf}, \text{Outlook})$
  $= 0.940 - 0.693$
  $= 0.247$

- $G(\text{PlayGolf}, \text{Temp}) = E(\text{PlayGolf}) - E(\text{PlayGolf}, \text{Temp})$
  $= 0.940 - 0.91$
  $= 0.030$

- $G(\text{PlayGolf}, \text{Humidity}) = E(\text{PlayGolf}) - E(\text{PlayGolf}, \text{Temp})$
  $= 0.940 - 0.788$
  $= 0.152$

- G(Play Golf, Wind) = E(Play Golf) − E(Play Golf, ~~Hum~~Wind)
  = 0.940 − 0.0892
  = 0.048

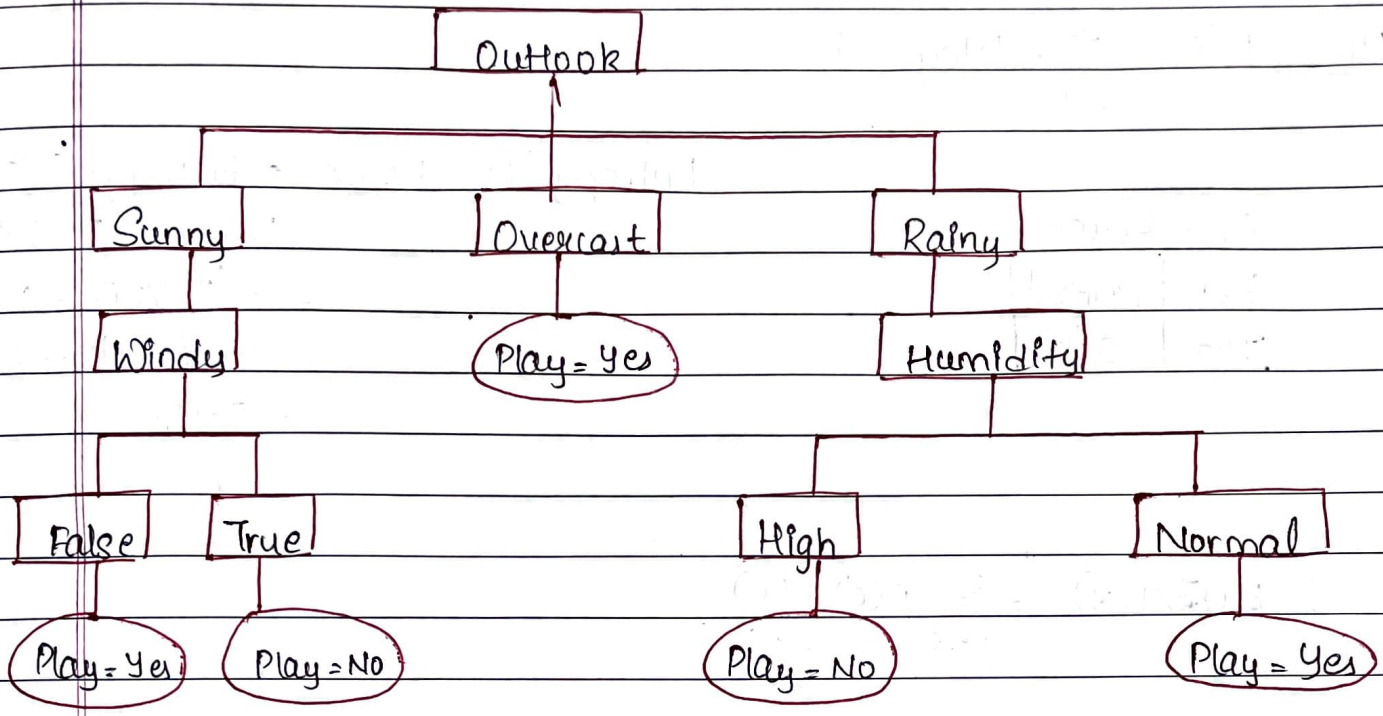
∵ The highest value is the outlook,
∴ Root Node will be the outlook.

Outlook = Overcast (Mini table) → Refer the main table.

Temp	Humidity	Windy	Play Golf
Hot	High	False	Yes
Cool	Normal	True	Yes
Mild	High	True	Yes
Hot	Normal	False	Yes
Hot	High	False	yes

NOTE : Keep calculating the entropies of the sub-division also until you get the final leaf note.

R1 : IF (Outlook = Sunny) AND (Windy = False) THEN Play = Yes
R2 : IF (Outlook = Sunny) AND (Windy = TRUE) THEN Play = No
R3 : IF (Outlook = Overcast) THEN Play = Yes
R4 : IF (Outlook = Rainy) AND (Humidity = High) THEN Play = No
R5 : IF (Outlook = Rain) AND (Humidity = Normal) THEN Play = Yes



DECISION TREE

(2) Create classification model using decision tree for the following training data set.

SR.NO.	Income	Age	Own House
1	Very High	Young	Yes
2	High	Medium	Yes
3	Low	Young	Rented
4	High	Medium	Yes
5	Very High	Medium	Yes
6	Medium	Young	Yes
7	High	Old	Yes
8	Medium	Medium	Rented
9	Low	Medium	Rented
10	Low	Old	Rented
11	High	Young	Yes
12	Medium	Old	Rented

Soln: $E(S) = \sum_{i=1}^{C} -P_i \log_2 P_i$

Own House	
Yes	Rented
7/12	5/12

Entropy (Own House) = Entropy (7/12, 5/12)
= Entropy (0.58, 0.41)
$= -(0.58 \log_2 0.58) - (0.41 \log_2 0.41)$
= 0.98

$\Rightarrow E(\text{Own House}, T, X) = \sum_{c \in X} P(c) E(c)$

		Own House		
		Yes	Rented	
INCOME	Low	0	3	3
	Medium	1	2	3
	High	4	0	4
	Very high	2	0	2

E(Own House, Income) = P(Low) * E(0, 3/3) + P(Medium) * E(1/3, 2/3)
+ P(High) * E(4/4, 0) + P(very high) * E(2/2, 0)

$= \frac{3}{12} \times 0 + \frac{8}{12} \times 9.18 + \frac{4}{12} \times 0 + \frac{2}{12} \times 0$

= 0.23

		Own House: Yes	Own House: Rented	
AGE	Young	3	1	4
	Medium	3	2	5
	Old	1	2	3

E(own house, Age) = P(Young) * E(3/4, 1/4) + P(Medium) * E(3/5, 2/5) + P(old) * E(1/5, 2/5)

$$= \frac{4}{12} \times 0.81 + \frac{5}{12} \times 0.97 + \frac{3}{12} \times 0.92$$

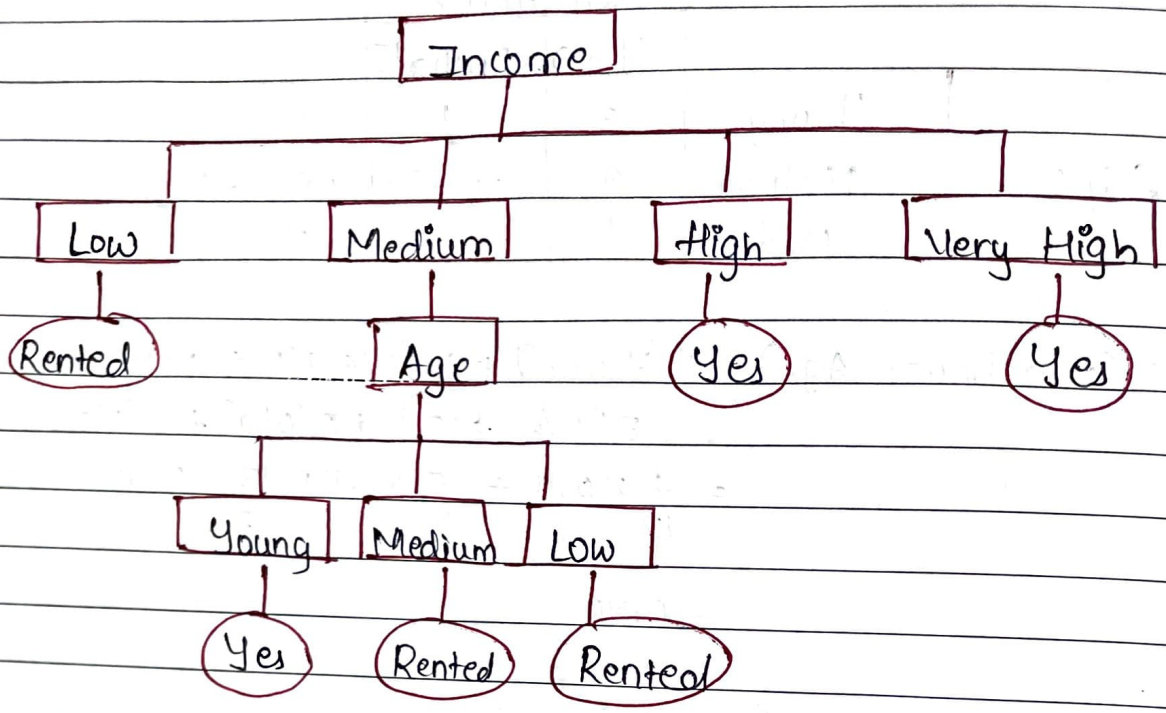
$$= 0.90$$

∴ The higher value is age
∴ The root note will be the age.

- G(O, A) = E(O) − E(O, A)
  = 0.98 − 0.90
  = 0.08

- G(O, I) = E(O) − E(O, I)
  = 0.98 − 0.823
  = 0.75

∴ Income has the higher gain, ∴ The root note will be the income.



* PREDICTION (Data Mining Technique)

- We can predict the continues value of response variable with the help of predictor variable.
- It can be done with the help of statistical technique of regression.
- It assumes the data to fit in some kind of function & involve study of those function.
- Most widely used approach for numeric prediction is regression.

* TYPES OF REGRESSION -

1. Linear
2. Multiple Linear
3. Non-Linear

⇒ **LINEAR REGRESSION** (Single predictor variable)

- Data is modeled using straight line.
- FORMULA = $y = \alpha + \beta x$ (x → independent variable; y → Dependent variable)

  Where $x \rightarrow$ predictor variable
  $y \rightarrow$ Response variable

⇒ **MULTIPLE LINEA REGRESSION** (Multiple predictor variable)

- FORMULA = $y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3$

⇒ **NON-LINEAR REGRESSION.**

- If the given response variable & predictor variable have got polynomial relationship then it is called non-linear regression.
- FORMULA = $y = \alpha + \beta_1 x + \beta_2 x^2 + \beta_3 x^3$

* **How to calculate α & β.**

① **Using least square method**

$$\beta = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

$$\alpha = \bar{y} - \beta \bar{x}$$

$\bar{x} \rightarrow$ Mean value of x
$\bar{y} \rightarrow$ Mean value of y

Eg

①

Midterm (x)	Final year (y)
45	60
70	70
60	54
84	82
75	68
84	76

Find: 1) Equation of prediction (linear regression formula)
2) What will be the final year marks if the midterm marks is 40?

Soln:

x	y	$(x-\bar{x})$	$(y-\bar{y})$	$(x-\bar{x})(y-\bar{y})$	$(x-\bar{x})^2$
45	60	-24.67	-8.33	205.5	608.608
70	70	0.33	1.67	0.55	0.1089
60	54	-9.67	-14.33	138.57	93.508
84	82	14.33	13.67	195.9	205.348
75	68	5.33	-0.33	-1.76	28.408
84	76	14.33	7.67	109.9	205.348
				$\Sigma = 648.66$	$\Sigma = 1141.32$

$\bar{x} = 69.67$

$\bar{y} = 68.33$

1) $$\beta = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

$$\beta = \frac{648.66}{1141.32}$$

$$\beta = 0.568 \text{ //}$$

$$\alpha = \bar{y} - \beta \bar{x}$$
$$= 68.33 - 0.568 \times 69.67$$
$$= 28.76 \text{ //}$$

Equation of prediction = $y = \alpha + \beta x$

$$y = 28.76 + 0.568x$$

2) For $x = 40$, we get $y =$?

$$y = 28.76 + 0.568x \qquad (\because x = 40)$$
$$\therefore y = 28.76 + 0.568 \times 40$$
$$y = 51 \text{ //}$$

$\therefore y = 51$ when $x = 40$.

* MODEL EVALUATION AND SELECTION

- Methods for estimating a classifier's accuracy
  - Holdout method, random subsampling
  - Cross-validation
  - Bootstrap
- Comparing classifier
  - ROC curves

(Q → Explain different methods that can be used to evaluate & compare the accuracy of different classification algorithm) (Ans is the above points).

Q → Explain Bagging & Boosting of classification

* CLASSIFIER EVALUATION METRICS: CONFUSION MATRIX

Confusion Matrix:

Actual class \ Predicted class	C_1	$\bar{C}_1$
C_1	True positives (TP)	False negative (FN)
$\bar{C}_1$	False positives (FP)	True negative (TN)

Eg:

Actual class \ Predicted class	Buy-computer = yes	Buy-computer = No	Total
Buy-computer = yes	6954 (TP)	46 (FN)	7000
Buy-computer = no	412 (FP)	2588 (TN)	3000
Total	7366	2634	10000

$$\text{Accuracy} = \frac{TP + TN}{\text{Total}}$$

$$= \frac{6954 + 2588}{10000}$$

$$= 95\%.$$

* **FORMULLAS:**

IMP ① Classifier accuracy $= \frac{TP + TN}{\text{Total}}$

IMP ② Error Rate $= \frac{FP + FN}{\text{Total}}$ OR $1 - \text{Accuracy}$

IMP ③ Precision $= \frac{TP}{TP + FP}$

IMP ④ Recall / Sensitivity $= \frac{TP}{TP + FN}$

Not IMP: ⑤ F (Measure) (F1 or F-score) $= \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$

IMP ⑥ Specification $= TN / TN + FP$

Eg:

①

Actual class \ Predicted class	cancer = yes	cancer = no	Total
Cancer = yes	90	210	300
cancer = No	140	9560	9700
Total	230	9770	10000

1) Accuracy = $\frac{TP + TN}{Total}$

$= \frac{90 + 9560}{10000}$

$= 96.5\%$ //

2) Error Rate = $\frac{FP + FN}{Total}$ OR Accuracy

$= 1 - 96.5\%$

$= 1 - 0.965$

$= 0.035$

$= 3.5\%$ //

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

$= \frac{90}{90 + 140}$

$= 0.391 = 39.1\%$ //

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

$= \frac{90}{90 + 210}$

$= 0.3$

$= 30\%$ //

5) Specificity = $TN / TN + FP$

$= 9650 / 9650 + 140$

$= 0.98.56$

$= 98.56\%$ //

MODULE 5 : CLUSTERING

* CLUSTER ANALYSIS :

↳ Cluster : a collection of data objects.
↳ Similar to one another within the same cluster.
↳ Dissimilar to the objects in other clusters.
↳ Cluster Analysis - Finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters.

* EXAMPLES OF CLUSTERING APPLICATIONS :

↳ Marketing
↳ Land use
↳ Insurance
↳ City-planning
↳ Earth-quake studies

* QUALITY : ~~WHAT~~ WHAT IS GOOD CLUSTERING ?

- A good clustering method will produce high quality clusters with
  ↳ High intra-class similarity
  ↳ Low inter-class similarity

* CLUSTERING

K-Means Clustering - (Steps) -

① Partition objects into k nonempty subsets.
② Compute seed points as the centroids of the clusters of the current partition.
③ Assign each object to the cluster with the nearest seed point.

④ Go back to step 2, stop when no more new assignment

Example:

① Explain K-means clustering and solve the following with K=2. {2,3,4,10,11,12,20,25,30} (one dimensional)

Soln:

Step 1: $M_1 = -4$, $M_2 = 12$ (We can take any random pts)

$K_1 = \{2,3,4\}$

$K_2 = \{10,11,12,20,25,30\}$

Step 2: New centroid points has to be calculated.

$M_1 = (2+3+4)/3 = 3$

$M_2 = (10+11+12+20+25+30)/6 = 18$

$\therefore M_1 = 3$

$M_2 = 18$

$K_1 = \{2,3,4,10\}$

$K_2 = \{11,12,20,25,30\}$

Step 3: Reassign the centroid

$M_1 = (2+3+4+10)/4 = 4.75$

$M_2 = (11+12+20+25+30)/5 = 19.6$

$\therefore M_1 = 4.75$

$\therefore M_2 = 19.6$

$K_1 = \{2,3,4,10,11,12\}$

$K_2 = \{20,25,30\}$

Step 4: Reassign the centroid

$M_1 = 7$ $M_2 = 25$

$K_1 = \{2,3,4,10,11,12\}$ $K_2 = \{20,25,30\}$

Step 5: Reassign the centroid
$M_1 = 7$, $M_2 = 25$ (Same as previous in step 4)

∴ We have similar means
Final clusters are $K_1 = \{2, 3, 4, 10, 11, 12\}$ & $K_2 = \{20, 25, 30\}$

(2) Cluster the given set of values $\{2, 3, 6, 8, 9, 12, 15, 18, 22\}$ into three clusters.

Soln:

Step 1: $M_1 = 2$, $M_2 = 3$, $M_3 = 6$ (Given)
$K_1 = \{2\}$, $K_2 = \{3\}$, $K_3 = \{6, 8, 9, 12, 15, 18, 22\}$

Step 2: $M_1 = 2$, $M_2 = 3$, $M_3 = 12.86$
$K_1 = \{2\}$ $K_3 = \{8, 9, 12, 15, 18, 22\}$
$K_2 = \{3, 6\}$

Step 3: $M_1 = 2$, $M_2 = 3$, $M_3 = 12.86$
$K_1 = \{2, 3\}$, $K_2 = \{6, 8, 9\}$, $K_3 = \{12, 15, 18, 22\}$

Step 4: $M_1 = 2.5$, $M_2 = 7.6$, $M_3 = 16.75$
$K_1 = \{2, 3\}$, $K_2 = \{6, 8, 9, 12\}$ $K_3 = \{15, 18, 22\}$

Step 5: $M_1 = 2.5$, $M_2 = 8.75$, $M_3 = 18.33$
$K_1 = \{2, 3\}$, $K_2 = \{6, 8, 9, 12\}$, $K_3 = \{15, 18, 22\}$

Final mean values are $M_1 = 2.5$, $M_2 = 8.75$, $M_3 = 18.33$
& final clusters are $K_1 = \{2, 3\}$ $K_3 = \{15, 18, 22\}$
$K_2 = \{6, 8, 9, 12\}$

* ALGORITHM FOR K-MEANS CLUSTERING METHOD -

Given K, the K-means algorithm is implemented in 4 steps :

1) Partition objects into k nonempty subsets.
2) Compute seed points as the centroids of the clusters.

Example.

①

Object	Attribute 1 (x):	Attribute 2 (y):
A	2	2
B	3	2
C	1	1
D	3	1
E	1.5	0.5

Euclidean Distance: $D(i,j) = \sqrt{|x_2 - x_1|^2 + |y_2 - y_1|^2}$

$C_1 = A(2,2)$, $C_2 = C(1,1)$

Distance Matrix -

$D^0 =$

A	B	C	D	E	Cluster	Centroid
0	1	1.41	1.41	1.51	C_1	GP_1
1.41	2.24	0	2	0.71	C_2	GP_2

$G^0 = (0<1.4)\ (1<2.24)\ (0<1.41)\ (1.41<2)\ (0.71<1.58)$

A	B	C	D	E	Cluster centroid
1	1	0	1	0	$C_1 - GP_1$
0	0	1	0	1	$C_2 - GP_2$

$K_1 = \{A, B, D\}$

$K_2 = \{C, E\}$

$[A(x) + B(x) + D(x)], [A(y) + B(y) + D(y)]$

Step 2: $C_1 = [(2+3+3)/3, (2+2+1)/3] = (2.67, 1.67)$

$C_2 = [(1+1.5)/2, (1+0.5)/2] = (1.25, 0.75)$

D^2 =	A	B	C	D	E	Cluster Centroid
	0.75	0.47	1.79	0.75	1.65	$C_1 - GP_1$
	1.45	2.25	0.32	1.76	0.35	$C_2 - GP_2$

G^1 =	A	B	C	D	E	Cluster centroid
	1	1	0	1	0	$C_1 - GP_1$
	0	0	1	0	1	$C_2 - GP_2$

$K_1 = \{A, B, D\}$ $K_2 = \{C, E\}$

$\therefore G^0 = G^1$

Final clusters are: $GP_1 = \{A, B, D\}$

$GP_2 = \{C, E\}$

(2) Group the following objects into K=2 group of medicine based on the two factors (ph & weight index).

Soln:

Object	Attribute 1 (x)	Attribute 2 (y)
Medicine A	1	1
B	2	1
C	4	3
D	5	4

$C_1 = A(1,1)$, $C_2 = B(2,1)$

$D^0 =$

A	B	C	D
0	1	3.6	5
1	0	2.83	4.24

$G^0 =$

A	B	C	D
1	0	0	0
0	1	1	1

$K_1 = \{A\}$, $K_2 = \{B, C, D\}$

Step 2: $C_1 = A(1,1)$

$C_2 = [(2+4+5)/3, (1+3+4)/3] = (3.67, 2.67)$

$D^1 =$

A	B	C	D
0	1	3.6	5
3.15	2.36	0.467	1.88

$G^1 =$

A	B	C	D
1	1	0	0
0	0	1	1

$K_1 = \{A, B\}$

$K_2 = \{C, D\}$

Step3: $C_1 = (1.5, 1)$

$C_2 = (4.5, 3.5)$

$D^2 =$	A	B	C	D
	0.5	0.5	3.20	4.61
	4.30	3.53	0.71	0.71

$G^2 =$	A	B	C	D
	1	1	0	0
	0	0	1	1

$K_1 = \{A, B\}$

$K_2 = \{C, D\}$

$\therefore G^1 = G^2$

Final clusters are - $GP_1 = \{A, B\}$

$GP_2 = \{C, D\}$

* PROBLEM OF K-MEANS METHOD

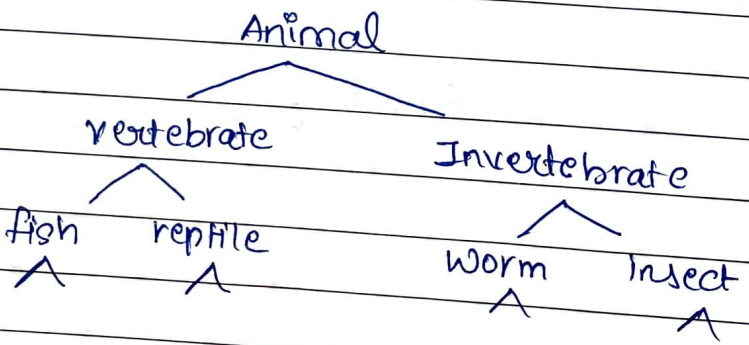
- K-means algorithm is very sensitive to outliers.
- K-Medoids - instead of taking the mean value of the object in a cluster as a reference point, medoids can be used, which is the most centrally located object in a cluster.

* K-MEDOID

- Name of the algorithm - PAM (Partitioning Around Medoid

* HIERARCHICAL CLUSTERING

- Build a tree-based hierarchical taxonomy (dendrogram) from a set of documents.
- Eg:



- Clustering obtained by cutting the dendogram at a desired level.
- Each connected components forms a cluster.

* HIERARCHICAL CLUSTERING ALGORITHMS

i) Agglomerative (bottom-up)
ii) Divisive (Top-bottom) (Not in syllabus)

AGGLOMERATIVE HIERARCHICAL CLUSTERING METHOD:

↳ Single Link algorithm
↳ Complete Link algorithm
↳ Average Link algorithm

* TYPICAL ALTERNATIVES TO CALCULATE THE DISTANCE BETWEEN CLUSTERS

↳ Single Link: Smallest distance between an element in one cluster and an element in the other.
i.e, $dis(K_i, K_j) = min(t_{ip}, t_{jq})$

↳ Complete Link: $dis(K_i, K_j) = max(t_{ip}, t_{jq})$

↳ Average: $dis(K_i, K_j) = avg(t_{ip}, t_{jq})$

Not imp ↳ Centroid: $dis(K_i, K_j) = dis(C_i, C_j)$

Not imp ↳ Medoid: $dis(K_i, K_j) = dis(M_i, M_j)$

* Example

①

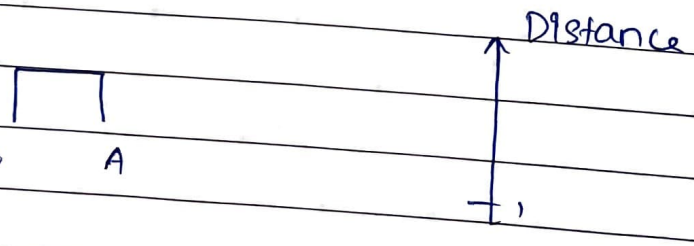
	E	A	C	B	D
E	0				
A	1	0			
C	2	2	0		
B	2	5	1	0	
D	3	3	6	3	0

Solve the following eg using Single Link Hierarchical clustering.

Soln: Step 1: Identify the two clusters with the shortest distance in the matrix, and merge ~~do~~ them together. Re-compute, the distance matrix, as those two clusters are now in a single cluster, (no longer exist by themselves).

By looking at the distance matrix above, we see that E & A have the smallest distance from all. So, we merge those two in a single cluster, & re-compute the distance matrix.

Dendogram



Distance Matrix

→ dist((EA), C) = MIN (dist(E, C), dist(A, C))
= MIN(2, 2)
= 2

→ D((EA), B) = MIN(D(E, B), D(A, B))
= MIN(2, 5)
= 2

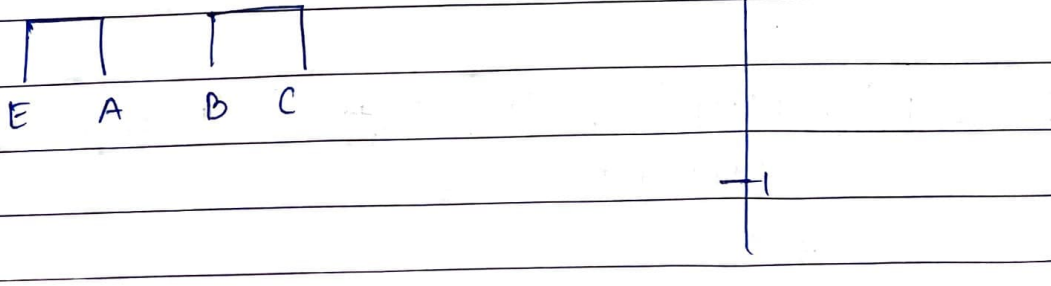
→ D((EA), D) = MIN(D(E, D), D(A, D))
= MIN(3, 3)
= 3

	E,A	C	B	D
E,A	0			
C	2	0		
B	2	1	0	
D	3	6	3	0

Step 2: Consider the distance matrix obtained in step 1 (given above)

∵ B,C distance is minimum, we combine B & C.

Dendogram



Distance Matrix

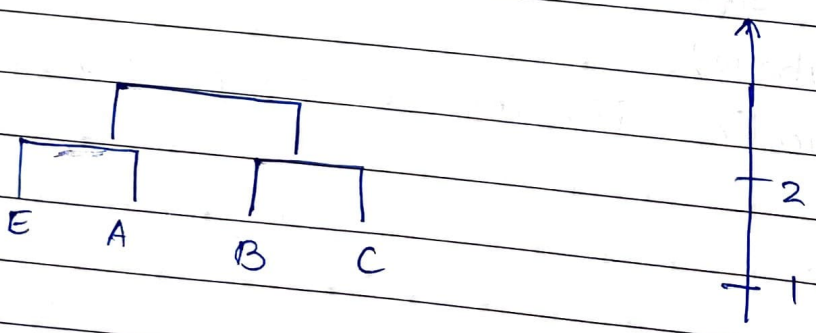
$D((B,C),(EA)) = MIN(D(BE), D(BA), D(CE), D(CA))$
$= MIN(2, 5, 2, 2)$
$= 2$

$D((BC),D) = MIN(D(B,D), D(CD))$
$= MIN(3, 6)$
$= 3$

	E,A	B,C	D
E,A	0		
B,C	2	0	
D	3	3	0

Step 3: Consider the distance matrix obtained in step 2 (given above)
∵ (E,A) & (B,C) distance in minimum, we combine them.

Dendogram

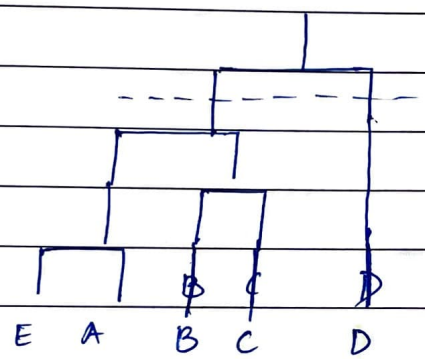


$D(EABC, D) = MIN(D(E,D), D(A,D), D(B,D), D(C,D))$
$= MIN(3, 3, 6, 3)$
$= 3$

	E,A,B,C	D
E,A,B,C	0	
D	3	0

Step 4 : Finally combine D with (EABC)

Final Dendogram

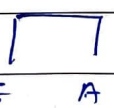


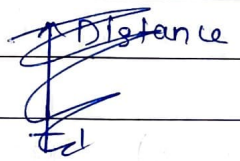
(2)

	E	A	C	B	D
E	0				
A	1	0			
C	2	2	0		
B	2	5	1	0	
D	3	3	6	3	0

Solve the following eg using complete link Hierarchical clustering.

Soln: Step 1:





$$D((EA), C) = MAX(D(E,C), D(A,C))$$
$$= MAX(2,2)$$
$$= 2$$

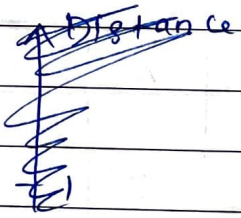
$D((EA), B) = MAX(D(E,B), D(A,B))$
$= MAX(2,5)$
$= 5$

$D((EA), D) = MAX(D(E,D), D(A,D))$
$= MAX(3,3)$
$= 3$

	E,A	C	B	D
E,A	0			
C	2	0		
B	5	1	0	
D	3	6	3	0

Step 2:

Dendogram



Distance Matrix

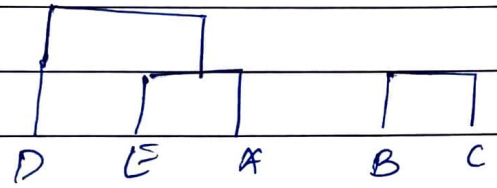
$D((B,C),(EA)) = MAX(D(BE), D(BA), D(CE), D(CA))$
$= MAX(2,5,2,2)$
$= 5$

$D(BC, D) = MAX(D(B,D), D(CD))$
$= MAX(3,6)$
$= 6$

	E,A	B, C	D
E,A	0		
B, C	5	0	
D	3	6	0

Step 3:

Dendogram



D(EAD, BC) = MAX (D(E,D), D(A,D), D(B,D), D(C,D)) (AB), (ED)

= MAX (2, 5, 6, 2, 2, 3)

= MAX = 6 //

	E,A,B,C	D
E,A,B,C	0	
D	6	0

Step 4: Finally combine D with (EA B C)

Final Dendogram