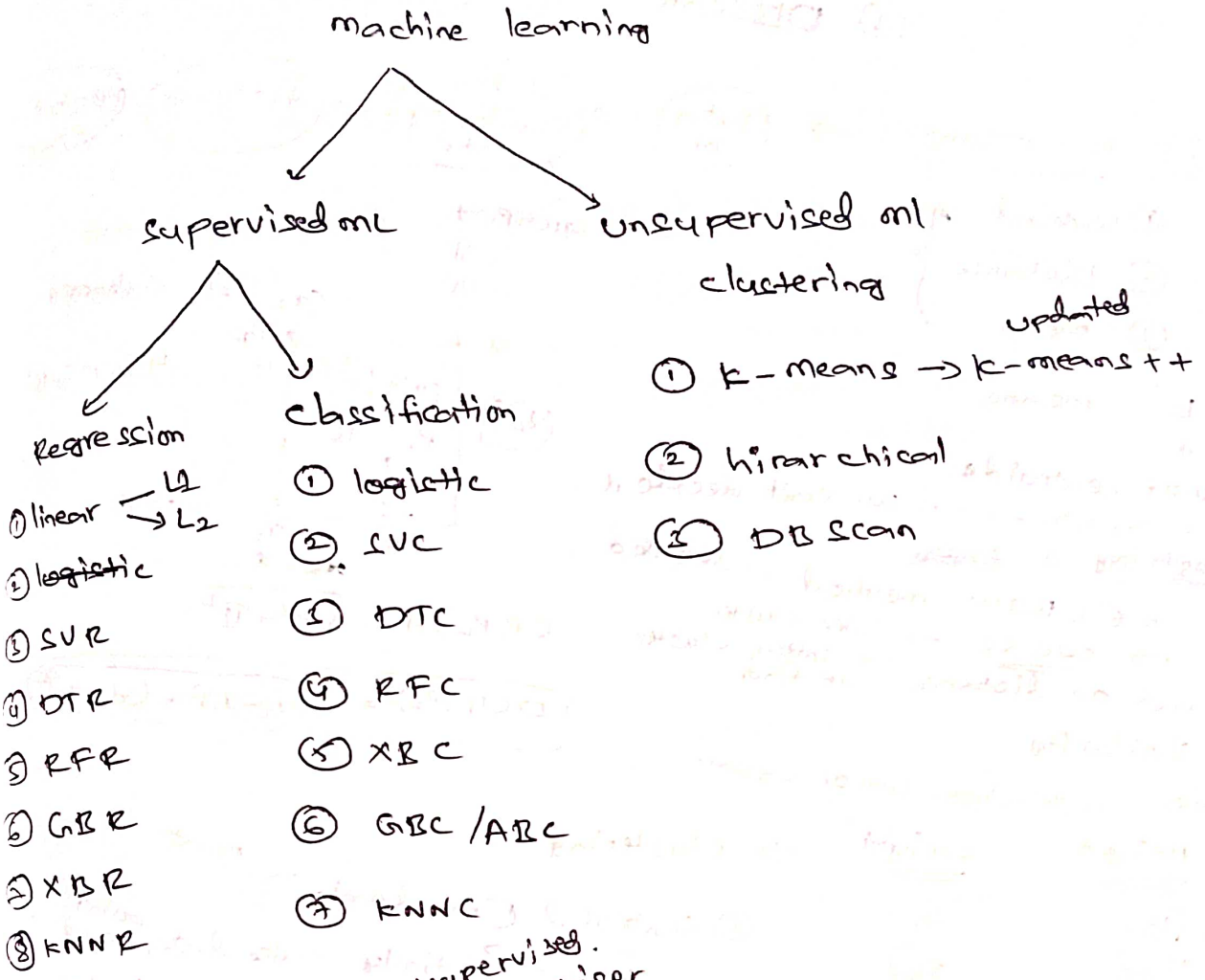
Unsupervised Machine learning



target / dependent / supervised. supervised ml

supervisor ↓ (body mass Index)

Height	weight	BMI
170	60	21
180	65	22
160	70	20
165	75	18
140	75	19

unsuper ml ↓

dataset

unsupervised ml → clustering → grouping

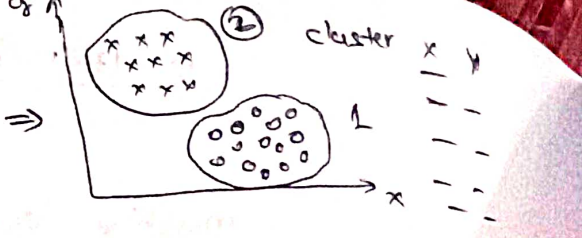
Height	weight	BMI	Country
170	60	21	IND
180	65	22	UK
160	70	20	USA
165	75	18	IND
140	75	19	USA

2 cluster

1- groups

mathematically

① K means
② hirarchical
③ DBScan



① K - means → data set → find similarity data → distance → euclidean dis

grouping

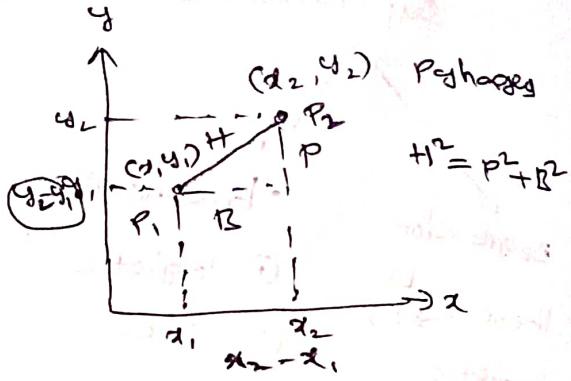
① centroid
② Distance
③ mean

K - means
↑
no. of centroids
2, 3, 4, 5, - - - - we can't decide d

~~clustering~~ → ~~score~~
→ ELBOW method → decided
→ WCSS → inter cluster
→ intera cluster

based on distance we find similarity

wcss = withincluster sum of square



Pythagoras
$H^2 = P^2 + B^2$

$(P_1, P_2) H = \sqrt{P^2 + B^2}$

$D(P_1, P_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$

	Height	weight
① c_1	185	72
② c_2	170	56
③	168	60
4	179	68
5	182	72
6	188	77
7	180	71
8	160	70
9	183	84
10	180	88
⑪	180	67
⑫	167	76

→ clustering

① centroid [randomly] point

K=2 → initially take data point and build around cluster

No of centroid = Number of cluster

validation :-
→ Dunn index
→ silhoutee coff

K=2

c_1 (185, 72)

distance 1 point

E.D → $d(c_1, 1)$ → $c_1 \to 1$ ⇒ 5 dis
→ $d(c_2, 1)$ → $c_2 \to 1$ ⇒ 8 dist

c_1 (185, 72) ③ (168, 60)

$\Rightarrow \sqrt{(168-185)^2 + (60-72)^2}$

$\Rightarrow \sqrt{\quad}$

$D(c_1, 3) \Rightarrow 20.80$

c_2 (170, 56) ③ (168, 60)

$D(c_2, 3) \Rightarrow \sqrt{(168-170)^2 + (60-56)^2}$

$\Rightarrow \sqrt{(-2)^2 + (4)^2}$

$D(c_2, 3) \Rightarrow \sqrt{20} = 4.4$

if belongs to 2 cluster. because distance minimum

* nearest distance we consider that cluster

[C2 (170, 56) , ③ (168, 60)]

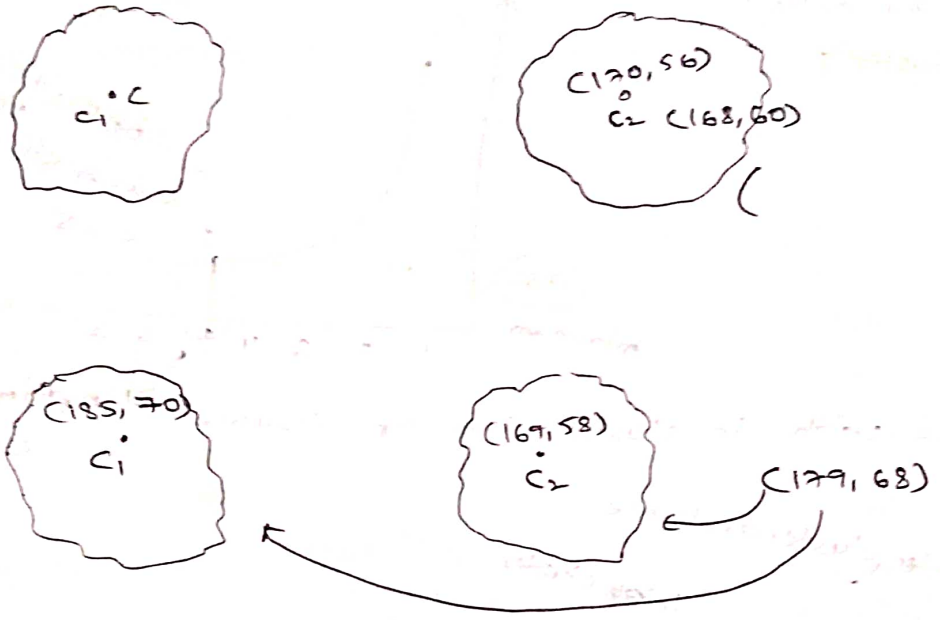
↓

$\left(\frac{170+168}{2}\right), \left(\frac{56+60}{2}\right)$ update the centroid

(169, 58)

k-means → 2 centroid → around build cluster → dis → similarity → nearest point

k=2



$D(C_1,4) = \sqrt{(185-179)^2+(72-68)^2} = 7.21$

$D(C_2,4) = \sqrt{(169-179)^2+(58-68)^2} = 14.14$

conly=9 low distance min distance

update centroid

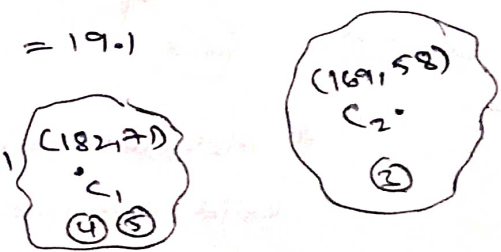
$\frac{179+185}{2} = 182$ $\frac{68+72}{2} = 70$

68+

$D(C_1,5) = \sqrt{(182-182)^2+(72-70)^2} = 2$

$D(C_2,5) = \sqrt{(182-169)^2+(22-58)^2} = 19.1$

update $\frac{182+182}{2} = 182, \frac{70+72}{2} = 71$



k means

① centroid

② Distance (compare , min)

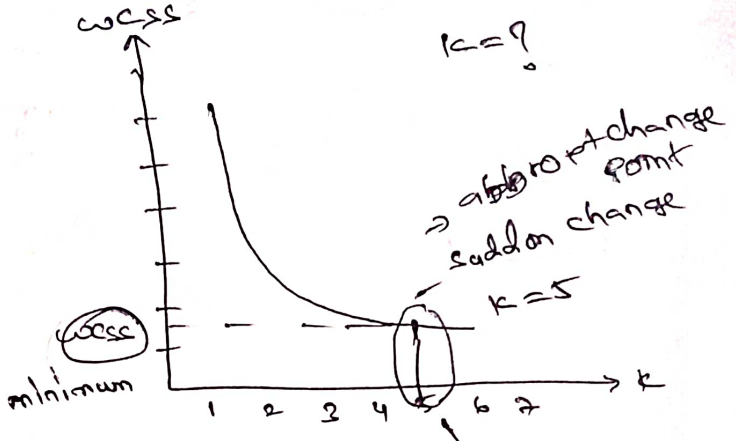
③ include point in cluster, update the cluster

iterate all the data point

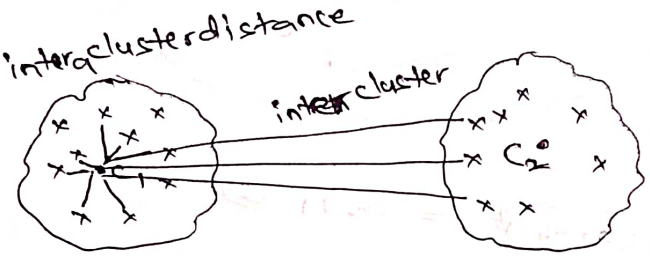
k - means [ELBOW] method

k = 2, 3, 4, 5

[2 cluster]



wcss → with in cluster sum of square



Point and centroid

inter = [C1, C2]

intera = with in cluster

average of cluster

k=1

1 cluster

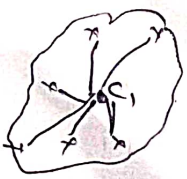
wcss = 25

= wcss1

k = 2

2 cluster

two groups



wcss = 20

wcss2 =

wcss1 > wcss2

* distance minimized in two groups

* solid compare to k=1

wcss = with in cluster sum of

$$\sum_{i=1} d(c,x_i)$$

wcss$_1$ > wcss$_2$ > wcss$_3$

k=3

wcss=15

cluster is going to more solid

minimum wcss

difference between K-means / k-means++

Validate clustering ? k=5

Regression [0-1]
R^2 0 worst
Adj R^2 1 best

classification
→ ROC AUC
confusion matrix

① dunn index
② silhouette score

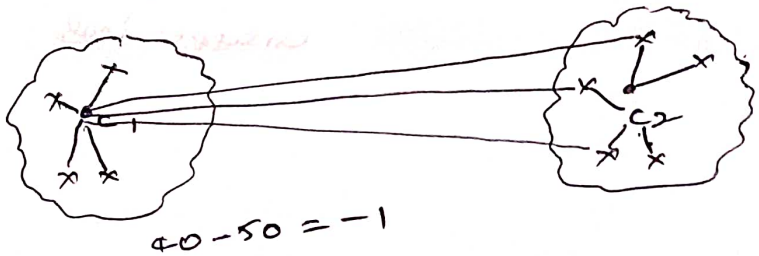
distance

① dunn index = $\frac{\max dis(x_i, z_i)}{\max dis(y_i, y_i)}$

② silhouette score = $\frac{b_i - a_i}{\max(b_i - a_i)}$ → c_1 to c_2 → same cluster [-1 to 1]

worst ⇒ -1 best +1

$b_i = 50 - 40 = 10$



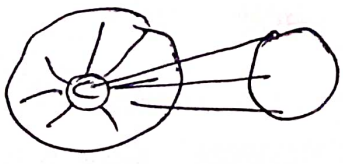
40 - 50 = -1

Same cluster

a_i

b_i

$a_i > b_i$



① Unsupervised

② K-means → K=?

③ how to choos optional K=?

ELBOW guess

④ K=5 → how to validate

silhoute - (-1, +1)

① random centroid

② Distance, min

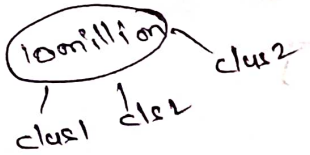
③ update the centroid

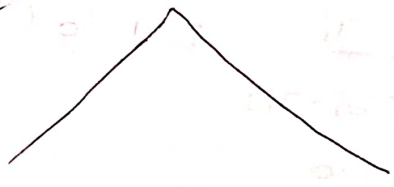
How to make a best model or optimize sol^n

Custom learning or custom model = supervised + unsupervised

(semi supervised)

weight	height	Gender
170	55	M
180	60	F
165	70	M
180	80	F
155	50	F
160	100	F





weight, height, Gende
super

weight, height, Gender
unsupervised

① K-means → K-means ++

② hirarchical

③ ~~kappar~~ DBScan

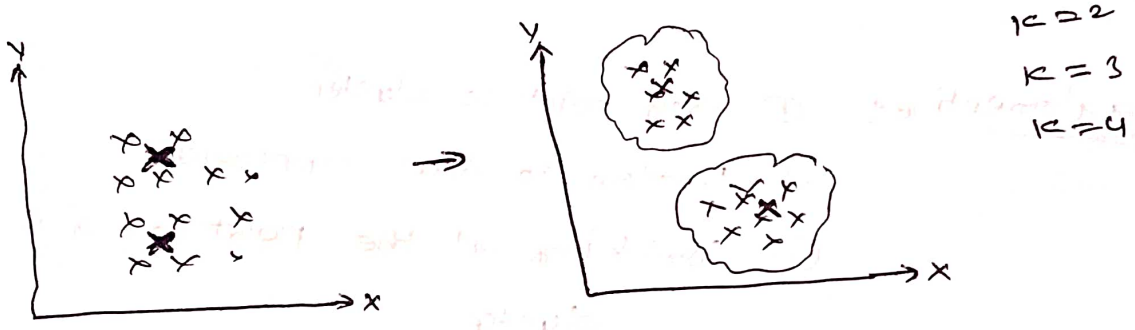
① K-means

① random initialization of centroid

② find out the dist to all the point and make cluster (min dist)

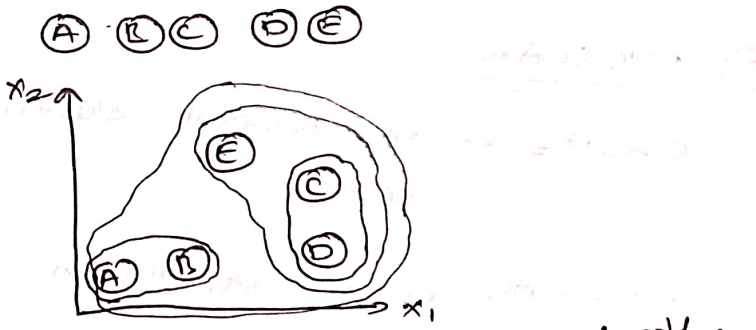
③ update the centroid

K-means algo is centroid based algo

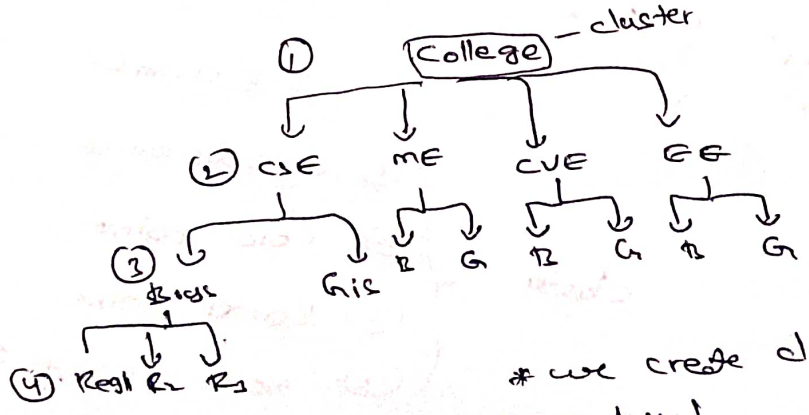


K = 2
K = 3
K = 4

② hirarchical clustering:



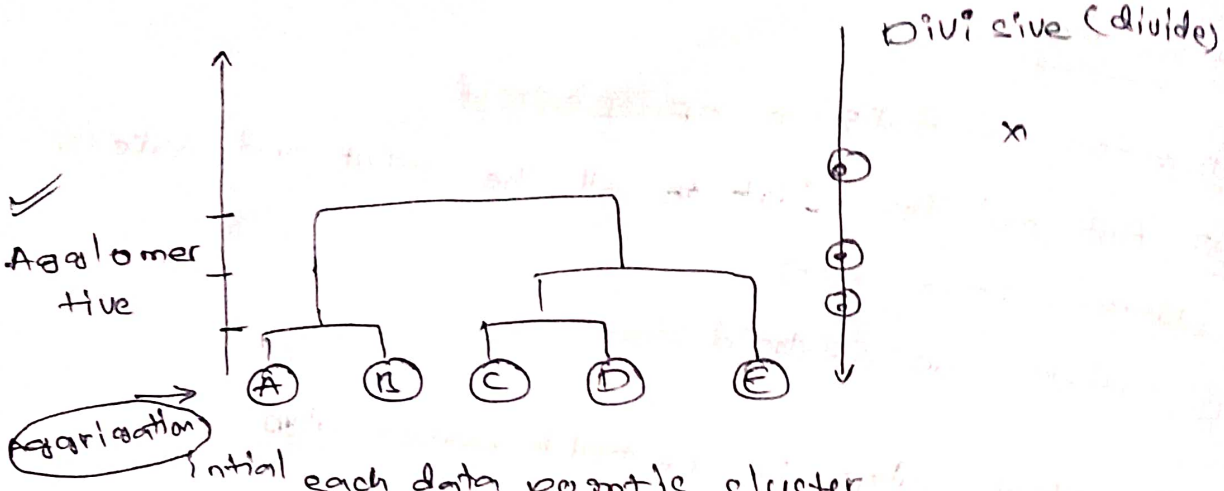
make cluster near eachother
* each data point is cluster



* we create cluster each every level

how to repest hirarchical cluster ?

Dendogram → just a representation of hirarchical clustering



Agglomertive:
① each point is cluster
② Bottom to top approach
③ Combining all the point as a single cluster

③ DBSCAN

Density based spatial clustering with application with noise

⇒ Density based approach

k-mean
hiram
DBscan

cluster
① eps distance (unit distance)
② core point
③ Border point
④ Noise point (outlyer)

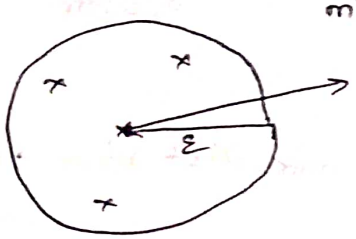
minimum_point

EPS - distance = ε ⇒ draw circle

minimum point = 4 ⇒ this much point should be inside circle

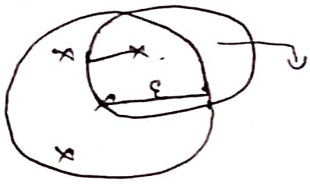
* if inside circle

minimum condition satisfy is core point



hyper parameters ① EPS distance

② min point to be consider in a circle

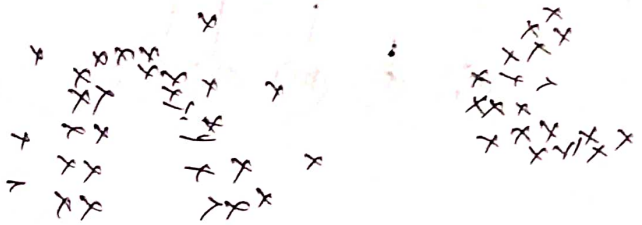


boundry point / Noise point

* i will draw another circle if inside any core point come in the circle it is called boundry point

* it should be neigbour of the core point

Noise point is which not satisfy both conditions



K-mean is unable to solve this type of data

so we use DBSCAN

K-mean
hirarchical } point matrix
DBSCAN

How many point → 5 point (P1) (P2) (P3) (P4) (P5)

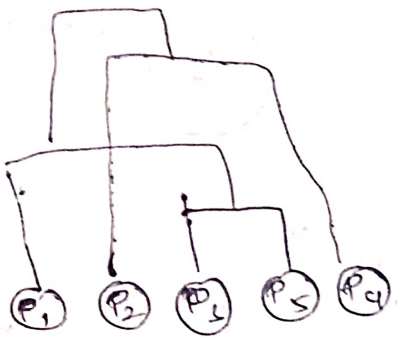
find dist betw the point and creat cluster

	P1	P2	P3	P4	P5
P1	0				
P2	9	0			
P3	3	7	0		
P4	6	5	9	0	
P5	11	10	2		0

min - dist b/w point

	P1	P2	(P3 P5)	P4
P1	0			
P2	9	0		
(P3 P5)	3	7	0	
P4	6	5	8	0

10 9 8 7 6 5 4 3 2 1 0



d (P1, (P3 P5))

min (d (P1, P3), d (P1, P5))

min (3, 11)

= (3)

d (P2, (P3, P5))

(d (P2, P3), d (P2, P5))

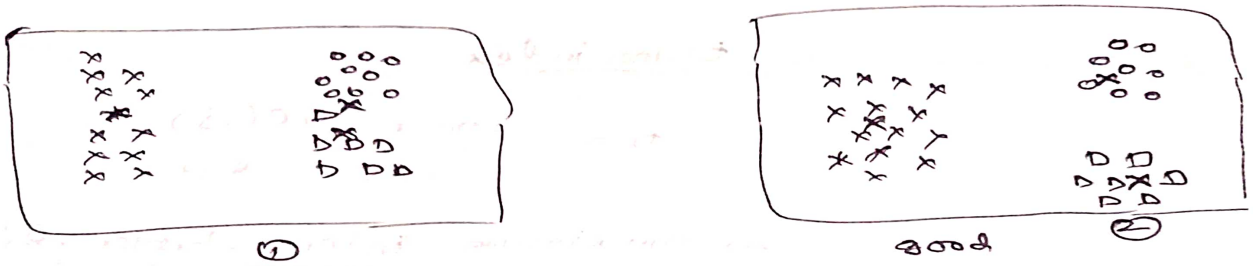
min (7, 10)

⇒ 7

	P_1 P_3 P_5	P_2	P_4
$P_1 P_3 P_5$	0		
P_2		0	
P_4		5	0

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

min $[(P_2, P_1), (P_2, P_3), (P_2, P_5)]$



① it all about centroid initialization

k-means — Randomly × (in many cases we are find ① thing)

① updated

k-means ++

↓ initialized 1 centroid

Probabilistic

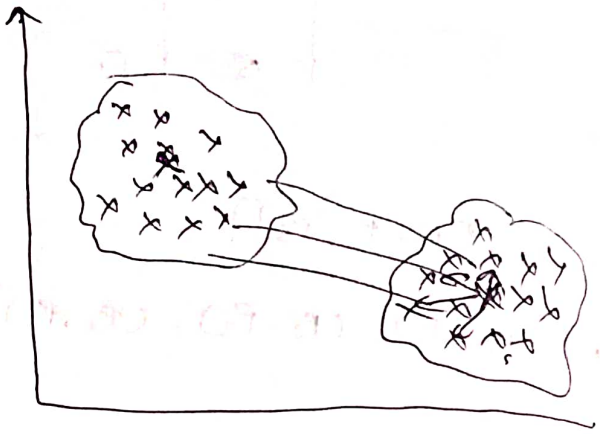
$C \xrightarrow{d} x_i$

farest → make another centroid

* initialize centroid as fare as possible

cluster ① intra cluster dist (inside same cluster)

② inter cluster dist (between a cluster)

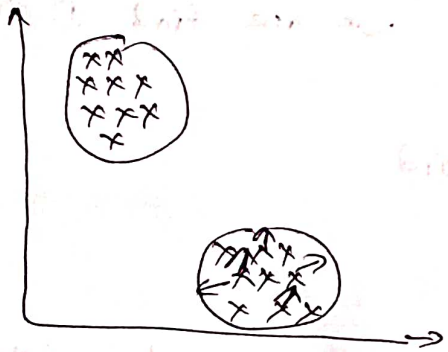


Dunn index

$$D = \frac{\max d(i,j)}{\max d(k)}$$

⇒ maximum inter cluster dist / max intra cluster dist

more solid



Good

$$= \frac{\uparrow \text{as much as possible}}{\downarrow \text{as low as possible}}$$

$[0, \infty]$ maybe + dum index $= \frac{\max(\text{inter})}{\max(\text{intra})}$