Dimensionality Reduction

To obtain the accurate result, we tend to add as many features as possible at first. However, after a certain point, the performance of the model will decrease with the increasing number of elements. This phenomenon is refered as 'Curse of Dimensionality'. ⇒

When we keep adding features, without increasing the number of training samples, dimensionality of feature space grows, which leads to overfitting. Overfitting will work well for training dataset and fails on future data.

One solution to having many features and few training samples, our approch is dimensionality reduction.

→ Advantages of dimensionality reduction is reduced memory and computation,

→ Save the cost of extracting it.

There are two main methods for reducing the dimensionality.

a) Feature selection:- is to select a subset of the original features for use in the machine learning model. Remove redundant irrevalent features without incurring much loss of information.

b) Feature extraction:- we are interested in finding a new set of K dimensions that are combination of original K dimension. The best known feature extraction methods are Principal Component Analysis (PCA) and LDA (linear discriminant Analysis)

Subset selection :-

Subset selection :- Finding the best subset of the set of features. The best subset contains the least number of dimensions that most contribute to accuracy. We discard the remaining ~~[illegible]~~ unimportant dimensions.

There are 2^d possible subset of d variables :

Forward selection :- We start with no variables and add them one by one, at each step adding the one that decreases the error the most, until any further addition does not decrease the error.

Backward selection :- We start with all variables, and remove them one by one, at each step removing the one that decreases the error the most until any further removal increases the error significantly.

With more features, generally we have lower training error, but not necessarily low validation error.

Sequential forward selection

Input $Y = \{y_1, y_2, \ldots y_d\}$

SFS algm takes the d-dimensional features as input

Output: $X_k = \{x_j \mid j = 1, 2 \ldots k, x_j \in Y\}$

SFS returns a subset of features, the number of selected features k, where $k < d$, has to be specified apriori.

Initialization :- $X_0 = \phi$ $k = 0$

. We initialize the algm with an empty set ϕ.

$\cancel{x^+ = \arg\max J(x_k + x)}$ $x^+ = \max \sigma(x_k + x, x_k)$

$x_{k+1} = x_k + x^+$

$k = k+1$

- We add an additional feature x^+ to our feature subset x_k
- x^+ is the feature that maximizes our criterion function i.e the feature that is associated with the best classifier performance (less error) if it is added to x_k.
- We repeat this procedure until the termination criterion is satisfied.

Termination $k = p$. where p is the desired features.

It is also called wrapper approach.

Local search procedure :-

Eg: we have Iris data set. we use twenty for training and thirty for validation.

we use nearest mean as classifier.

→ we start with one (single) feature :- we perform validation accuracy → which is 0.76, 0.~~8~~, 0.92, 0.94, we select fourth feature F_4 as our first feature fig(5)

→ we check whether adding another feature leads to improvement. as shown in fig (6)

validation accuracies are 0.87, 0.92, 0.96.

F_1, F_4 F_2, F_4 F_3, F_4.

→ and so on.

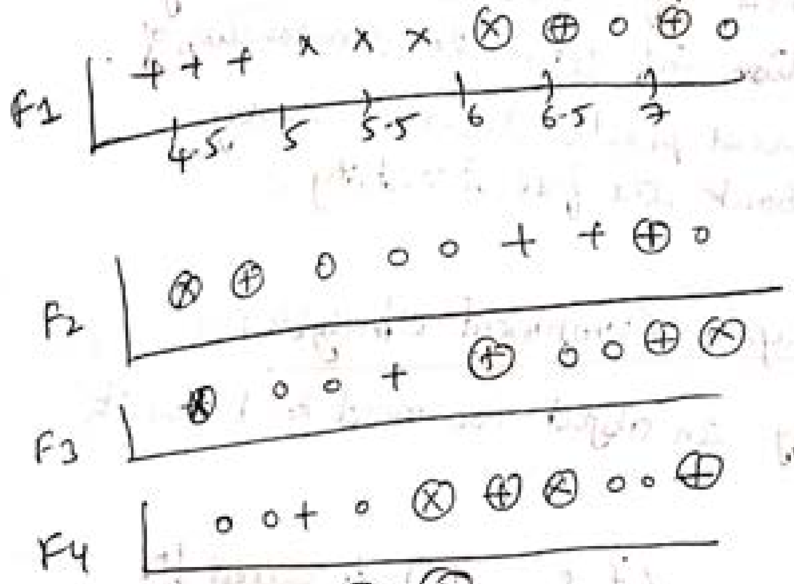


Fig (a)

→ Features we select depend on classifier we use;
→ Instead on small datasets, we have to work on large datasets

~~Sequential selection~~ Floating search: The number of added features and removed features can also change at each step

Sequential backward selection: We start with F with all features and remove one attribute from F ~~as opposed to adding it~~, we remove one by one attribute that causes the least error.

$j = \arg\max E(F - x_i)$

remove x_j from F if $E(F - x_j) < E(F)$

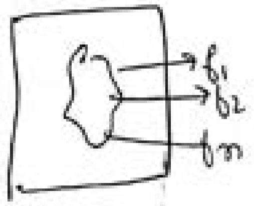
Error

We stop removing the features if it does not decrease the error

In face recognition, feature selection is not a good method for dimensionality reduction because, individual pixels by themselves do not carry much discrimination inf. It is the combination of values of several pixels together carry information about the face identity.

PCA (Principal Component Analysis)

To identify an object we need to know its features.



$(f_1, f_2 \ldots f_n)$ is called the feature pattern.

How many features to select?
Which features to select?

As the number of features increases the training samples should also increase.

Datapoints represented on 2D

Properties

- PCA is an unsupervised algm and it is a popular dimensionality reduction algm.
- PCA is used to find the patterns in data and detect the correlation b/w variables.
  If high correlation is found b/w variables, we can reduce the dimensionality using PCA.
  If high correlation means there is a redundant data.
- Principal components are orthogonal (independent to each other). It is used to find the dimensions with most variations.

Eg: Based on wt & ht classify the girls and boys.

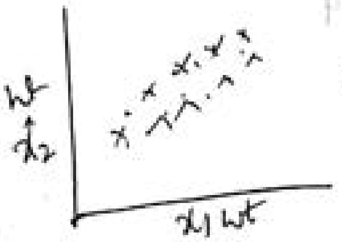


fig (b)

when we apply PCA → converts 2D to 1D

i.e reduce the two dimensional features to one dimensional.

fig (c)

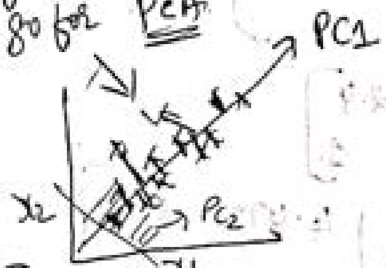
If we choose only weight feature, as shown in fig (c)

all the data points are close to each other, and with

single feature weight, not able to classify all the data points

So we go for PCA

In PCA



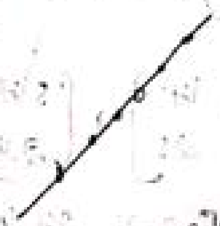


fig (d)

In PCA, finds the dimension which is used to represent

all the data points individually.

Fig (d) shows the chosen line which high variance

and all data points are adequently seperate.

This chosen line is first principal component (PC_1)

⇒ steps (1) compute the ~~covariance~~ variance covariance matrix C.

$$C = \frac{1}{N-1}\sum (x-\bar{x})(x-\bar{x})^T$$

(2) Find eigen values and corresponding eigen vectors

$$Ax = \lambda x \Rightarrow \lambda_1 \lambda_2$$

(3) Eigen vector corresponding to largest eigen value is called Principal component.

X	Y
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2	2.7
2	1.6
1	1.1
1.5	1.6
1.1	0.9

Step 1:- Substract the mean from the corresponding data component to recentre the data set.

$x - \bar{x}$: 0.69 −1.31 0.39 0.09 1.29 0.49 0.19 −0.81 −0.31 −0.

$y - \bar{y}$: 0.49 −1.21 0.99 0.29 1.09 0.79 −0.31 −0.8 −0.31 −

Step 2 ~~Compute the sample covariance Variance covariance matrix~~

$$C = \frac{1}{N-1} \sum (x-\mu)(x-\mu)^T$$

$$= \begin{pmatrix} 0.69 \\ 0.49 \end{pmatrix} (0.69 \; 0.49) =$$

- Once the eigen vectors are found from covariance matrix, next step is to order them by eigen values highest to lowest. This gives you the components in order of significance.

- Choose the eigen vector which has largest eigen value, choose the next eigen vector which has next largest eigen value and soo on.

x	2.5	0.5	2.2	1.9	3.1	2.3	2	2	1.5	1.9
y	2.4	0.7	2.1	2.2	3.0	2.7	1.6	1.1	1.6	0.9

$= 1.81$

$= 1.91$

PCA

$$C = \begin{bmatrix} Cov(x,x) & Cov(x,y) \\ Cov(y,x) & Cov(y,y) \end{bmatrix}$$

Step 1: Find the covariance matrix

$$Cov(x,y) = \sum_{i=1}^{n} \frac{(x_i-\bar{x})(y-\bar{y})}{n-1} \quad \text{or} \quad \frac{1}{n-1}(X)(X)^T$$

	x	$x-\bar{x}$	$(x-\bar{x})(x-\bar{x})$
①	2.5	0.69	0.476
②	0.5	−1.31	1.7161
⋮	⋮	⋮	⋮
⑩			$\frac{5.5490}{9}$

y	$y-\bar{y}$	$(y-\bar{y})(y-\bar{y})$
2.4	0.49	0.2401
0.7	−1.21	1.4641
⋮	⋮	⋮
sum		$\frac{6.449}{9}$

	x	y	$x-\bar{x}$	$y-\bar{y}$	$(x-\bar{x})(y-\bar{y})$
①	2.5	2.4	0.69	0.49	0.338
②	0.5	0.7	−1.31	−1.21	1.5851
⋮	⋮	⋮	⋮	⋮	⋮
⑩				sum =	$\frac{5.5350}{9}$

$$Cov(x,y) = \begin{bmatrix} 0.6165 & 0.6154 \\ 0.6154 & 0.7165 \end{bmatrix}$$

Step 3 Find the Eigen values

$$\boxed{|C - \lambda I| = 0}$$

$$\begin{bmatrix} 0.6165 & 0.6154 \\ 0.6154 & 0.7165 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = 0$$

$$\begin{vmatrix} 0.6165-\lambda & 0.6154 \\ 0.6154 & 0.7165-\lambda \end{vmatrix} = 0$$

$$(0.6165-\lambda)(0.7165-\lambda) - (0.6154)^2 = 0$$

Solve using quadratic Eqn $\lambda = \frac{-b \pm \sqrt{b^2-4ac}}{2a}$

$$\lambda_1 = 0.4908$$
$$\lambda_2 = 1.2840$$

Step 4 Finding the ith Eigen vector is done by

$$|C - \lambda I| [X] = 0$$

Let $\lambda_1 = 0.4908$

$$= \begin{bmatrix} 0.6165-0.4908 & 0.6154 \\ 0.6154 & 0.7165-0.4908 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = 0$$

$$= \begin{bmatrix} 0.1257 & 0.6154 \\ 0.6154 & 0.2257 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 0$$

$$= 0.1257x + 0.6154y = 0$$
$$= 0.6154x + 0.2257y = 0$$

$$\begin{matrix} x = \\ y = \end{matrix} \begin{bmatrix} -0.735 \\ -0.677 \end{bmatrix} \rightarrow \text{Eigen vector for } \lambda_1 = 0.4908$$

$\lambda_2 = 1.2840$

$$= \begin{bmatrix} 0.6165 - 1.2840 & 0.6154 \\ 0.6154 & 0.7165 - 1.2840 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 0$$

$$= \begin{bmatrix} 0.6675 & 0.6154 \\ 0.6154 & +0.5674 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 0$$

$x^2 + y^2 = 1$ — ①

$0.6675x + 0.6154y = 0$

$0.6154x - 0.5674y = 0 \rightarrow 0.6154x - 0.5674y = 0$

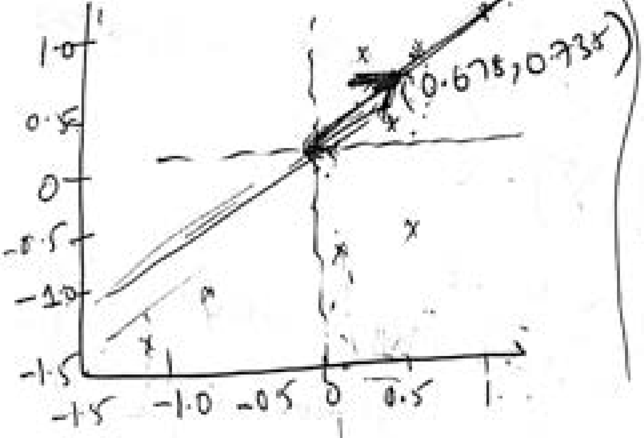
$0.6154x = 0.5674y$

$x = \frac{0.5674}{0.6154} y$

$x = 0.922y$ — ②

$(0.922y)^2 + y^2 = 1$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0.678 \\ 0.738 \end{bmatrix}$$



- Thus eigen vectors able to extract lines that characterise the data and gives the direction

Eg: PCA

X	Y
2	4
1	3
0	1
-1	0.5

x	y	$(x-\bar{x})$	$(y-\bar{y})$	$(x-\bar{x})(x-\bar{x})$	$(y-\bar{y})(y-\bar{y})$	$(x-\bar{x})(y-\bar{y})$

$$Cov(x,y) = \begin{bmatrix} Cov(x,x) & cov(x,y) \\ Cov(y,x) & Cov(y,y) \end{bmatrix}$$

$$cov(x,x) = \frac{1}{n-1}\sum_{i=1}^{n}(x-\bar{x})(x-\bar{x})$$

$$cov(x,y) = \frac{1}{n-1}\sum_{i=1}^{n}(x-\bar{x})(y-\bar{y})$$

$$Cov = \begin{array}{c|cc} & x & y \\ \hline x & 1.69 & 2.083 \\ y & 2.083 & 2.73 \end{array}$$

$$|C - \lambda I| = 0$$

$$\begin{bmatrix} 1.67 & 2.083 \\ & \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = 0$$

$$\begin{vmatrix} 1.67-\lambda & 2.083 \\ 2.083 & 2.73-\lambda \end{vmatrix} = 0$$

$$(1.67-\lambda)(2.73-\lambda) - (2.083)^2 = 0$$

$$\lambda^2 - 4.4\lambda + 0.2202 = 0$$

$$\lambda_1 = 4.3494$$

$$\lambda_2 = 0.0586$$

$\lambda_1 = 4.394$

$$[C - \lambda I]\begin{bmatrix} x \\ y \end{bmatrix} = 0$$

PCA

x	4	8	13	7
y	11	4	5	14

$\bar{x} = 8$

$\bar{y} = 8.5$

Step1: calculate the mean

$$\bar{x} = 8 \quad \bar{y} = 8.5$$

Step2: $C_2 = \begin{bmatrix} \mathrm{cov}(x,x) & \mathrm{cov}(x,y) \\ \mathrm{cov}(y,x) & \mathrm{cov}(y,y) \end{bmatrix}$

$$\mathrm{cov}(x,x) = \sum_{i=1}^{n} \frac{(x_i - \bar{x})(x_i - \bar{x})}{n-1}$$

$$\mathrm{cov}(x,y) = \sum_{i=1}^{n} \frac{(x_i - \bar{x})(y_i - \bar{y})}{n-1}$$

$$\mathrm{cov}(x,x) = \sum_{i=1}^{n} \frac{(4-8)^2 + (8-8)^2 + (13-8)^2 + (7-8)^2}{N-1}$$

$$= 14$$

$$\mathrm{cov}(x,y) = \sum_{i=1}^{n} (4-8)(11-8.5) + (8-8)(4-8.5) + (13-8)(5-8.5) + (7-8)(14-8.5)$$

$$= -11$$

$-+$	$++$
$--$	$+-$

$$\mathrm{cov}(y,x) = \mathrm{cov}(x,y) = -11$$

$$\mathrm{cov}(y,y) = \frac{1}{4-1}\left((11-8.5)^2 + (4-8.5)^2 + (5-8.5)^2 + (14-8.5)^2\right)$$

$$= 23$$

eg. PCA

x	y
3	4
1	3
0	2
−1	0.5

$$Cov(x,y) = \begin{bmatrix} 14 & -11 \\ -11 & 23 \end{bmatrix}$$

Step3:- Find the Eigen value

$$|C - \lambda I| = 0$$

$$\begin{bmatrix} 14 & -11 \\ -11 & 23 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = 0$$

$$\begin{vmatrix} 14-\lambda & -11 \\ -11 & 23-\lambda \end{vmatrix} = 0$$

$$(14-\lambda)(23-\lambda) - (-11 \times -11) = 0$$

$$\lambda^2 - 37\lambda + 201 = 0 \qquad \lambda = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$\lambda_1 = 30.38$$
$$\lambda_2 = 6.651$$

Step4- Find the eigen vector for largest Eigen Value

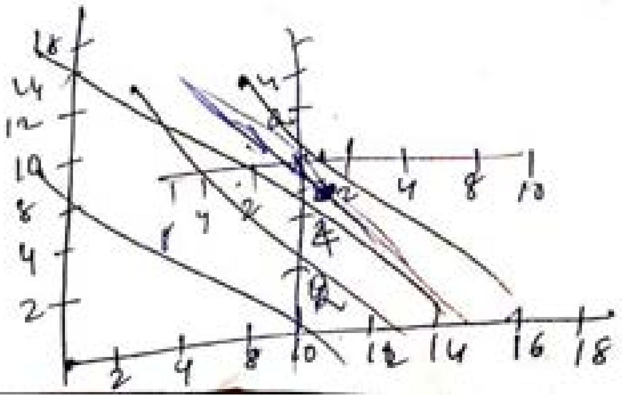
$$|C - \lambda I|[X] = 0$$

$$\lambda_1 = 30.38$$

$$= \begin{bmatrix} 14 - 30.38 & -11 \\ -11 & 23 - 30.38 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$= (14 - 30.38)x - 11y = 0$$
$$-11x + (23 - 30.38)y = 0$$

$$e_1 = \begin{bmatrix} 0.5574 \\ -0.8303 \end{bmatrix}$$



$$-16.8x = 11y$$
$$y = -\frac{16.8}{}$$

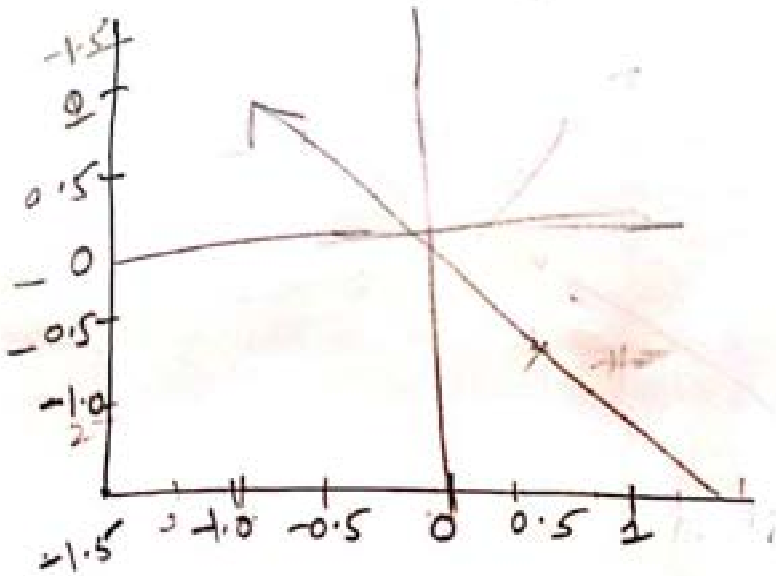
$-16.38x - 11y = 0$

$-11x - 7.38y = 0$

$$\frac{x}{+11} = \frac{y}{-16.38} = K \downarrow = 1$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} +11 \\ -16.38 \end{bmatrix}$$

$$= \begin{bmatrix} \frac{+11}{\sqrt{(-11)^2 + (-16.38)^2}} \\ \frac{-16.38}{\sqrt{(-11)^2 + (-16.38)^2}} \end{bmatrix} = \begin{bmatrix} +11/19.7 \\ \frac{-16.38}{19.7} \end{bmatrix} = \begin{bmatrix} +0.558 \\ -0.831 \end{bmatrix}$$



→ Consider the Two dimensional patterns (2,1) (3,5) (4,3) (5,6) (6,7) (7,8)

Compute the PCA

Clustering

Clustering is the process of grouping togeth data objects into clusters or groups so the Objects within a cluster have high similarity

→ Similarity is assessed based on attributes (features) of the object.

→ Similarity is measured by distance metrics [Euclidean distance]

K-means Clustering

→ Select randomly K cluster centers

→ Calculate the distance between each data point and cluster centers

→ Assign the data point to the cluster having a minimum distance from it and the cluster center i.e determining which sample belongs to which cluster

$$\underset{\text{Which cluster}}{K} = \underset{\text{Index}}{\text{argmin}} \, ||(x_i - \mu_i||^2$$

→ Recalculate the new cluster center.

→ Recalculate the distance between each datapoint and newly obtained cluster centers, and repeat step 3 & 4.

→ If no datapoint was recognised then stop.

: UTS - KN2050

Eg: kmeans: $k = \{2, 3, 4, 10, 11, 12, 20, 25, 30\}$

$\underline{k = 2}$

$m = 3$ $m = 18$

$k_1 = \{2, 3, 4, 10\}$ $k_2 = \{11, 12, 20, 25, 30\}$

$m_1 = 4.75$ $m_2 = 19.6$

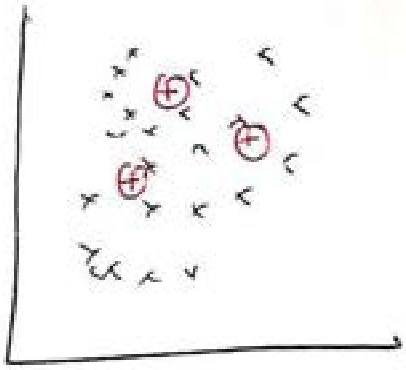
$k_1 = \{2, 3, 4, 10, 11, 12\}$ $k_2 = \{20, 25, 30\}$

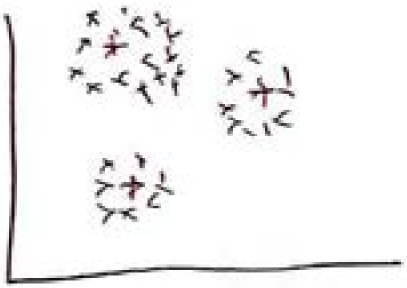
$m_1 = 7$ $m_2 = 25$

$k_1 = \{2, 3, 4, 10, 12\}$ $k_2 = \{20, 25, 30\}$

$m_1 = 7$ $m_2 = 25$

We are getting same mean we can stop.





K-Means Algm

	Height	Weight
1)	185	72
2)	170	56
3)	168	60
4)	179	68
5)	182	72
6)	188	77
7)	180	71
8)	180	70
9)	183	84
10)	180	88
11)	180	67
12)	177	76

K_1 (185, 72) K_2 (170, 56)

ED for 3 $K_1 = \sqrt{(168-185)^2 + (60-72)^2}$
$= 20.80$

$K_2 = \sqrt{(168-170)^2 + (60-56)^2}$
$= 4.48$

Person 3 datapoint is closer to cluster K_2, it is assigned to cluster K_2

New centroid calculation

for $K_2 = \left(\frac{170+168}{2}, \frac{60+56}{2}\right) = (169, 58)$

K_1 (185, 72) K_2 (169, 58)

...means

We have 4 document, as one training data point object and each document has 2 attributes. Each attribute represents coordinates of the object. Lets say the attributes are its word and count. We have to determine which document belongs to cluster 1 and which document belongs to other cluster

Object	Attribute: word x	Attribute: count y
D_1	1	1
D_2	2	1
D_3	4	3
D_4	5	4

$3^2 = 9 + 3^2$
$9+9=18$

K_1 (1, 1)

K_2 (2, 1)

Step 1. $Doc_3 \rightarrow K_1 = \sqrt{(4-1)^2 + (3-1)^2}$

$= \sqrt{3^2 + 2^2} = \sqrt{9+4} = \sqrt{13} = 3.6$

$K_2 = \sqrt{(4-2)^2 + (3-1)^2} = 2.82$

Document D_3 is nearest to K_2 so it is assigned to 2^{nd} cluster

$0.34+$
$= 0.1154$ 0.1156

Date : ....

k_1 (1,1) k_2 (3,2)

~~Since~~,

Since document 3 is assigned to cluster 2, we need to recalculate the new mean.

Cluster 2 will have new centroid = ~~(...)~~

$$k_2 = \left(\frac{2+4}{2}, \frac{1+3}{2}\right) = (6/2, 4/2) = (3, 2)$$

No we calculate the distance b/w each document & new centroids

Keep going until No document shifts its cluster

$$D4 \to k_2 \sqrt{(5-3)^2+(4-2)^2} = \sqrt{2^2+2^2} = \sqrt{8} = 2.8$$

$$k_1 = \sqrt{(5-1)^2+(4-1)^2} = \sqrt{4^2+3^2} = \sqrt{16+9} = \sqrt{27} = 5.1$$

So → D4 is nearer and belongs to $\underline{\underline{k_2}}$

Need to calculate the k_2 centroid once again.

$$k_2 = \left(\frac{2+4+5}{3}, \frac{1+3+4}{3}\right) = (3.66, 2.66)$$

I

	(1,1)	(2,1)	
D_1	0		1
D_2		0	2
D_3	3.6	2.82	2
D_4	5.1	4.2	2

II

	(1,1)	(3.66, 2.66)	cls
D_1	0	4.408	1
D_2	1	[illegible]5	1
D_3	[illegible]	3.135	2
D_4	5.1		2

III

	(1.5, 1)	(4.5, 3.5)
D_1	(0.5)	4.3
D_2	0.5	
D_3		
D_4		

12.25 6.25

K_1 $(1,1)$

K_2 $(3.66, 2.6)$

let $D_2 \rightarrow$ $K_1 = \sqrt{(2-1)^2+(1-1)^2} = \sqrt{1+0} = \sqrt{1} = 1$

Distance

$K_2 = \sqrt{(2-3.66)^2+(1-2.66)^2} = \sqrt{}$

$= \sqrt{2.75+2.756} = 4.408$

Now since D_2 is now nearer to Cluster 1 it is shifted to K_1.

<u>Object/document</u>

D_1 — Cluster 1

D_2 — cluster 1

D_3 — cluster 2

D_4 — cluster 2.

K-means clustering

(3 cf)

Example:

Cluster the following eight points (with (x,y)) into three clusters $A_1(2,10)$, $A_2(2,5)$, $A_3(8,4)$, $A_4(5,8)$, $A_5(7,5)$, $A_6(6,4)$, $A_7(1,2)$, $A_8(4,9)$

Soln:

Initial cluster centers are. $A_1(2,10)$ $A_4(5,8)$ & $A_7(1,2)$

Iteration 1

Point	1 (2,10) Dist mean1	2 (5,8) Dist mean2	3 (1,2) Dist mean3	Cluster
A_1 (2,10)	0	5	9	1
A_2 (2,5)	5	6	4	3
A_3 (8,4)	12	7	9	2
A_4 (5,8)	5	0	10	2
A_5 (7,5)	10	5	9	2
A_6 (6,4)	10	5	7	2
A_7 (1,2)	9	10	0	3
A_8 (4,9)	3	2	10	2

Cluster 1

$A_1(2,10)$

Cluster 2

$A_3(8,4)$
$A_4(5,8)$
$A_5(7,5)$
$A_6(6,4)$
$A_8(4,5)$

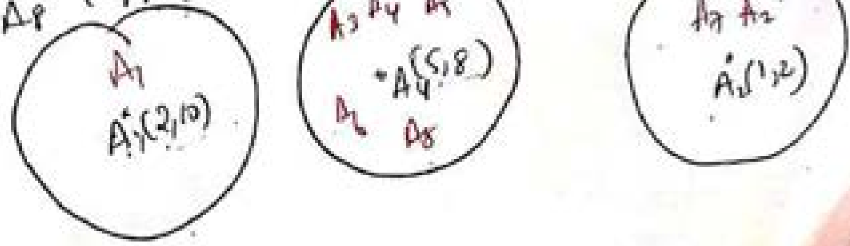
Cluster 3

$A_2(2,5)$
$A_7(1,2)$

2,10 5,8

$\sqrt{(5-2)^2+(10-8)^2}$

$3+2=5$

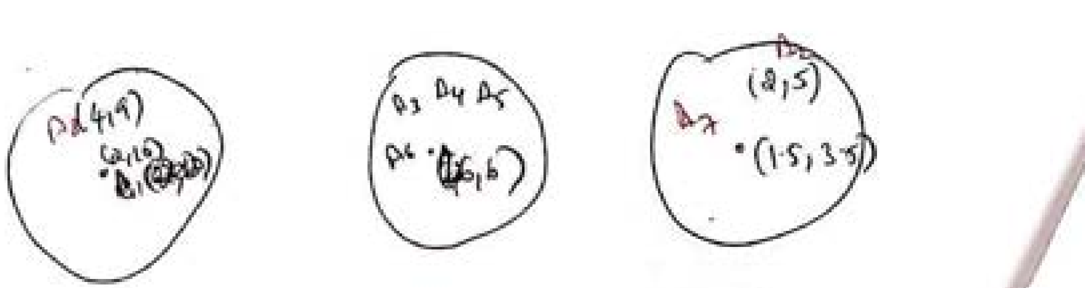


Iteration 2

Next we need to recompute the new cluster centre means, we do so by taking the mean of all points in each cluster.

- For cluster 1, we only have one point A1(2,10) which was the old mean, so cluster centre remains the same.
- For cluster 2, we have $((8+5+7+6+4)/5, (4+8+5+4+9)/5) = (6,6)$
- For cluster 3, we have $((2+1)/2, (5+2)/2) = (1.5, 3.5)$

Point		1 (2,10) Dist mean1	2 (6,6) Dist mean2	3 (1.5,3.5) Dist mean3	cluster
A1	(2,10)	0	8	7	1
A2	(2,5)	5	5	2	3
A3	(8,4)	12	4	7	2
A4	(5,8)	5	3	8	2
A5	(7,5)	10	2	7	2
A6	(6,4)	10	2	5	2
A7	(1,2)	9	5	2	3
A8	(4,9)	3	5	8	1



iterations

Cluster 1. $((2+4)/2, (10+9)/2) = (3, 9.5)$

Cluster 2: we have (3,4,5,&6) points
Mean is $((8+5+7+6)/4, (4+8+5+4)/4) = (6.5, 5.25)$

Cluster 3: we have 2 and 7 points. The
Mean is $((2+1)/2, (5+2)/2) = (1.5, 3.5))$

Points	(3.67, 9) Dist mean1	(~~4~~ 6.5, ~~5.3~~ 5.25) Dist mean2	(1.5, 3.5) Dist mean3	Cluster
A_1 (2,10)	2.67	10.7	7	1
A_2 (2,5)	5.67	5.7	2	3
A_3 (8,4)	9.33	1.3	7	2
A_4 (5,8)	2.33	5.7	8	1
A_5 (7,5)	7.33	0.7	7	2
A_6 (6,4)	7.33	3.3	5	2
A_7 (1,2)	9.67	8.3	2	3
A_8 (4,9)	0.33	7.7	8	1

K-means will stops each of the data points does not change the clusters.

C

Cluster 1 we have 1, 4, & 8. Therefore

mean is $((2+5+4)/2, (10,8+9)/2) = (3.67, 9)$

Cluster2 we have 3, 5, & 6

Centroid/mean is $((8+7+6)/4, (4+5,+4)/4) = (7, 4\,3)$

Cluster3 we have point (2, & 7)

Mean is $((2+1)/2, (5+2)/2) = (1.5, 3.5)$

	Point	(3.67,9) mean1	(7,4.3) mean2	(1.5,3.5) mean3	Cluster
A1	(2,10)	2.67	10.7	7	1
A2	(2,5)	5.67	5.7	2	3
A3	(8,4)	9.33	1.3	7	2
A4	(5,8)	2.33	5.7	8	1
A5	(7,5)	7.33	8.7	7	2
A6	(6,4)	7.33	1.3	5	2
A7	(1,2)	9.67	8.3	2	3
A8	(4,9)	0.33	7.7	8	1

Expectation Maximization Algm

In k-means clustering, we just allocated the pixels based on means, It would be better if we consider variance also

→ If we map the image size onto the graphs, which gives the probability density function.

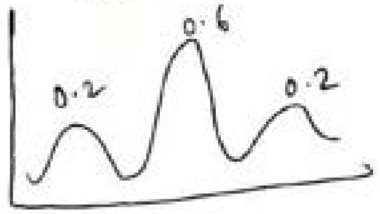
→ Given a training samples, what is the probability that a feature vector x

$$p(x) = \sum_{k=1}^{k} \pi_k N(x/\mu_k, \Sigma_k)$$

It is the summation of different gaussians which means μ_1, μ_2, μ_3 and variance $\sigma_1^2, \sigma_2^2, \sigma_3^3$

π_k is the weight vector

$$p(x) = \sum_{k=1}^{k} \pi_k N(x/\mu_k, \Sigma)$$



Given a training sample, what is the probability it belongs to which gaussian.

First gaussian is assigned to variable z_1
second " " " " " z_2
third " " " " " z_3

$P(z/_{x=5})$ what is the probability that test image $x=5$ falls in first gaussian, or second gaussian or third gaussian.

If probability of z_1 is higher then test image belongs to first gaussian ......

Apply Bayes rule.

$$P(z_k/x) = \frac{P(x/z_k)\,P(z_k)}{P(x)}$$

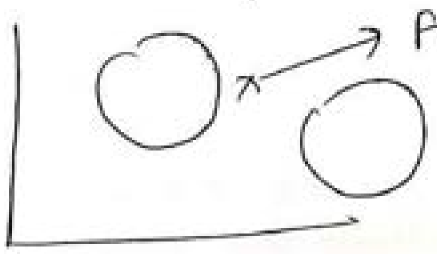
B

Algorithm for expectation Maximization Algm

1) Initialize the means μ_k, Σ_k.

Find the probability which sample belongs to which cluster.

2) <u>E - step</u>:- Evaluate the responsibilities using current parameters, μ, Σ.

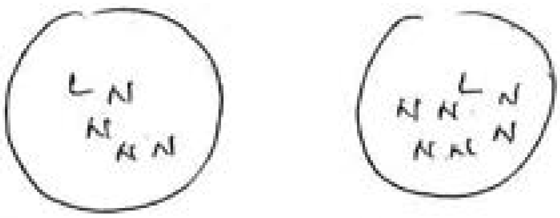


Find the probability of the datapoint with respect to first & second cluster.
If the probability of datapoint belonging to first cluster is more then assign to first cluster.

3) <u>M-step</u>:- Restimate the parameters.

Supervised Learning after clustering

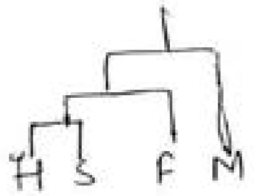
→ Analysis the groups of labelled and unlabelled points.
Find the spacially closed points and checking how unchried unlabelled points are towards each of their surrounding labelled points.
The method is used to find the closest labelled point for each unlabelled point and assign the point.



L- labelled data point
N → Unlabelled data point.

Hierarchical clustering.

Starts with one cluster, individual items in its own cluster and iteratively merge clusters until all the items belongs to one cluster is known Hierarchical agglomerative clustering.



→ It is Bottom up Approach.

→ It uses a similarity or distance measure defined b/w instances.

Divisive clustering: Start with single group & dividing large group into smaller groups until each group contains single instances.

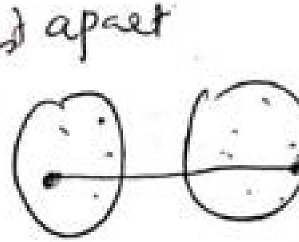
Agglomerative clustering is represented by following

1) **Single link clustering** - This is the distance b/w closest members of two clusters

$$d(G_i, G_j) = \min d(x^r, x^s)$$

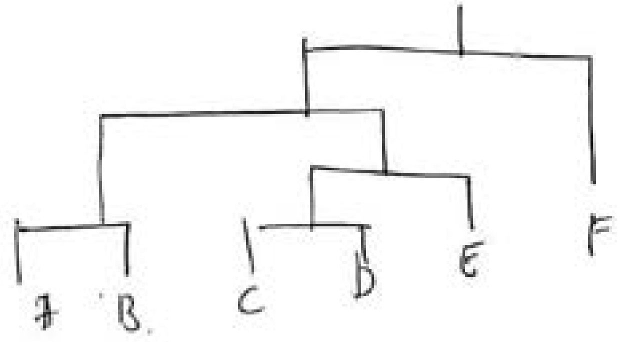
2) **Complete link clustering** :- This is the distance between the members that are farthest apart

$$d(G_i, G_j) = \max d(x^r, x^s)$$



3) **Average link clustering** This method involves looking at the distances b/w all pairs and averages all of these distances

1. Dendogram : A tree like structure which represent hierarchial technique



Find the cluster using single link technique using Euclidean distance and draw the dendogram.

	x	y
P_1	0.40	0.53
P_2	0.22	0.38
P_3	0.35	0.32
P_4	0.26	0.19
P_5	0.08	0.41
P_6	0.41	0.30

Distance matrix use Euclidean distance b/w every pair of point

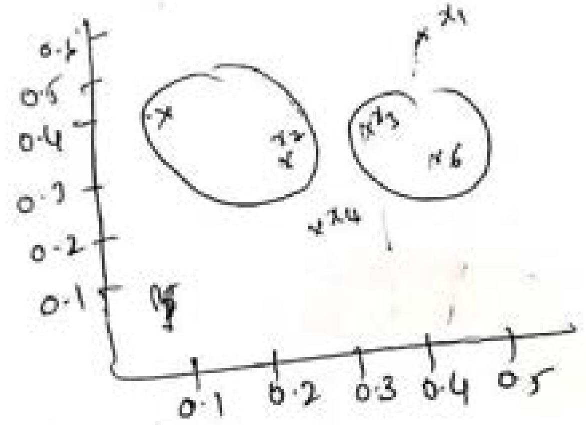
Eg: $\text{Dist}(P_1, P_2) = \sqrt{(0.40-0.22)^2 + (0.53-0.38)^2}$

$= 0.23$

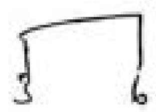
	P_1	P_2	P_3	P_4	P_5	P_6
P_1	0					
P_2	0.23	0				
P_3	0.22	0.15	0			
P_4	0.37	0.20	0.15	0		
P_5	0.34	0.14	0.28	0.29	0	
P_6	0.23	0.25	<u>0.11</u>	0.22	0.39	0

Step 1

In this distance matrix find the lowerbound value [common for single link, complete link & Average link)

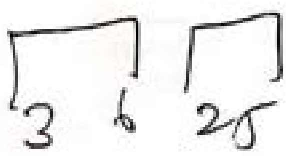


Step 1



Note:

	P_1	P_2	P_3, P_6	P_4	P_5
P_1	0				
P_2	.23	0			
P_3P_6	0.22	0.15			
P_4	0.37	0.20	0.15		
P_5	0.34	<u>0.14</u>	0.28	0.29	



Note: In single link clustering

Using single link $\Rightarrow \min((P_3, P_1), (P_6, P_1))$

Distance b/w (P_3, P_6) & P_1 $= \min(0.22, 0.23)$
$= 0.22$

In complete link clustering:

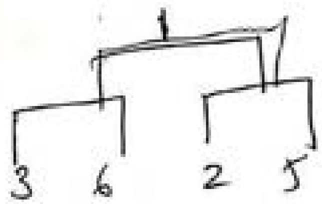
Distance b/w (P_3, P_6) & P_1 $= \max((P_3, P_1), (P_6, P_1))$
$= \max(0.22, 0.23)$
$= 0.23$

In Average link clustering:

Dist b/w (P_3, P_6) & P_1 $= \frac{1}{2}(\text{dist}(P_3, P_1) + \text{dist}(P_6, P_1))$
$= \frac{1}{2}(0.22 + 0.23)$
$=$

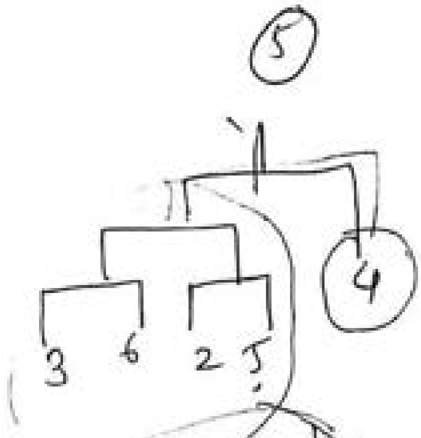
Step 2:

	P_1	P_2P_5	P_3P_6	P_4
P_1	0			
P_2P_5	0.23	0		
P_3P_6	0.22	0.15	0	
P_4	0.37	0.20	0.15	0

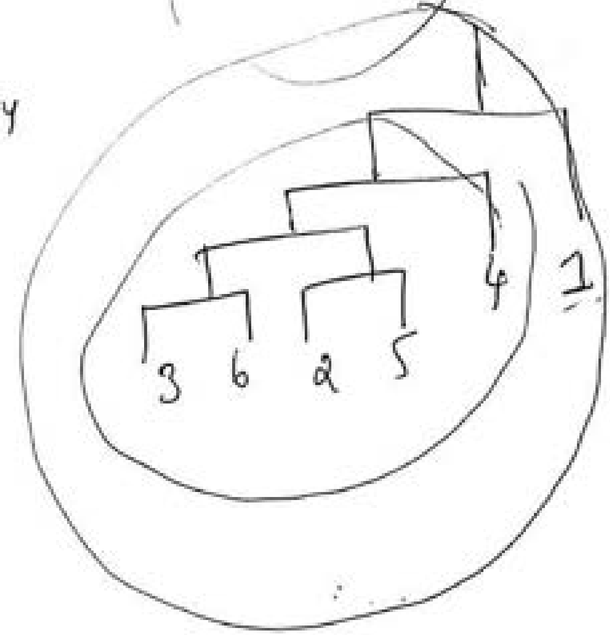


Eg: $\text{Dist}((P_3, P_6), (P_2, P_5))$
$= \min(\min((P_3, P_6), P_2), ((P_3, P_6), P_5))$
$= \min(0.15, 0.28) = 0.15$

	P_1	$P_2P_5P_3P_6$	P_4
P_1	0		
$P_2P_5P_3P_6$	0.22	0	
P_4	0.37	0.15	0



	P_1	$P_2P_5P_3P_6P_4$
P_1	0	
$P_2P_5P_3P_4P_6$	0.22	0



Complete link clustering

	P_1	P_2	P_3	P_4	P_5	P_6
P_1	0					
P_2	.23	0				
P_3	.22	.15	0			
P_4	.37	.20	.15	0		
P_5	.34	.14	.28	.29	0	
P_6	.23	.25	.11	.22	.39	0

	P_1	P_2	P_{36}	P_4	P_5
P_1	0				
P_2	.23	0			
P_3P_6	.23	.25	0		
P_4	.37	.20	.22	0	
P_5	.34	.14	.39	.29	0

	P_1	P_2P_5	P_3P_6	P_4
P_1	0			
P_2P_5	.34	0		
P_3P_6	.23	.39	0	
P_4	.37	.29	.22	0

	P_1	P_2P_5	$P_3P_6P_4$
P_1	0		
P_{25}	.34	0	
$P_3P_6P_4$	.37	.39	0

	$P_1P_2P_5$	$P_3P_6P_4$
$P_1P_2P_5$	0	
$P_3P_6P_4$	.39	0