Cluster Analysis

2) Data Types in Cluster Analysis

3) Categories of clustering methods
- Partitioning method ← K-mean, K-medoids, CLARANS
- Hierarchical clustering
  - Agglomerative (bottom-up) — ~~CURE~~ Chameleon
  - Divisive (Top-down) BIRCH, Chameleon
- Density Based Methods
  - DBSCAN
  - OPTICS
- Grid-Based Methods
  - STING
  - CLIQUE
- Model Based method
  - Statistical approach
  - Neural N/W approach
  - Outlier analysis

Applications –
1) Pattern Recognition
2) Spatial Data Analysis
3) Image Processing
4) Market Research
5) Document classification

A good clustering method will produce high quality cluster with
- high intra-class similarity
- low inter-class —.

⇒ Requirements of clustering in Data Mining

1) Scalability
2) Ability to deal with different types of attributes.
3) Ability to handle dynamic data
4) Discovery of cluster with arbitrary shape
5) Able to deal with noise & outliers
6) High dimensionality
7) Interpretability & usability

http://www.rtuexam.net/College/pRiNtPrAcMaRks.php

Cluster Analysis: K-Mean Clustering

$A_1 \rightarrow (2, 10)$

$A_2 \rightarrow (2, 5)$

$A_3 \rightarrow (8, 4)$

$A_4 \rightarrow (5, 8)$

$A_5 \rightarrow (7, 5)$

$A_6 \rightarrow (6, 4)$

$A_7 \rightarrow (1, 2)$

$A_8 \rightarrow (4, 9)$

Initial cluster centers are: (3)

A_1 (2, 10)

A_4 (5, 8)

& A_7 (1, 2)

Point	(2,10)	(5,8)	(1,2)	Cluster
(2,10)	0	5	9	1
(2,5)	5	6	4	3
(8,4)	12	7	9	2
(5,8)	5	0	10	2
(7,5)	10	5	9	2
(6,4)	10	5	7	3
(1,2)	9	10	0	3
(4,9)	3	2	10	2

Cluster 1: (2,10)

Cluster 2: (8,4), (5,8), (7,5), (6,4), (4,9)

Cluster 3: (2,5), (1,2)

Centre of the clusters: (4)

1) Cluster 1 → (2, 10)
Cluster 2 → (6, 6) $\left[\frac{8+5+7+6+4}{5}, \frac{4+8+5+4+9}{5}\right]$
Cluster 3 → (1.5, 3.5)

(2, 10) (6, 6) (1.5, 3.5)

(2, 10)
(2, 5)
(8, 4)
(5, 6)
(7, 5)
(6, 4)
(1, 2)
(4, 9)

2nd Epoch →

C_1	(3, 9.5)	(A_1, A_8)
C_2	(6.5, 5.25)	(A_3, A_4, A_5, A_6)
C_3	(1.5, 3.5)	(A_2, A_7)

3rd epoch -

C_1	(3.66, 9)	(A_1, A_4, A_8)
C_2	(7, 4.33)	(A_3, A_5, A_6)
C_3	(1.5, 3.5)	(A_2, A_7)

K- medoid

$X_1 \rightarrow (2,6)$

$X_2 \rightarrow (3,4)$

$X_3 \rightarrow (3,8)$

$X_4 \rightarrow (4,7)$

$X_5 \rightarrow (6,2)$

$X_6 \rightarrow (6,4)$

$X_7 \rightarrow (7,3)$

$X_8 \rightarrow (7,4)$

$X_9 \rightarrow (8,5)$

$X_{10} \rightarrow (7,6)$

1) Initialize K - centers:

Let $K = 2$

$C_1 = (3,4)$, $C_2 = (7,4)$

i	Cost distance
1	3
2	4
3	4
4	4
5	5
6	3
7	5
8	
9	6
10	6

Cluster 1 = { (3,4), (2,6), (3,8), (4,7) }

for c2 (7,4)

i	Cost distance
1	7
3	8
4	6
5	3
6	1
7	1
9	2
10	2

Cluster 2 → { (7,4), (6,2), (6,4), (7,3) (8,5) (7,6) }

So the total cost is

⇒ 3+4+4+3+1+1+2+2

= 20.

Step 2: Select one of the nonmediods(O')

Let us assume O' = (7,3)

& Now mediods are

c1 (3,4) & O' (7,3)

total cost = 22 > 20

So moving towards O' is bad idea.

Hierarchical Clustering.

(7)

⇒ is used to group similar objects into clusters.

⇒ It is type of unsupervised clustering methods.

Step 1 — Given N×N distance matrix / Proximity matrix

	A	B	C	D	E	F
A	0	662	877	255	412	996
B	662	0	295	468	268	400
C	887	295	0	754	564	138
D	255	468	754	0	219	869
E	412	268	564	219	0	669
F	996	400	138	869	669	0

1) Single linkage Method (min)
2) Complete - Linkage (max)
3) Average - Linkage (avg.)

→ Agglomerative Hierarchical Clustering also called as it merges clusters iteratively.

→ Divisive
Rarely used

Use Min / Single Linkage Method: (8)

Step-2

	A	B	C/F	D	E
A	0	662	877	255	412
B	662	0	295	468	268
C/F	887	295	0	754	564
D	255	468	754	0	219
E	412	268	564	219	0

Step-3:

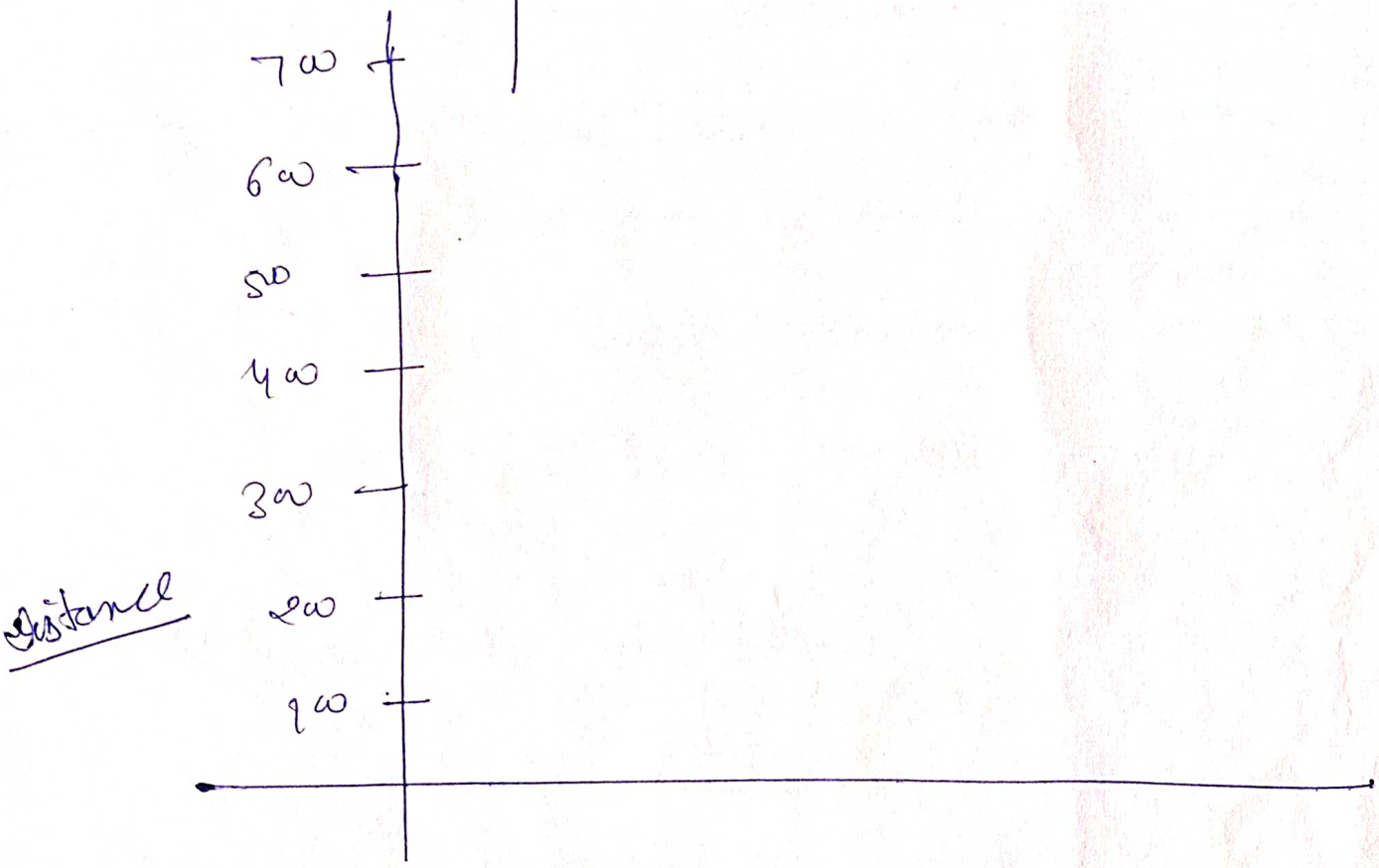
	A	B	C/F	D/E
A	0	662	877	255
B	662	0	295	268
C/F	887	295	0	564
D/E	~~412~~ ~~268~~ 255	268	564	0

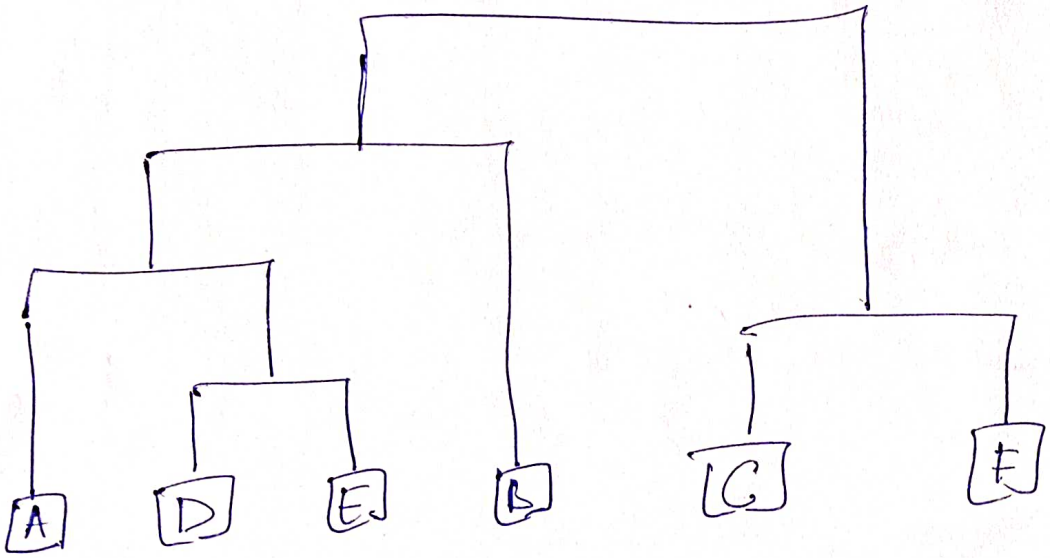
Step-4:

	A/D/E	B	C/F
A/D/E	0	~~268~~ 662	~~564~~ 564
B	~~662~~ 268	0	295
C/F	564	295	0

	A/B/D/E	C/F
A/B/D/E	0	295
C/F	295	0

(9)





Exp. No. Probabilistic Model Based Clustering (10)

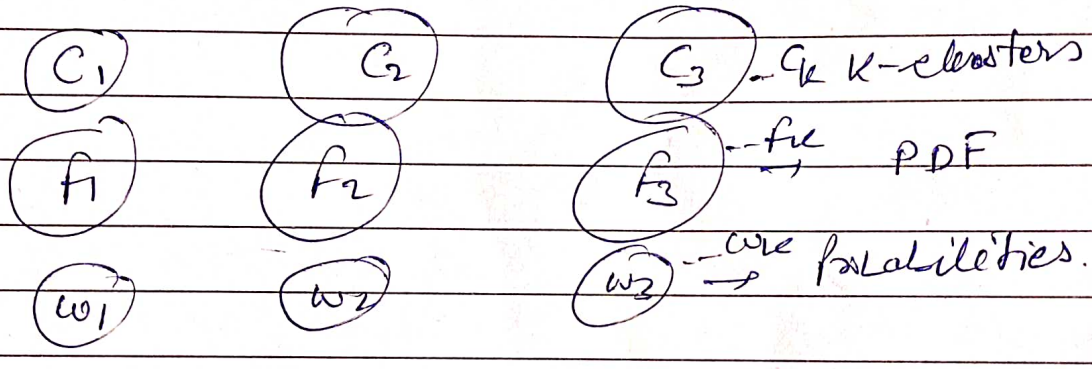
Page No.

Gaussian Mixture Model:

Given Data set → D

K → No. of clusters

The task of Probabilistic model based cluster analysis is to infer a set of probabilistic clusters that is most likely to generate D.



⇒ For an object O, the probability that O is generated by cluster C_j $(1 \le j \le k)$ is given by

$$P(O|C_j) = w_j f_j(O) \longrightarrow (1)$$

Therefore the probability that O is generated by the set C clusters is

$$P(O|C) = \sum_{j=1}^{k} w_j f_j(O) \longrightarrow (2)$$

DATE: / /

Teacher Signature

⇒ for a data set $D = \{o_1, o_2, o_3, \ldots o_n\}$ of n – objects, we have

$$P(D|C) = \prod_{i=1}^{n} P(o_i|C) = \prod_{i=1}^{n} \sum_{j=1}^{K} w_j f_j(o_i) \quad \text{---(3)}$$

⇒ Now it is clear that the task of probabilistic model-based cluster analysis on a data set D, is to find a set C of K probabilistic clusters such that $P(D|C)$ is maximized.

⇒ Let $o_1, o_2, \ldots o_n$ be n observed objects & $\theta_1, \theta_2, \ldots \theta_K$ be the parameters of k-distributions, denoted by

$$O = \{o_1, o_2, \ldots o_n\}$$

$$\Theta = \{\theta_1, \theta_2, \ldots \theta_K\} \text{ resp.}$$

⇒ Then for any object $o_i \in O$ $(1 \le i \le n)$ eqn (2) can be rewritten as

$$P(o_i|\Theta) = \sum_{j=1}^{K} w_j P_j(o_i|\theta_j)$$

DATE: / / Teacher Signature

where $P_j(o_i|\theta_j)$ is the probability that o_i is generated from j^{th} distribution using parameter θ_j, then eqn (3) can be rewritten as

$$P(O|\theta) = \prod_{i=1}^{n} \sum_{j=1}^{K} w_j P_j(o_i|\theta_j)$$

Univariate Gaussian Mixture Model:

⇒ Assume that PDF of each cluster follows a 1-D Gaussian distribution.

⇒ Let no. of clusters = K

⇒ Two parameters for the PDF of each cluster are center, μ_j & std. Dev. σ_j $(1 \le j \le K)$, we denote parameters as

$$\theta_j = (\mu_j, \sigma_j)$$

& $\theta = \{\theta_1, \theta_2, \ldots \theta_K\}$

⇒ Let the Data Set be $O = \{o_1, o_2, \ldots o_n\}$ where o_i $(1 \le i \le n)$ is a real number

⇒ for any point $o_i \in O$, we have

$$P(o_i|\theta_j) = \frac{1}{\sigma_j \sqrt{2\pi}} e^{-\frac{(o_i - \mu_j)^2}{2\sigma_j^2}}$$

DATE: / /
Teacher Signature

⇒ Assuming that each cluster has same probability that $\omega_1 = \omega_2 \cdots = \omega_K = 1/K$

$$\Rightarrow P(o_i|\theta) = \frac{1}{K} \sum_{j=1}^{K} \frac{1}{\sigma_j \sqrt{2\pi}} \cdot e^{\frac{(o_i - \mu_j)^2}{2\sigma_j^2}}$$

$$\Rightarrow P(O|\theta) = \frac{1}{K} \prod_{i=1}^{n} \sum_{j=} \frac{1}{\sqrt{2\pi}\sigma_j} e^{\frac{-(o_i-\mu_j)^2}{2\sigma^2}} \quad \hookrightarrow (4)$$

⇒ The task of probabilistic model-based cluster analysis using univariate Gaussian mixture model is to infer θ such that eqn (4) is maximized.

⇒ Using the EM algorithm for mixture models:

Given a set of objects $O = \{o_1, o_2, \cdots o_n\}$ we want to mine a set of parameters $\theta = \{\theta_1, \theta_2, \cdots \theta_K\}$ such that $P(O|\theta)$ is maximized.

where $\theta_j = (\mu_j, \sigma_j)$

j^{th} univariate Gaussian Distribution $(1 \le j \le K)$

EM Algo:

1) We assign random values to parameters θ as the initial values. We then iteratively conduct E-step & M-step until the parameters converge or the change is sufficiently small.

E-Step :→ i) for each object $O_i \in O (1 < i \leq n)$ we calculate probability that O_i belongs to each distribution that is

$$P(\theta_j | O_i, \theta) = \frac{P(O_i | \theta_j)}{\sum_{l=1}^{k} P(O_i | \theta_l)}$$

M-Step :→ We adjust the parameters θ so that the expected likelihood $P(O|\theta)$ in eqn (4) is maximized. This can be achieved by setting

$$\mu_j = \frac{1}{k} \sum_{i=1}^{n} O_i \frac{P(\theta_j | O_i, \theta)}{\sum_{i=1}^{n} P(\theta_j | O_i, \theta)}$$

$$\mu_j = \frac{1}{k} \frac{\sum_{i=1}^{n} O_i P(\theta_j | O_i, \theta)}{\sum_{i=1}^{n} P(\theta_j | O_i, \theta)}$$

DATE : / /

Teacher Signature

$$\sigma_j = \frac{\sum_{i=1}^{n} p(\theta_j | o_i, \theta)\,(o_i - \mu_j)^2}{\sum_{i=1}^{n} p(\theta_j | o_i, \theta)}$$

Exp. No.

16

Page No.

Probability Distribution Function (PDF)

PDF $\Rightarrow$ $P(a < x < b) = \int_a^b f(x)\,dx = \int_a^b \frac{1}{\sigma\sqrt{2\pi}} e^{\left[\frac{-(x-\mu)^2}{2\sigma^2}\right]} dx$

$x \rightarrow$ random variable

$\mu \rightarrow$ mean value

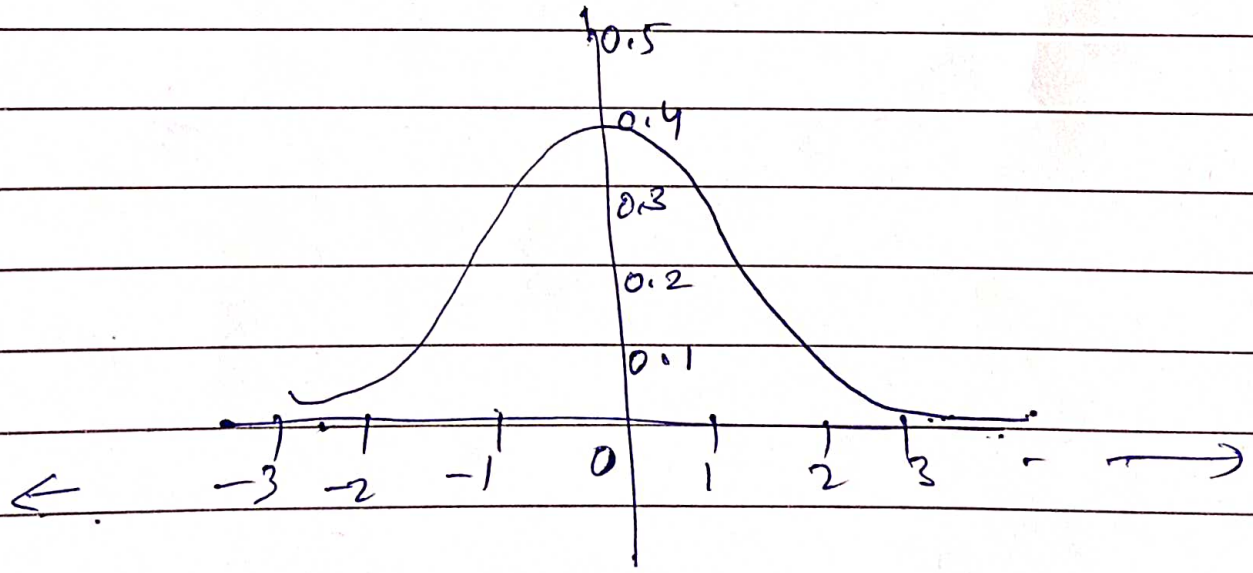
$\sigma \rightarrow$ std. devi.

Probability Density function for Normal Distribution

$$f(x; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)$$

if $\mu = 0$, $\sigma = 1$

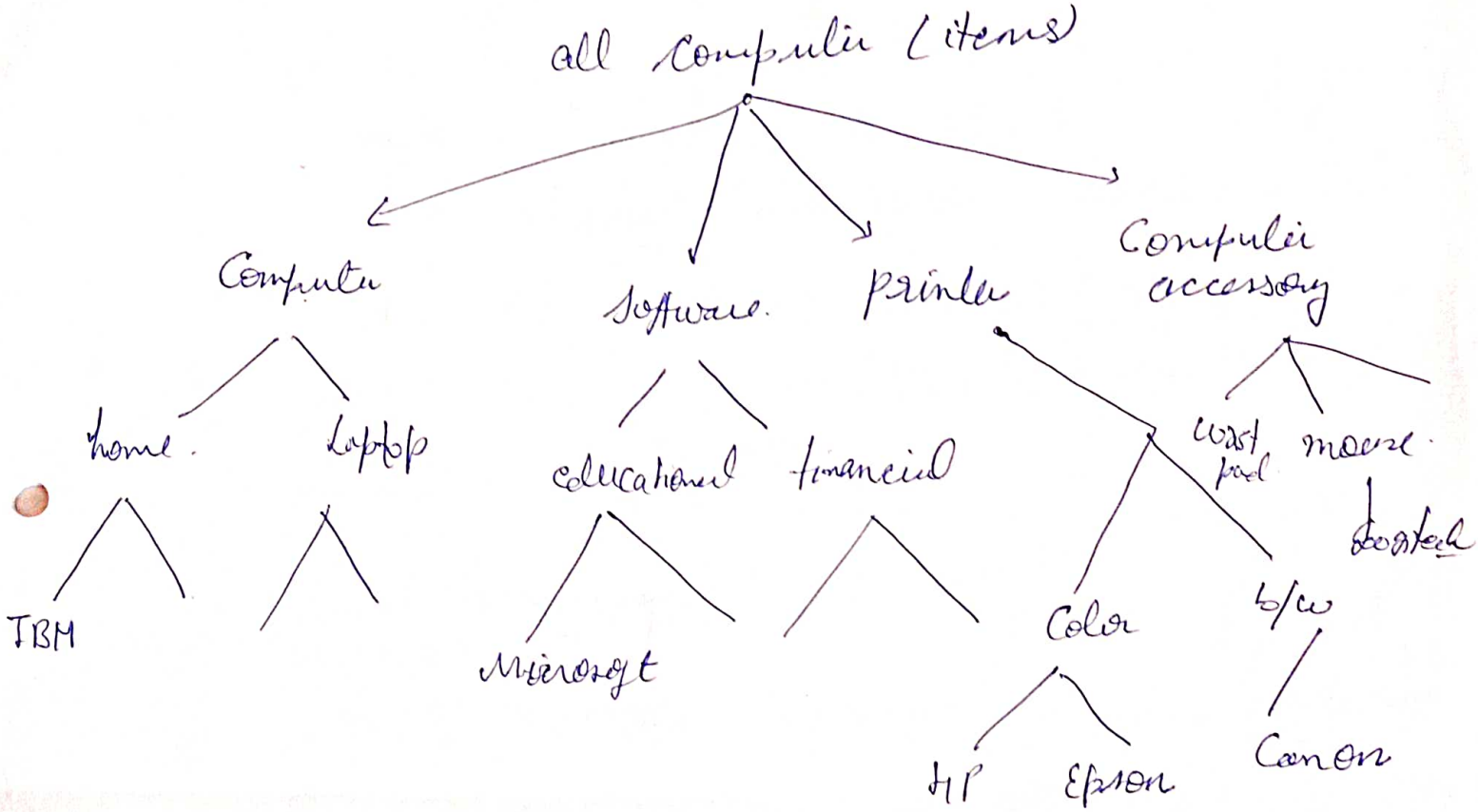
$$f(x; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-x^2}{2}\right)$$



DATE : / /

Teacher Signature

Association Rule:



1) ⟹ Single-Dimensional or Intra-Dimension Association Rules:

buys (X, "IBM home computer")
⇒ buys (X, "printer")
⇒ (buys (X, "sony b/w printer")
⇒ buys (X, "b/w printer")
⇒ - - - .

2) Multidimensional:

age (X, "19-24") ∧ occupation (X, "student")
⇒ buys (X, "Laptop").

⇒ MDAR with no repeated predicates are called: inter-dimension association rule.

⇒ with repeated predicates: called ⇒ hybrid-dimension association rules

age(X, "19-24") ∧ buys(X, "Laptop") ⇒ buys(X, "printer")

· Multilevel Mining:

1) Multilevel Mining with uniform support:

Level-1, min sup = 5% — Computer [support = 10%]

Level-2, min sup = 5% — Laptop [support = 6%] — home computer [support = 4%]

2) Multilevel Mining with reduced support:

Level-1 [min support = 5%]

Level-2 [min support = 3%]

⇒ Rules generated from association rule mining with concept hierarchies are called multilevel association rule.

⇒ Approaches to Mining multilevel association rules:

1). Using uniform min support for all levels :- (

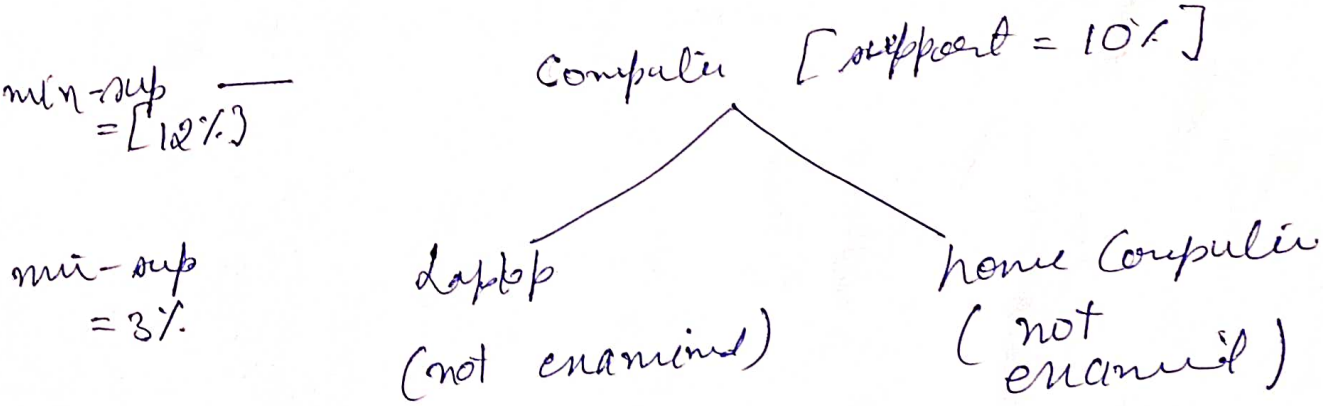
2) Using reduced min support at lower levels:

⇒ For mining MLA with reduced support, alternative search strategies are:

a) Level-by-Level independent:

→ each node is examined, regardless of whether or not its parent node is found to be frequent.

b) level-cross filtering by single item:



=> An item at i^{th} level is examined if & only if its parent node at $(i-1)^{th}$ level is frequent.

c) Level-across filtering by K-itemset. (20)

min-sup = 5%.

Computer & printer [support = 7%]

min sup = 2%

Laptop Computer & b/w printer [support = 1%]

Laptop Computer & color printer [support = 2%]

Home Computer & b/w printer [support = 1%]

Home Computer & color printer [support = 3%]

⇒ A K-itemset at the i^{th}-level is examined if & only if it's corresponding parent K-itemset at the $(i-1)^{th}$ level is frequent.

⇒ Computer & printer is frequent

⇒ Same concept level.

⇒ different concept level.
(Cross-level association rules)

⇒ How can ~~cross association~~ cross-level-association be mined?

level-i
level-j → reduced min support

threshold of level j should be used overall so that items from level j can be included in the analysis.

⇒ Checking for Redundant multilevel associations rules:

⇒ Concept hierarchies are useful in data mining since they permit the discovery of knowledge at different levels of abstraction, such as multilevel association rules.

R1 ⇒ home computer ⇒ b/w printer [s = 8%, c = 70%]

R2 ⇒ IBM home computer ⇒ b/w printer [s = 2%, c = 7%]

Multidimensional association rules:

1) Single D or Intra-D
2) M D.
   - with no repeated predicates (inter-D)
   - with repeated predicates (hybrid-D)

Ex: age (X, "19-24") ∧ buys (X, "laptop")
⇒ buys (X, "b/w printer")

⇒ database attribute can be
⇒ Categorical & Quantitative

⇒ Techniques for mining MDAR can be categorized according to three basic approaches regarding the treatment of quantitative attributes:

1) Quantitative attributes are discretized using predefined concept hierarchies
   → This discretization occurs prior to mining
   Ex: Concept hierarchy for income may be used to replace the original numeric values of this attribute by ranges "0-20K", "21-30K" & so on.

→ Here discretization is static & predetermined
⇒ Mining MDAR using static discretization of quantitative attributes.

(2) Quantitative AR:

Quantitative attributes are discretized into "bins" based on distribution of data.

=> These bins may be further combined during mining process. The discretization process is dynamic.

3) Distance-Based AR: Quantitative attributes are discretized so as to capture the semantic meaning of such interval data. This dynamic discretization procedure considers the distance between data points.

Mining Frequent Itemsets:
(Apriori Algorithm)

Introduction: — In data mining, association rule learning is popular & well researched method for discovering interesting relations b/w variables in large databases.

Ex. Ex. rule {onion, potato} ⇒ {burger}
{milk, bread} ⇒ {butter}

⇒ In Computer Science & Data mining, Apriori is a classic algorithm for learning association rules.

Apriori Algo. → is an influential algo. for mining frequent itemsets for boolean association rules.

1) **Frequent Itemset:** — The sets of item which has min support (denoted by L_i or i^{th} - itemset)

2) **Apriori Property** — Any subset of frequent itemset must be frequent.

3) **Join operation** — To find L_k, a set of candidate k-itemsets is generated by joining L_{k-1} with itself.

Steps:- 1) Find the frequent itemsets: the sets of items that have min support.

→ A subset of a frequent itemset must also be a frequent itemset.

i.e. if {A B} is a frequent itemset, both {A} & {B} should be frequent itemset.

2). Iteratively find frequent itemsets with cardinality from 1 to K. (K-itemset)

3) Use the frequent itemsets to generate association rule.

Pseudo-Code :-

C_k : Candidate itemset of size K.

L_K : frequent itemset of size K.

L_1 = {frequent items};

for (K=1 ; L_K != ∅, K++) do begin

C_{K+1} = candidates generated from L_K.

for each transaction t in database do increment the counter of all candidates in C_{K+1} that are contained in t.

L_{K+1} = candidates in C_{K+1} with min-support

end

return $\cup_K L_K$;

Ex:

TID	List of items
T100	I_1, I_2, I_5
T100	I_2, I_4
T100	I_2, I_3
T100	I_1, I_2, I_4
T100	I_1, I_3
T100	I_2, I_3
T100	I_1, I_3
T100	I_1, I_2, I_3, I_5
T100	I_1, I_2, I_3

11/9

⇒ Total 9- transactions in database D.

⇒ Suppose min support count is 2.
or (2/9) = 22%

⇒ Let min confidence required is 70%.

⇒ find out the freq. itemset using Apriori Algo.

⇒ Then association rules will be generated using min. support & min confidence.

Step-1 : Generating 1-itemset Freq. Pattern

Item	supp. count
$\{I_1\}$	6
$\{I_2\}$	7
$\{I_3\}$	6
$\{I_4\}$	2
$\{I_5\}$	2

C_1

$\Rightarrow$ Compare Candidate support count with min support count.

Itemset	Sup. Count
$\{I_1\}$	6
$\{I_2\}$	7
$\{I_3\}$	6
$\{I_4\}$	2
$\{I_5\}$	2

L_1

{set of frequent 1-itemset}.

Step 2 : generating 2-itemset freq. patterns

Generate C_2 candidates from L_1 →

Itemset	Supp Count
$\{I_1, I_2\}$	4
$\{I_1, I_3\}$	4
$\{I_1, I_4\}$	1
$\{I_1, I_5\}$	2
$\{I_2, I_3\}$	4
$\{I_2, I_4\}$	2
$\{I_2, I_5\}$	2
$\{I_3, I_4\}$	0
$\{I_3, I_5\}$	1
$\{I_4, I_5\}$	0

C_2

⇓

Compare Candidate support count with min support count

⇓

Itemset	supp. count
$\{I_1, I_2\}$	4
$\{I_1, I_3\}$	4
$\{I_1, I_5\}$	2
$\{I_2, I_3\}$	4
$\{I_2, I_4\}$	2
$\{I_2, I_5\}$	2

L_2

· Step-3: Generating 3-itemset freq. Pattern.

Itemset	Sup-Count
$\{I_1, I_2, I_3\}$	2
$\{I_1, I_2, I_5\}$	2

$\Rightarrow$

Itemset	Sup-Count
$\{I_1, I_2, I_3\}$	2
$\{I_1, I_2, I_5\}$	2

L_3

$\Rightarrow$ The generation of the set of Candidate 3-itemset C_3, involves use of the Apriori property.

→ In order to find C_3, we compute L_2 join L_2.

$\Rightarrow C_3 = L_2 \text{ join } L_2 = \{ \{I_1, I_2, I_3\}, \{I_1, I_2, I_5\}, \{I_1, I_2, I_4\}, \{I_1, I_3, I_5\}, \{I_2, I_3, I_4\}, \{I_2, I_3, I_5\}, \{I_2, I_4, I_5\} \}$

→ Now, Join step is complete and Prune step will be used to reduce the size of C. Prune step helps to avoid heavy computation due to large C_K.

⇒ Based on Apriori Algo. property that all subsets of a freq. itemset must also be frequent.

for Example: Let take

$\{I_1, I_2, I_3\}$ ⇒ The 2-item subset of it are $\{I_1, I_2\}$, $\{I_1, I_3\}$ & $\{I_2, I_3\}$. Since all 2-item subsets of $\{I_1, I_2, I_3\}$ are members of L_2, we will keep $\{I_1, I_2, I_3\}$ in C_3.

⇒ Lets take another example of $\{I_2, I_3, I_5\}$

Subset: $\{I_2, I_3\}$, $\{I_3, I_5\}$, $\{I_2, I_5\}$ ↓ is not in L_2, hence its is not frequent. Thus we will have to remove $\{I_2, I_3, I_5\}$ from C_3.

⇒ Therefore: $C_3 = \{\{I_1, I_2, I_3\}, \{I_1, I_2, I_5\}\}$

Step-4 : Generating 4-itemset frequent Pattern.

⇒ $C_4 = L_3$ join L_3

$= \{\{I_1, I_2, I_3, I_5\}\}$

⇒ this itemset is pruned, since its subset $\{I_2, I_3, I_5\}$ is not frequent.

⇒ Thus $C_4 = \phi$ & algo. terminates.

⇒ Now, these frequent itemsets will be used to generate strong association rules (where strong associalen rules satisfy both min support & min. confidence)

Step-5 : Generating Associalen Rules from Frequent Itemsets :

⇒ for each frequent itemset "l" generate all nonempty subsets of l.

⇒ for each nonempty subset s of l, output the rule "$s \rightarrow (l - s)$" if

support_count (l) / support_count (s) >= min-conf.

Ex:

We had $L = \{ \{I_1\}, \{I_2\}, \{I_3\}, \{I_4\}, \{I_5\}, \{I_1, I_2\}, \{I_1, I_3\}, \{I_1, I_5\}, \{I_2, I_3\}, \{I_2, I_4\}, \{I_2, I_5\}, \{I_1, I_2, I_3\}, \{I_1, I_2, I_5\} \}$

Lets take $I = \{I_1, I_2, I_5\}$

⇒ its all nonempty subsets are:

$\{ \{I_1, I_2\}, \{I_1, I_5\}, \{I_2, I_5\}, \{I_1\}, \{I_2\}, \{I_5\} \}$

⇒ Let min. Confidence is 70%

⇒ The resulting association rules are:

⇒ R1: $I_1 \wedge I_2 \rightarrow I_5$

⇒ Confidence = $sc\{I_1, I_2, I_5\} / sc(I_1, I_2)$

$= 2/4 = 50\%$

R_1 is rejected.

⇒ R2: $I_1 \wedge I_5 \rightarrow I_2$

Confidence = $sc\{I_1, I_2, I_5\} / sc(I_1, I_5)$

$= 2/2 = 100\%$.

R_2 is selected.

R_3 : $I_1 \rightarrow I_2 \wedge I_5$

confidence = $sc\{I_1, I_2, I_5\} / sc\{I_1\}$

= 2/6 = 33%.

R_3 is rejected.

R_4 : $I_2 \rightarrow I_1 \wedge I_5$

confidence = $sc\{I_1, I_2, I_5\} / sc\{I_2\}$

= 2/7 = 29%.

R_4 is rejected

R_5 : $I_5 \rightarrow I_1 \wedge I_2$

Confidence = $sc\{I_1 I_2 I_5\} / sc(I_5)$

= 2/2 = 100%.

R_5 is selected.

$\Rightarrow$ In this way we can found strong association rules.

FP Growth:

$SC = 3$

TID	Items
1	E, A, D, B
2	D, A, C, E, B
3	C, A, B, E
4	B, A, D
5	D
6	D, B
7	A, D, E
8	B, C

Priority

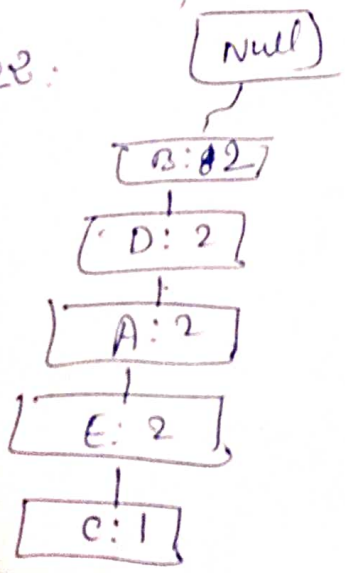
A → 5 → 3
B → 6 → 1
C → 3 → 5
D → 6 → 2
E → 4 → 4

B:6, D:6, A:5, E:4, C:3

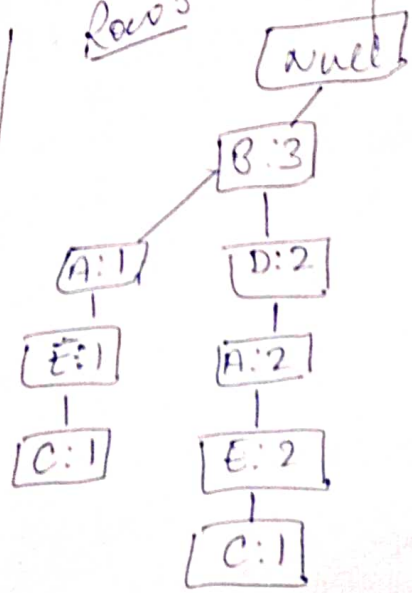
TID	Items	ordered Items
1	E, A, D, B	B, D, A, E
2	D, A C, E, B	B, D, A, E, C
3	C, A, B, E	B, A, E, C
4	B, A, D	B, D, A
5	D	D
6	D, B	B, D
7	A, D, E	D, A, E
8	B, C	B, C

FP-Tree

Row 2:

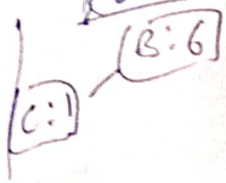


Row 3



Row 4:

B:4
D:3
A:3



Row 8:

Row 5

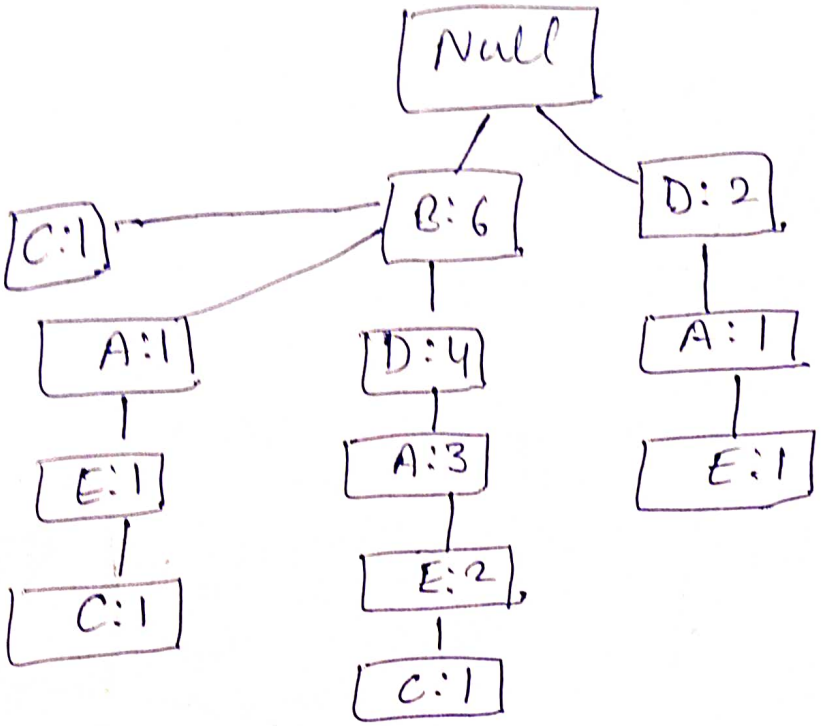
Row 6:

B:5
D:4

D:2
A:1
E:1

Row 7

Final Tree



Gaussian Mixture Models:

1) Probabilistic clustering methods assign instances to clusters probabilistically.

2) The foundation for probabilistic clustering is a statistical model called mixture model.

→ A mixture model is a set of K-probability distributions, representing K-clusters that govern the attribute values for members of that cluster.

→ GMM → Model → is a set of Gaussian (or normal) distributions with mean & covariances.

→ The clustering problem is to take a set of instances & a predefined no. of clusters (Gaussian distributions), & work out each cluster's mean & covariance, & population distribution b/w the clusters.

→ Let $x = \{x_j ; j = 1, 2, \ldots n\}$ be an n-D vector to be modelled using Gaussian mixture distribution

→ Let us assume that the model has K-subclasses (clusters).

→ Then the following parameters are required to completely specify the k^{th} subclass, $k = 1, \ldots K$

→ w_k → the probability that a data sample $x^{(i)}$, $i = 1, 2, \ldots n$, belongs to subclass k, $w_k \geq 0$ with $\sum_{k=1}^{K} w_k = 1$

→ This parameter gives population distribution b/w clusters; $\omega_R = N_R / N$;

where N_R = No. of samples belonging to R^{th} cluster.

μ_R = the n-D mean vector for subclass k.

Σ_R = the $n \times n$ covariance matrix for subclass k.

1) Mixture Model: is a model comprised of an unspecified combination of multiple probability distribution functions.

2) GMM :- is a mixture model that uses a combination of Gaussian (Normal) probability distribution and requires the estimation of the mean & std. deviation parameters for each.

⇒ There are many techniques for estimating the parameters of GMM, although a maximum likelihood is most common.

3) Expectation Maximization Algo :- or EM Algo.

is an approach for max. likelihood estimation in the presence of latent variables.

4) Latent variables :- missing / hidden / unobserved variables.

E-Step :- Estimate the expected value for each latent variable.

M-Step :- optimize the parameters of the distribution using maximum likelihood.

Teacher's Signature

DATE: / /
PAGE NO.:

5) Max. likelihood :- Estimation involves treating the problem as an optimization or search problem, where it seek a set of parameters that result in the best fit for the joint probability of the data sample.

6) Density Estimation :- involves selecting a probability distribution function & the parameters of that distribution that best explain the joint probability distribution of observed data.

Example of Gaussian Mixture Model:

Problem :- Dataset where points are generated from one of two Gaussian Processes. The points are 1-D.

The mean of the first distribution is $\mu_1 = 20$

second $\mu_2 = 40$

both distributions have same

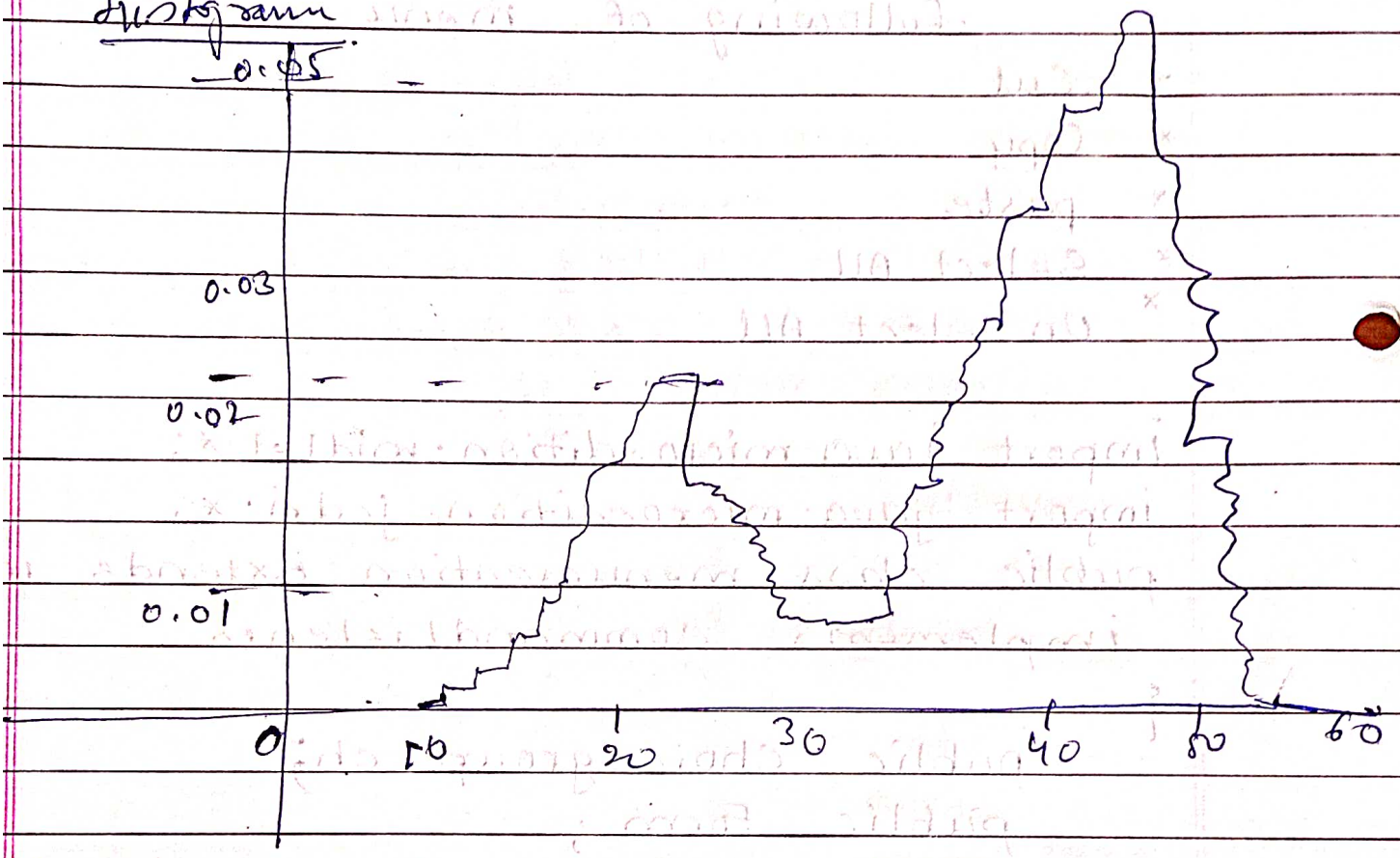
std. dev. = 5

first process → no. of points → 3000

second process → no. of points → 7000

Teacher's Signature

⇒ Mix them together.



Q. ⇒ Many of the points in the middle of two peaks that it is ambiguous as to which distribution they were drawn from.

⇒ Model the problem of estimating the density of this dataset using GMM.

⇒ GMM → can be used to model this problem & estimate the parameters of the distributions using EM-Algo.

Teacher's Signature

EM Coin Flip Example:

H T T T H H T H T H
H H H H T H H H H H
H T H H H H H T H H
H T H T T T H H T T
T H H H T H H H T H

1) Initialise θ_A & θ_B to
Let $\theta_A = 0.6$
$\theta_B = 0.5$

2) Compute a probability distribution of possible completions of the data using current parameters.

Set 1: H T T T H H T H T H

what is the probability that observe 5 heads & 5 tails in Coin A & Coin B

⇒ Computer likelihood of set 1 coming from Coin A or B using Binomial distribution with mean probability θ on n- trials with K- success.

$$P(K) = \binom{n}{k} \theta^K (1-\theta)^{n-K}$$

⇒ likelihood of A → 0.00079
" " B → 0.00097

Teacher's Signature