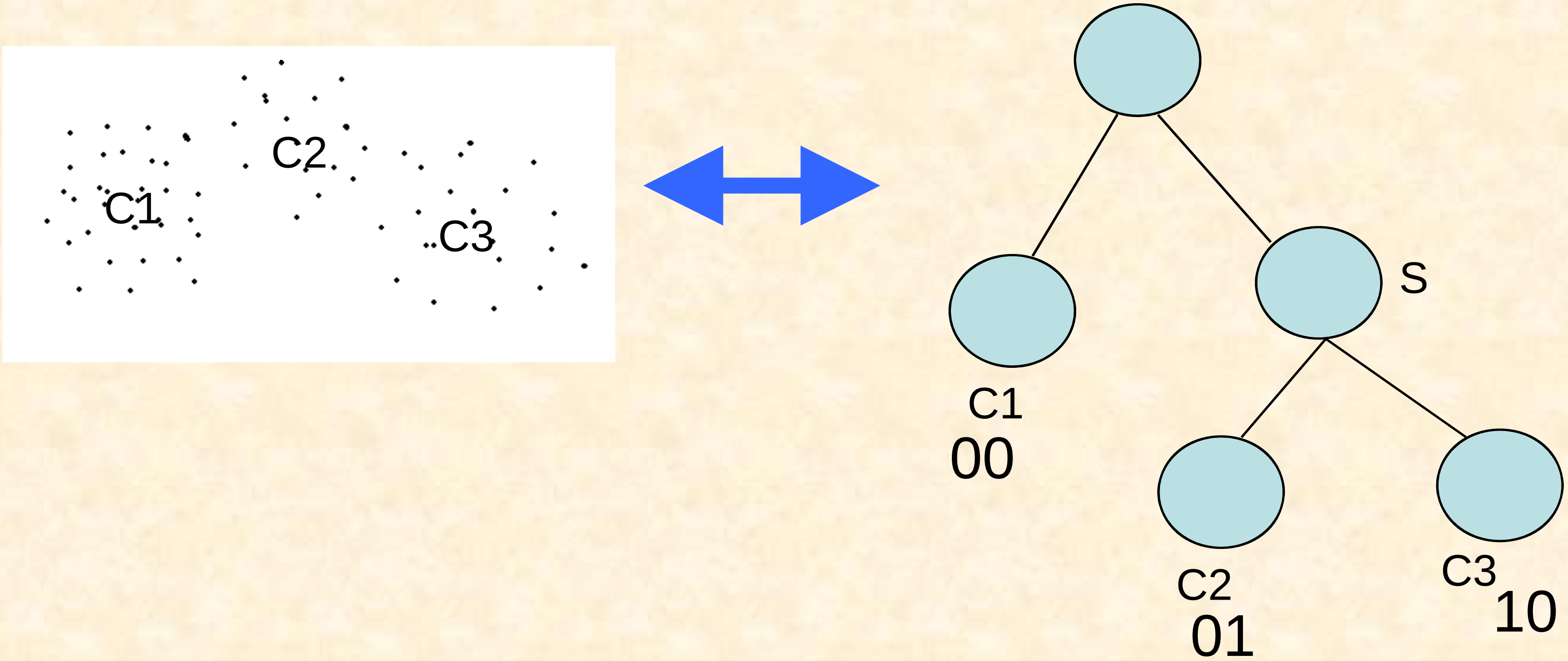
Variable-Branch Decision Tree based on Genetic Algorithm

楊雄彬

Contents

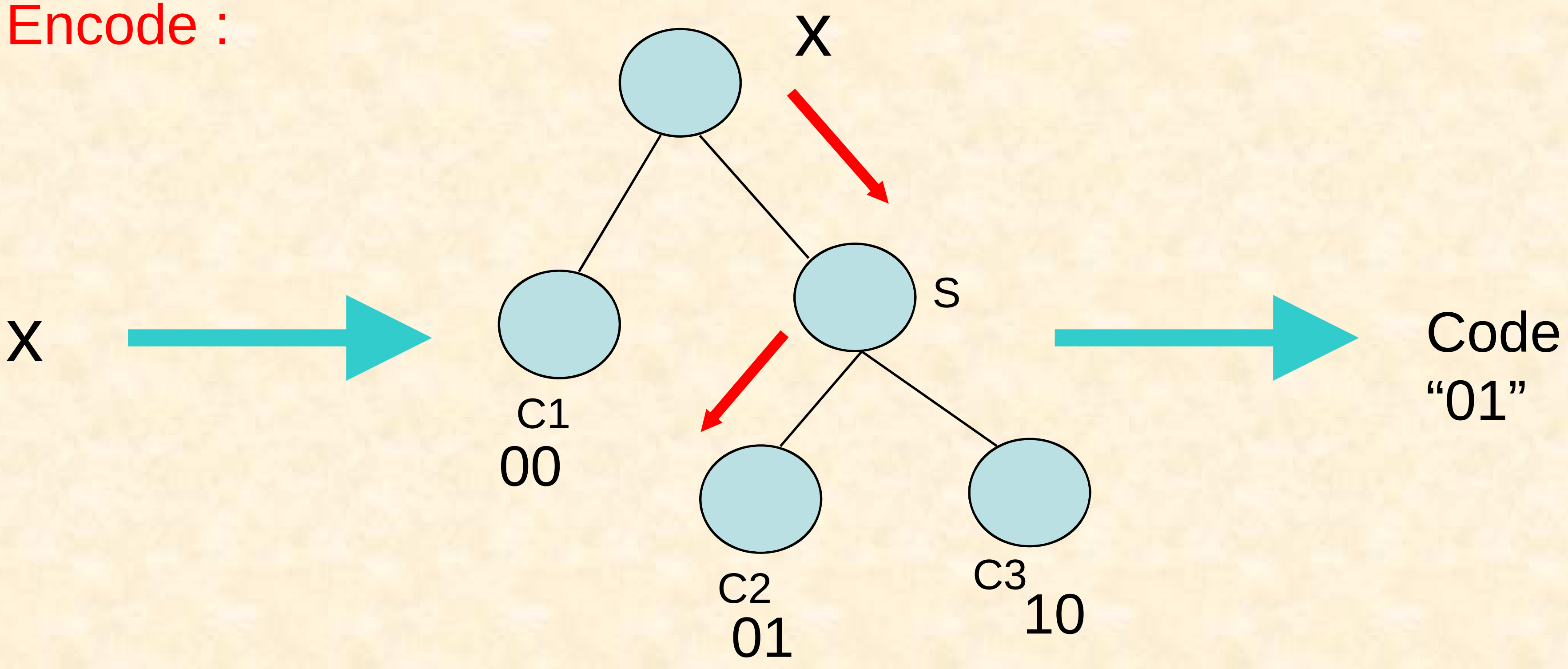
- Decision tree for compression and recognition
- K-means algorithm
- Binary decision tree
- Greedy decision tree
- Two problems of greedy decision tree
- Genetic algorithm for solving problem1.
- Classification points for solving problem 2.
- Conclusions

Decision tree-compression



Codebook tree

Encode :



Codebook tree

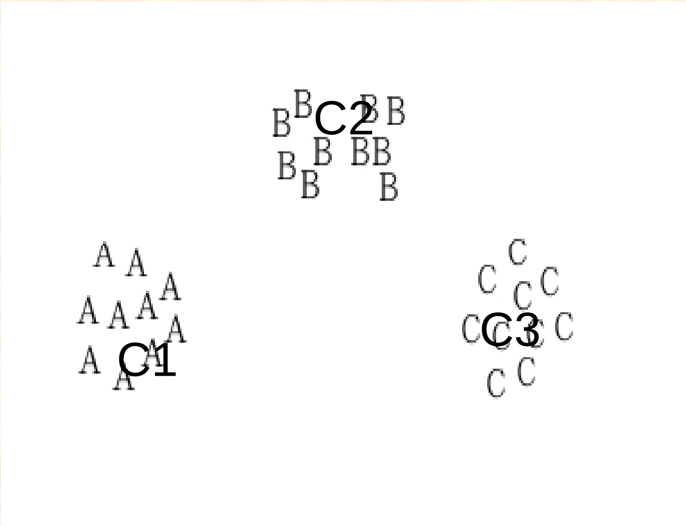
Decode :

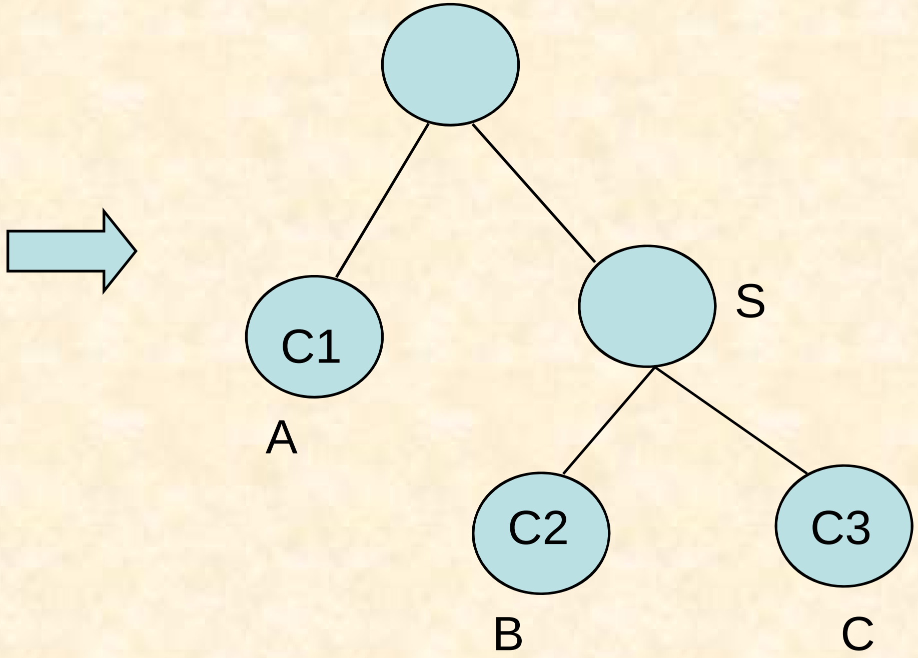
Code “01” →

00	C1
01	C2
10	C3

→ C2

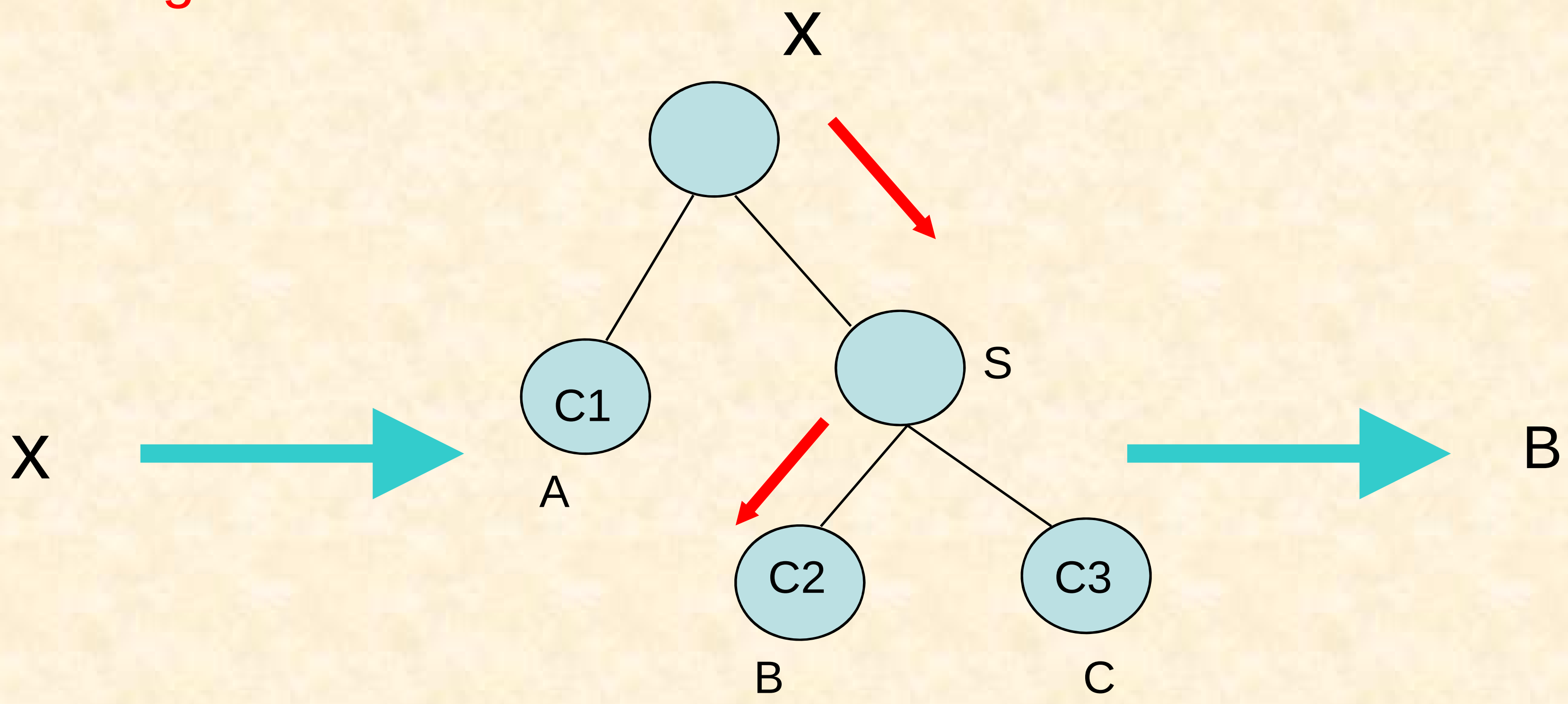
Decision tree-Recognition





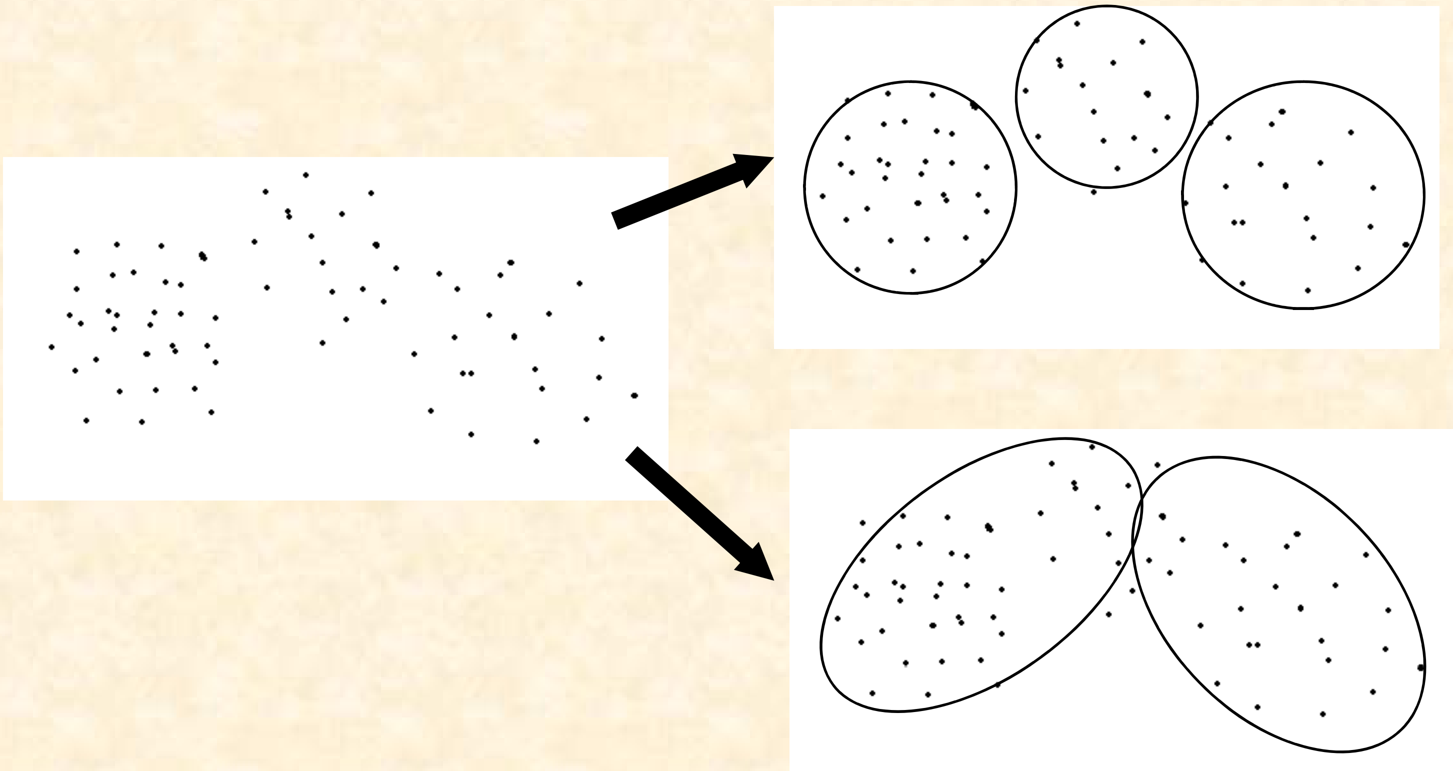
Classification tree

Recognition:

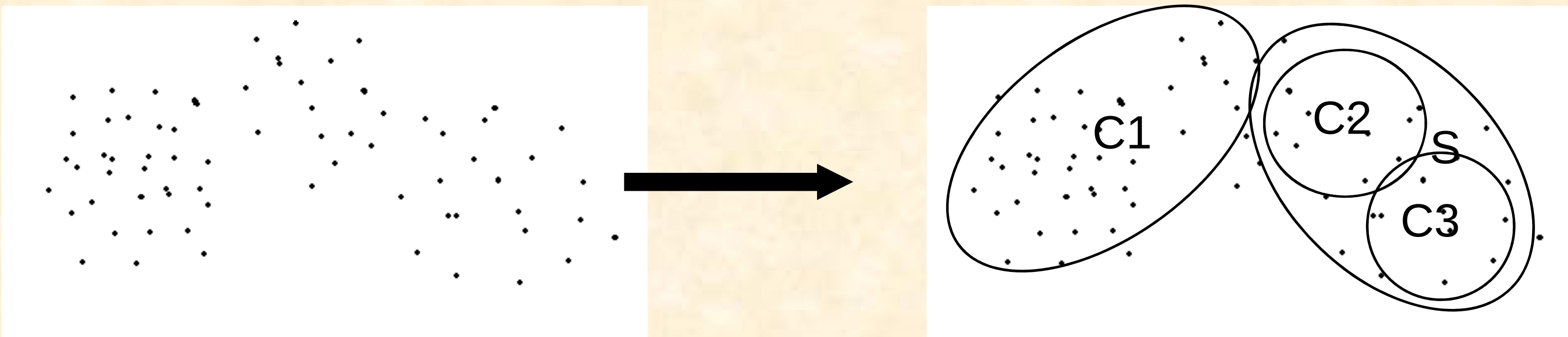


Classification tree

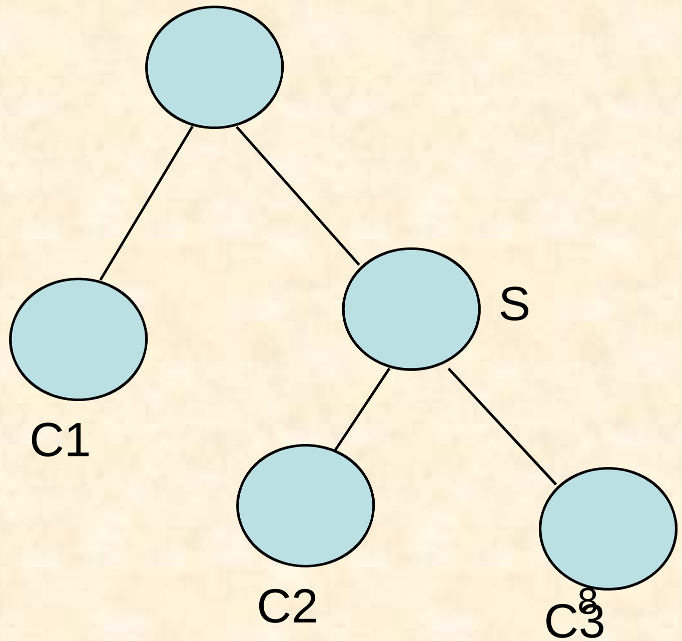
- K-means (C-means) algorithm
  - Divide the data set into k clusters.



- Decision tree ----- Compression
  - Binary decision tree (k=2)

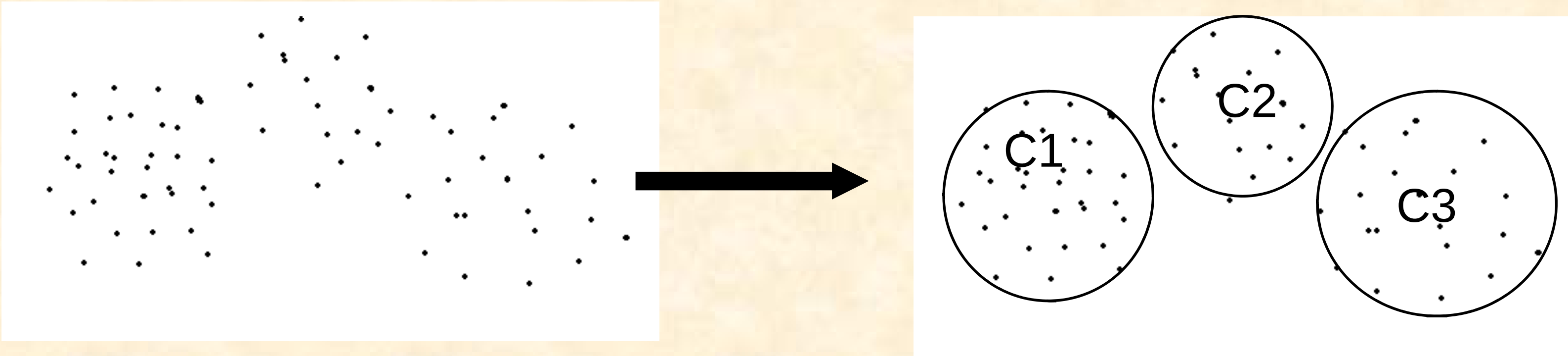


  - Input X is encoded in T1.
    X is compared with three
    codewords, C1, C2 and C3.

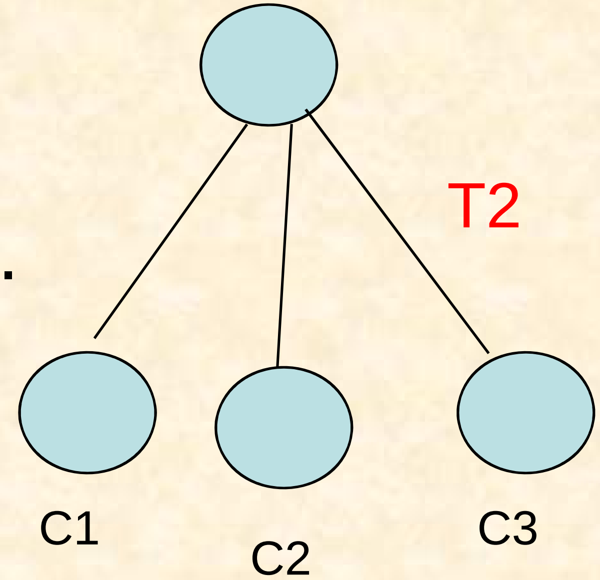


T1

- Decision tree ----- Compression
  - 3-ary decision tree (k=3)



  - Input X is encoded in T2.
    X is compared with three codewords, C1,C2 and C3.

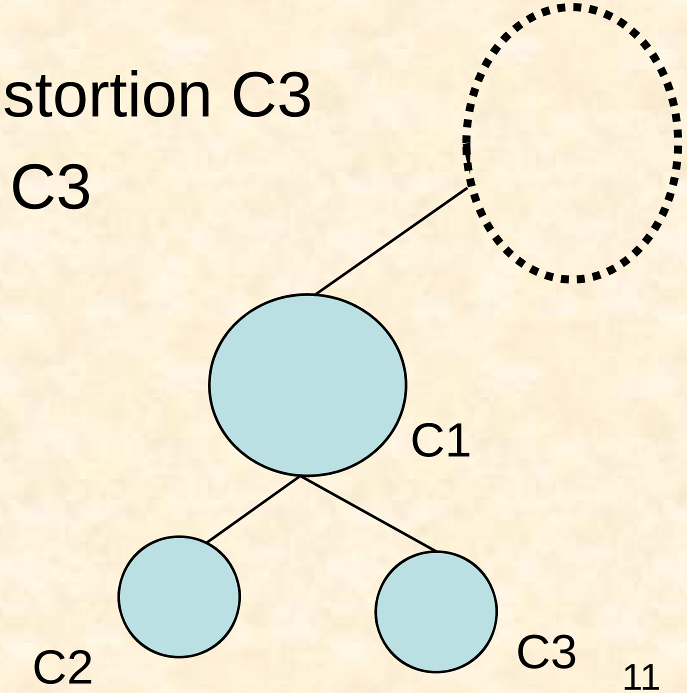


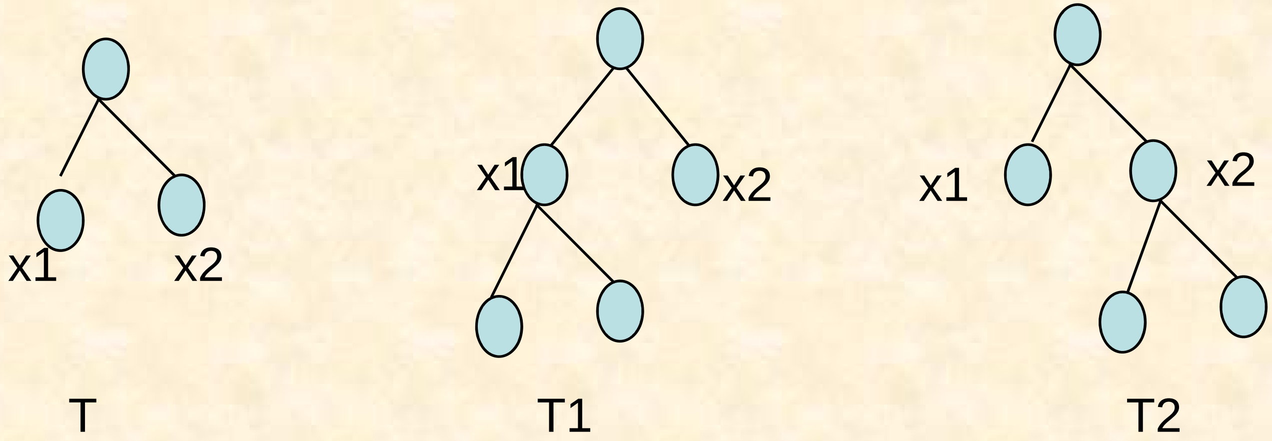
- T1 and T2, which one is better ? Optimal?
  - It is hard for the users to determine which one is better.
  - The users usually have no ideal about the value of k. Thus, T1 and T2 are not optimal.
  - Compression performance is depended on the coding quality and bit rate. The coding quality is as high as possible. The bit rate is as low as possible.

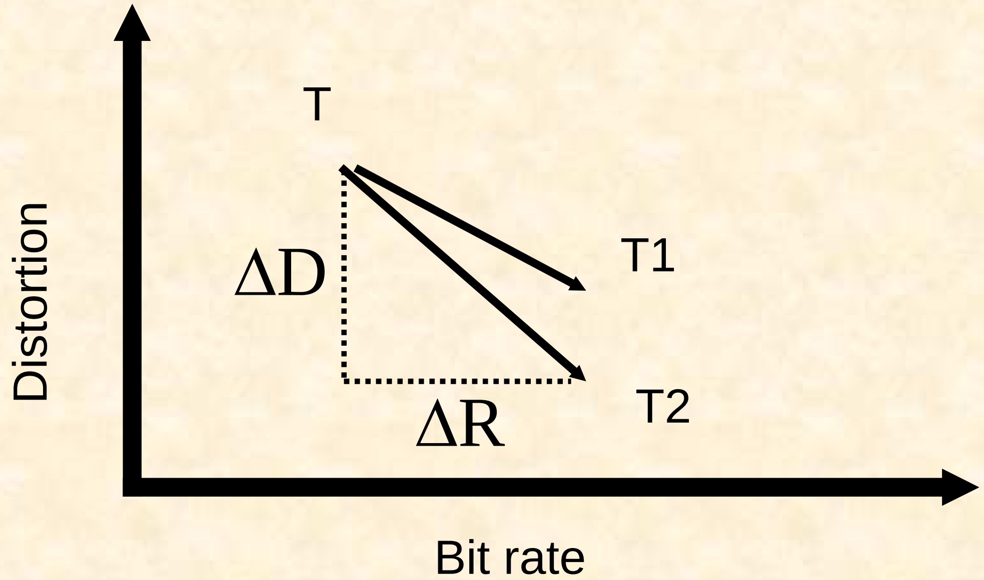
- Greedy decision tree (K=2)
  - The growing method selects nodes with the maximum value of λ to split during the design of decision tree.

$$\lambda = \frac{\Delta \mathrm{Distortion}}{\Delta \mathrm{Bitr\ ate}} = \left| \frac{\Delta D}{\Delta R} \right|$$

  - Distortion C1>Distortion C2+Distortion C3
  - Bit rate C1<Bit rate C2+Bitrate C3

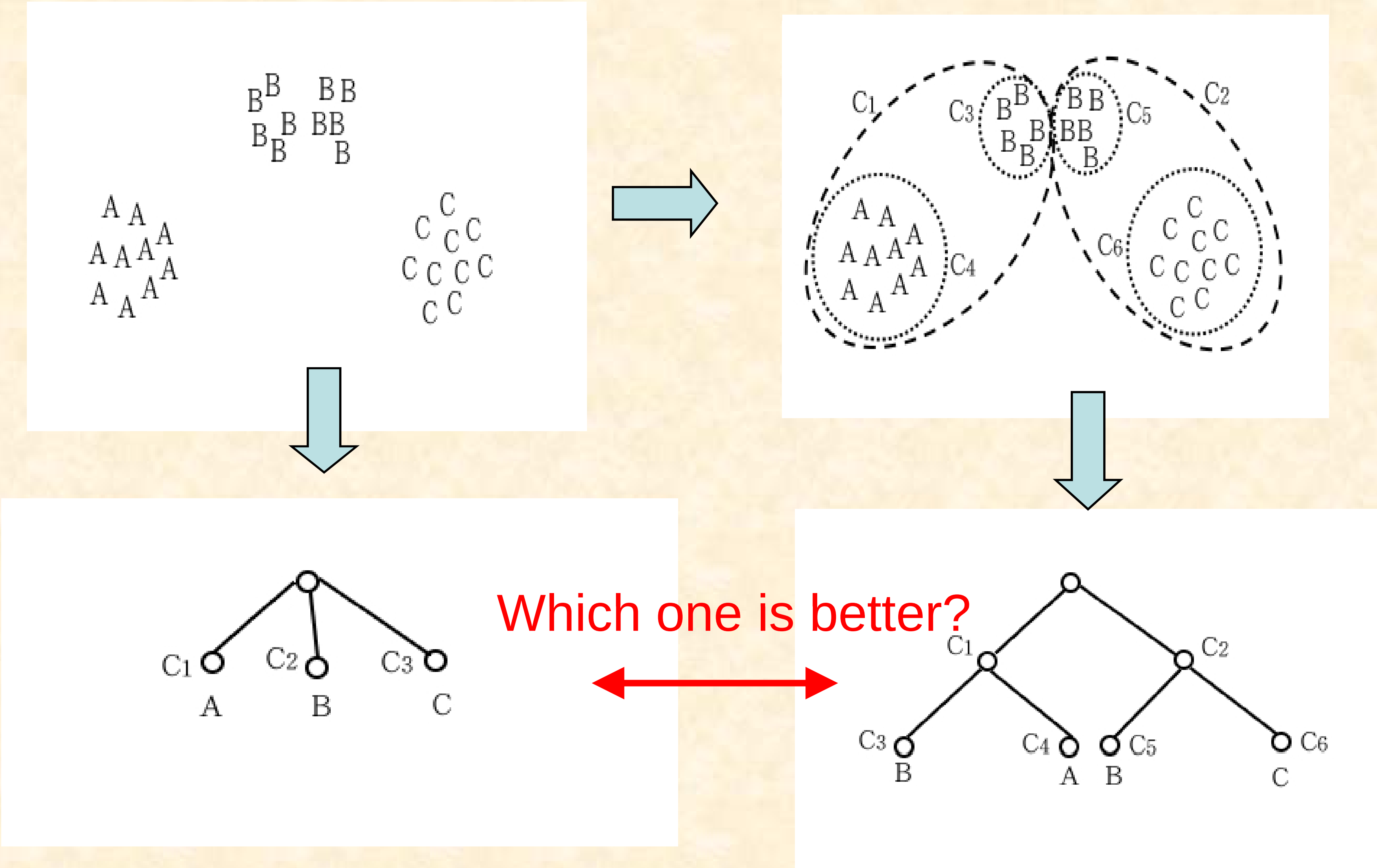






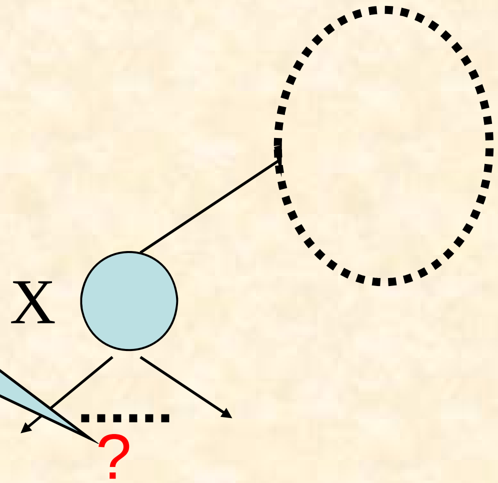
T2 is better than T1.

- Problem1:The greedy decision tree is a fixed-branch decision tree. It is still not an optimal decision tree.

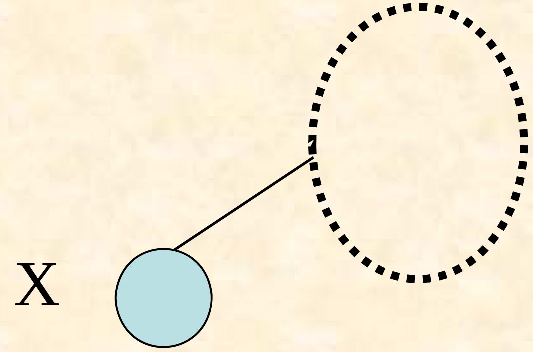


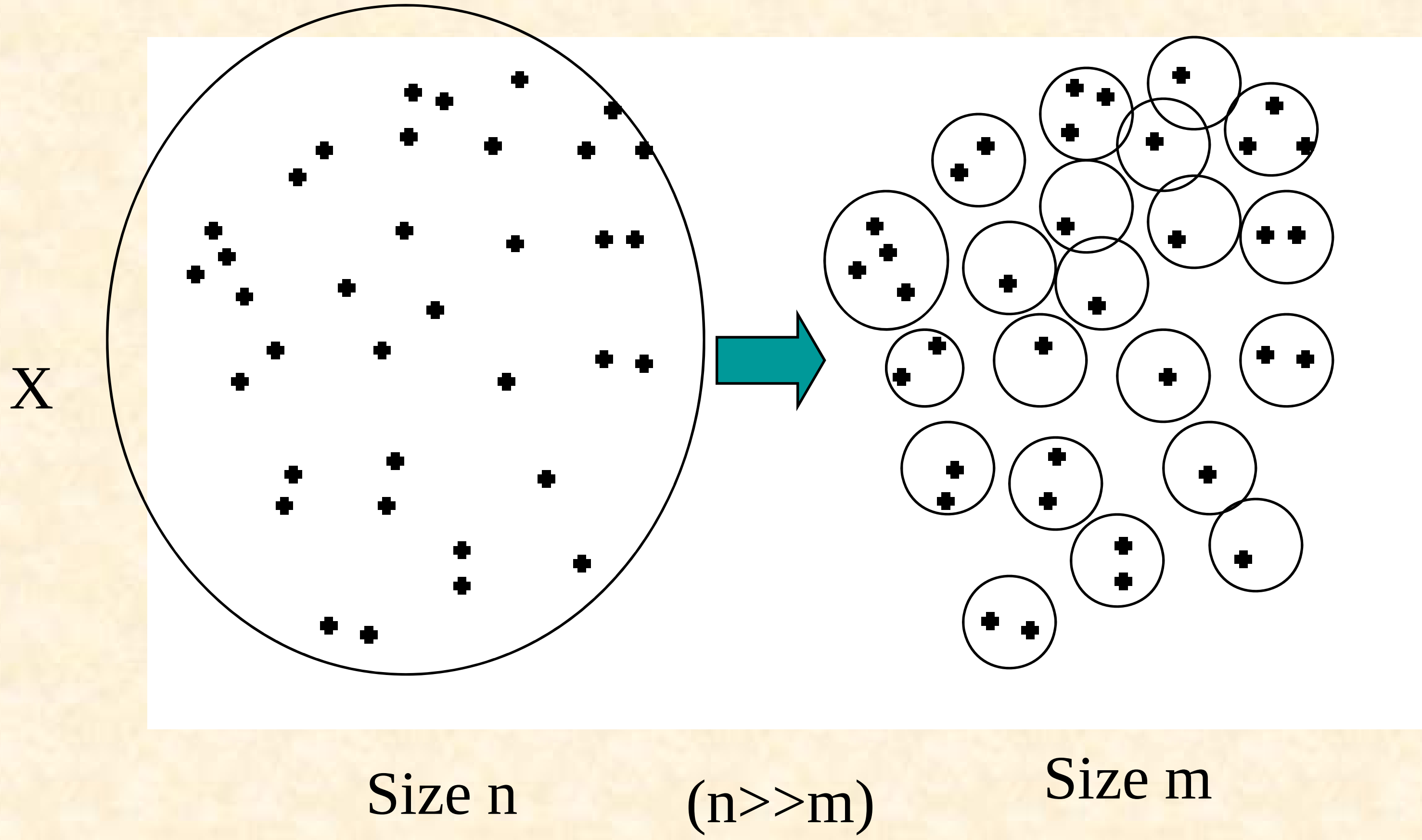
- Solution for Problem1
  - Variable-branch decision tree is proposed to replace the fixed-branch decision tree.
  - How to determine the proper number of branches of a node?
  - "Nearest-neighbor algorithm + Genetic clustering algorithm" automatically determines the proper number of branches in a node.

"NN+GA" searches for the proper number of branches of node X.

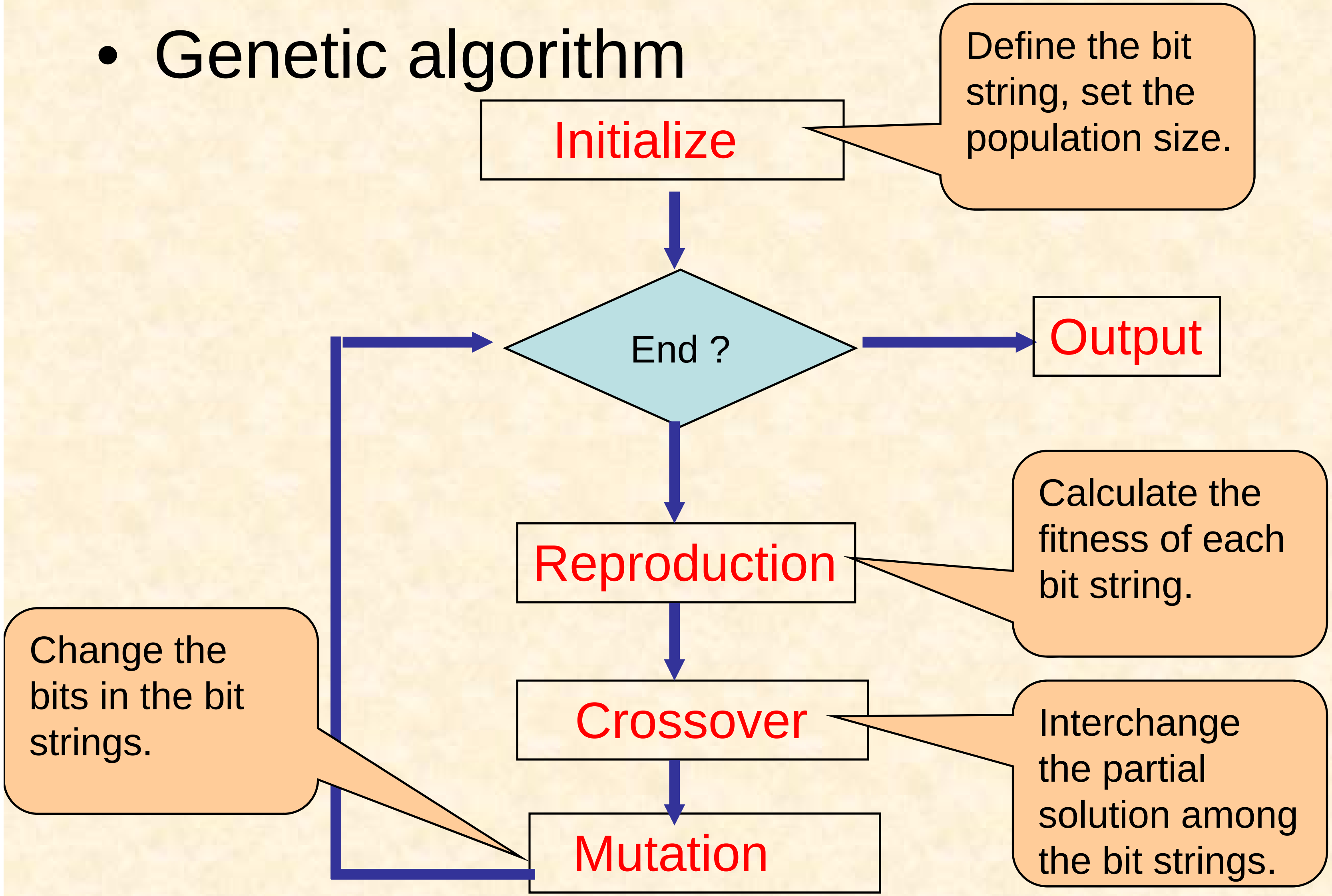


- Reduce the training data set using the nearest-neighbor algorithm.

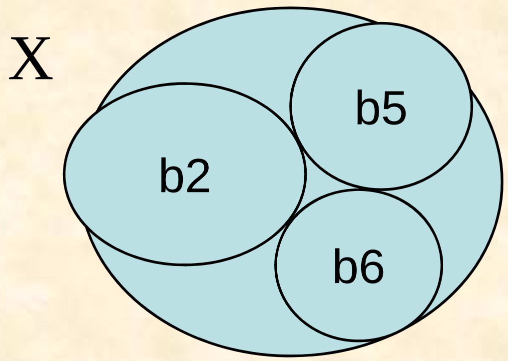




- Genetic algorithm



- Initialize



m

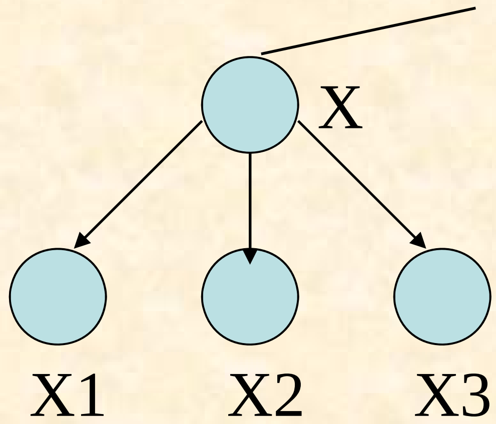
01001………………..
10110………………..
00110………………..

Population size

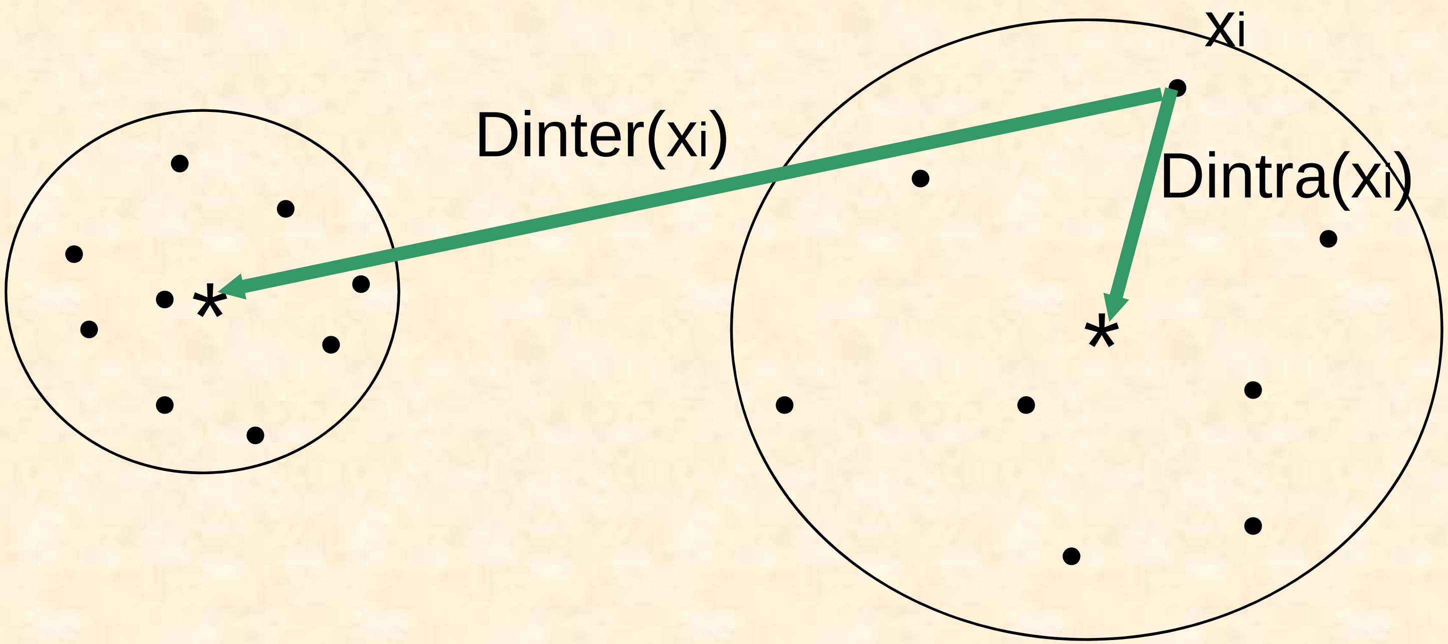
Ex: m=8

	R=	0	1	0	0	1	1	0	0
		b1	b2	b3	b4	b5	b6	b7	b8

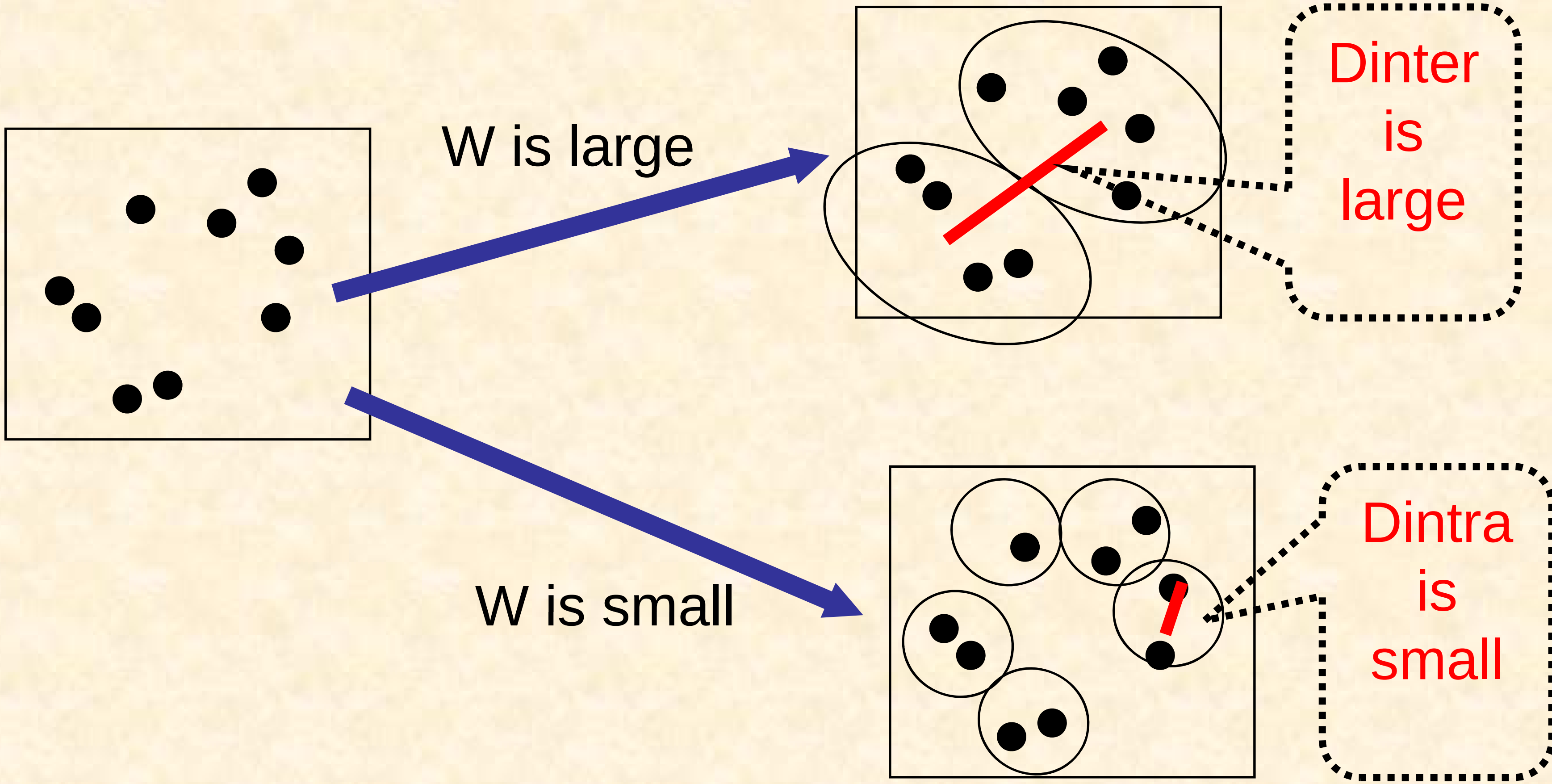
Three initial seeds, b2,b5 and b6, generate three clusters, X1, X2 and X3.



- Reproduction(1)
  - $Fitness(R) = \sum_{x_i \in \text{Dtat set}} [\text{Dinter}(x_i) * w - \text{Dintra}(x_i)]$
  - Dinter(xi) denotes the minimal distance between the sample xi and its nearest cluster.
  - Dintra(xi) denotes the distance between the sample xi and its center.



– If w is large, Fitness is determined by Dinter.
– If w is small, Fitness is determined by Dintra.

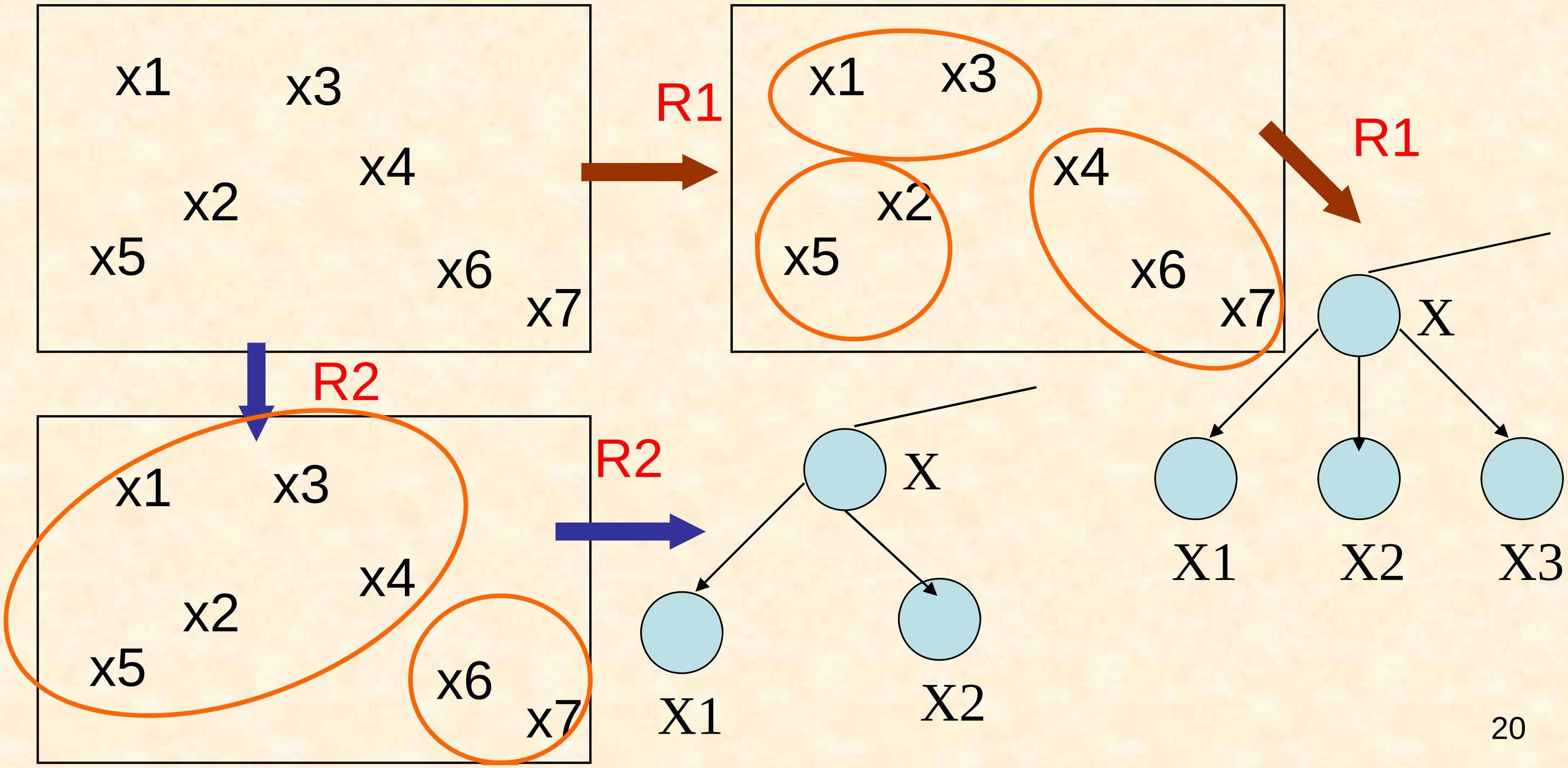


- Ex:

R1= $\boxed{1\ 0\ 0\ 0\ 1\ 1\ 0\ 0}$ (3 clusters)

b1 b2 b3 b4 b5 b6 b7 b8

R2= $\boxed{0\ 0\ 1\ 0\ 0\ 1\ 0\ 0}$ (2 clusters)



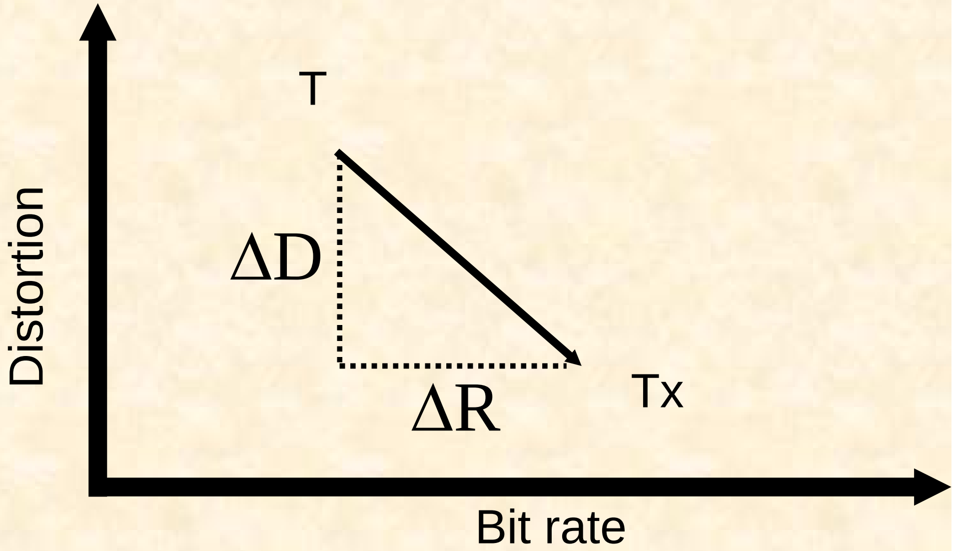
A good clustering result

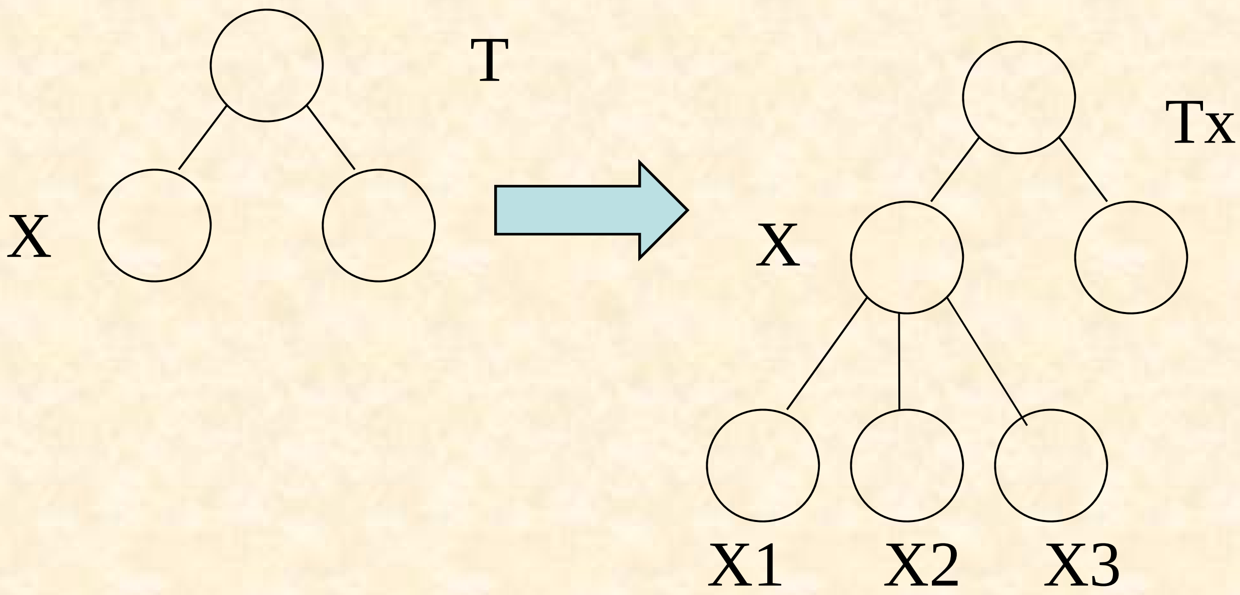
?=?

A good decision tree for compression

- Reproduction(2)

– $Fitness(R) = \lambda = \dfrac{\Delta D}{\Delta R}$



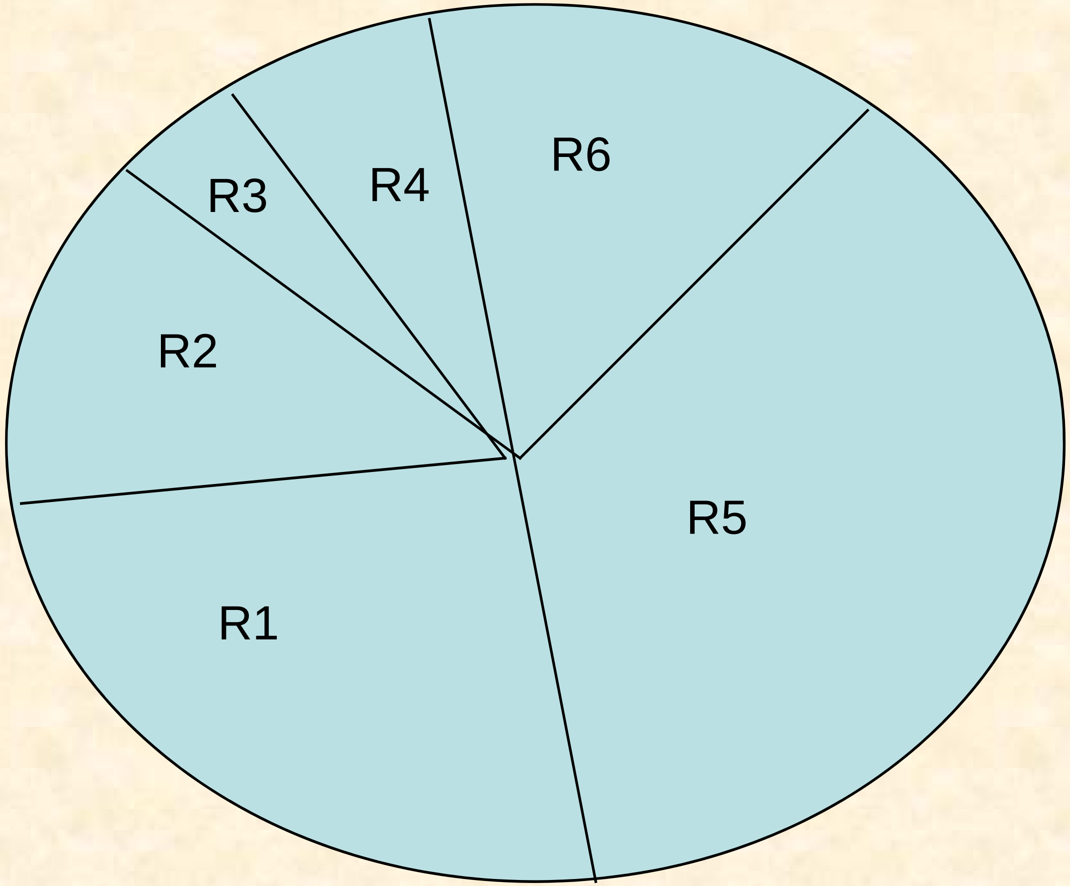


- Ex:

R1=	0	1	0	0	1	1	0	0	(3 branches)
	b1	b2	b3	b4	b5	b6	b7	b8	
R2=	0	0	1	0	0	1	0	0	(2 branches)
R3=	1	1	1	0	0	1	0	0	(4 branches)
R4=	0	0	0	0	1	1	1	1	(4 branches)
R5=	0		1	0	0	1	0	0	(2 branches)
R6=	0	1	0	1	0	1	0	0	(3 branches)

•Fitness(R5)>Fitness(R1)>Fitness(R6)>Fitness(R2)>Fitness(R4) >Fitness(R3)

- Prob(R5)>Prob(R1)>Prob(R6)>Prob(R2)>Prob(R4)> Prob(R3)



- Ex: R5, R1, R5, R6, R5, R4 are selected to be the next population.

- Crossover

R1= 0 0 1 0 0 1 0 0

R2= 1 1 1 0 0 1 0 0

→

R1= 0 1 1 0 0 1 0 0

R2= 1 0 1 0 0 1 0 0

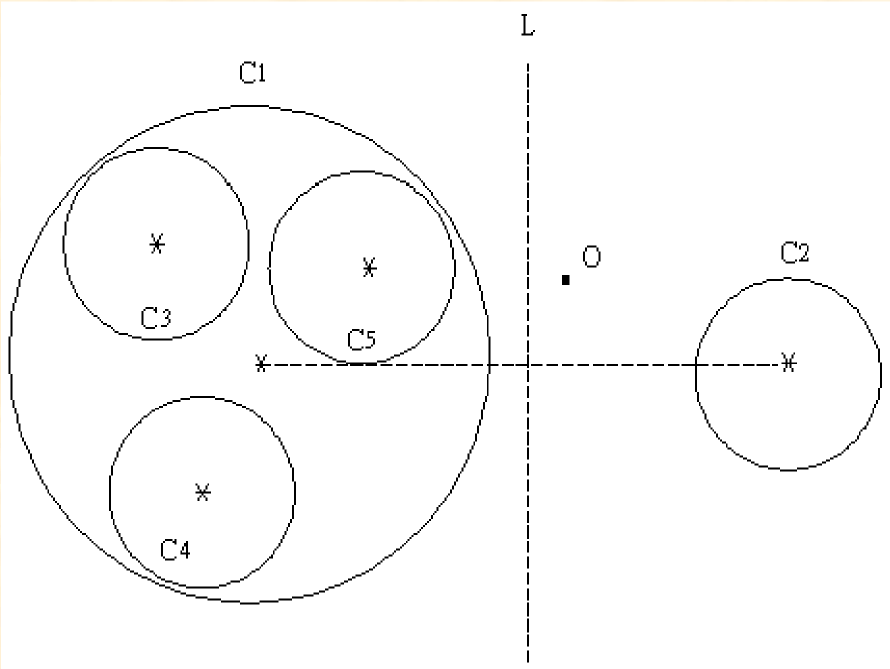
- Mutation

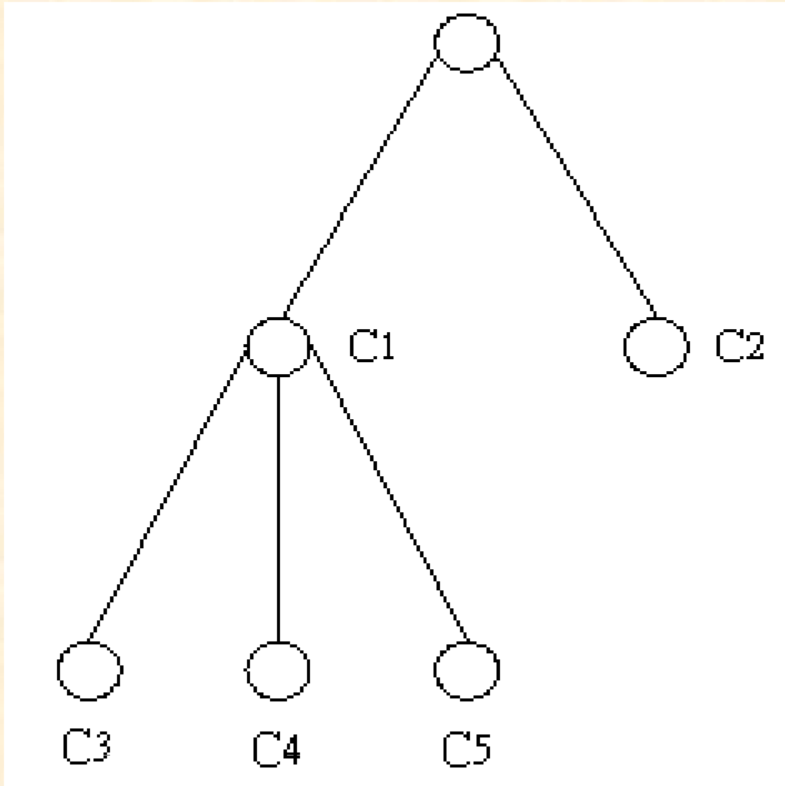
R1= 0 0 1 0 0 1 0 0 → R1= 0 0 0 0 0 1 0 0

0→1

1→0

- Problem 2:The encoding codeword is not the closest codeword to the input X.

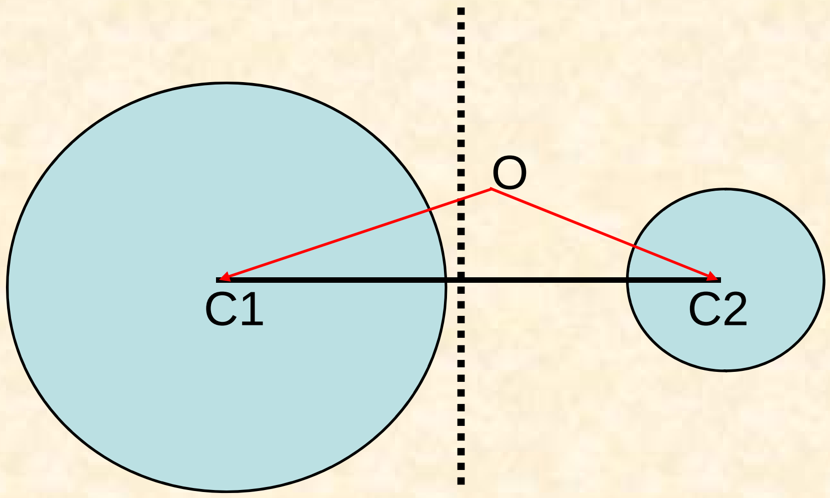




– Input O is encoded by C2 in the decision tree. However, O is closed to C5.

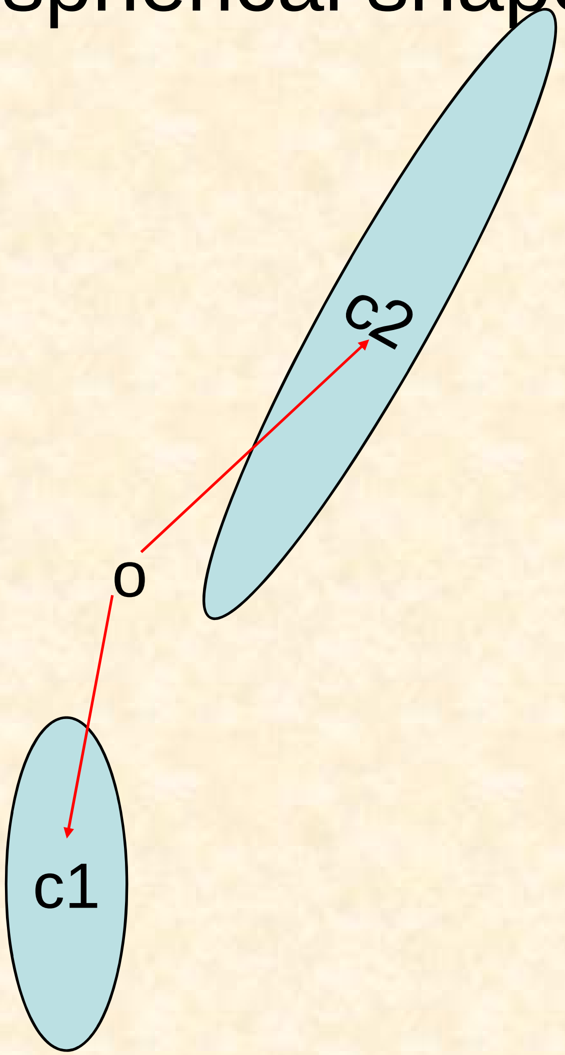
- Solution for Problem 2
  - The cluster center is not proper to classify the input vector in the decision tree.
  - Ex:

(1)Large cluster

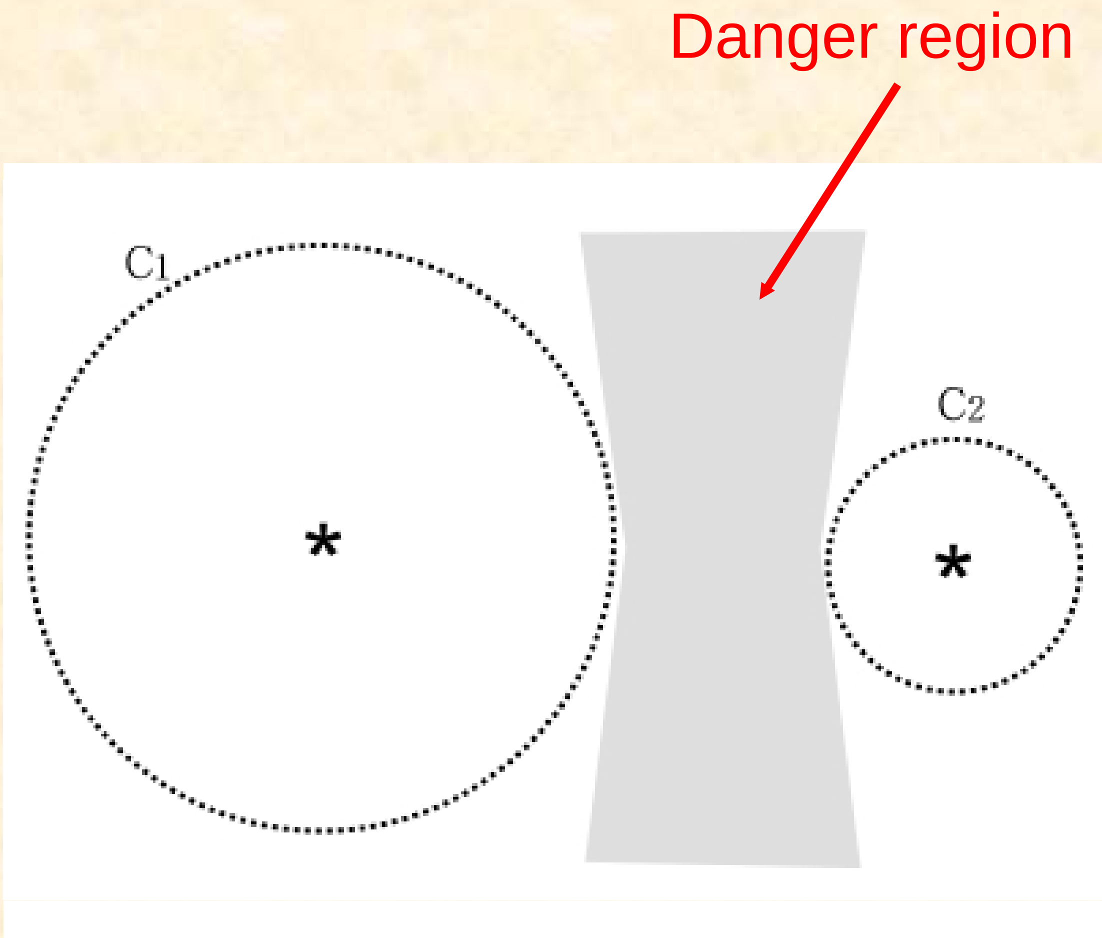


– Ex:

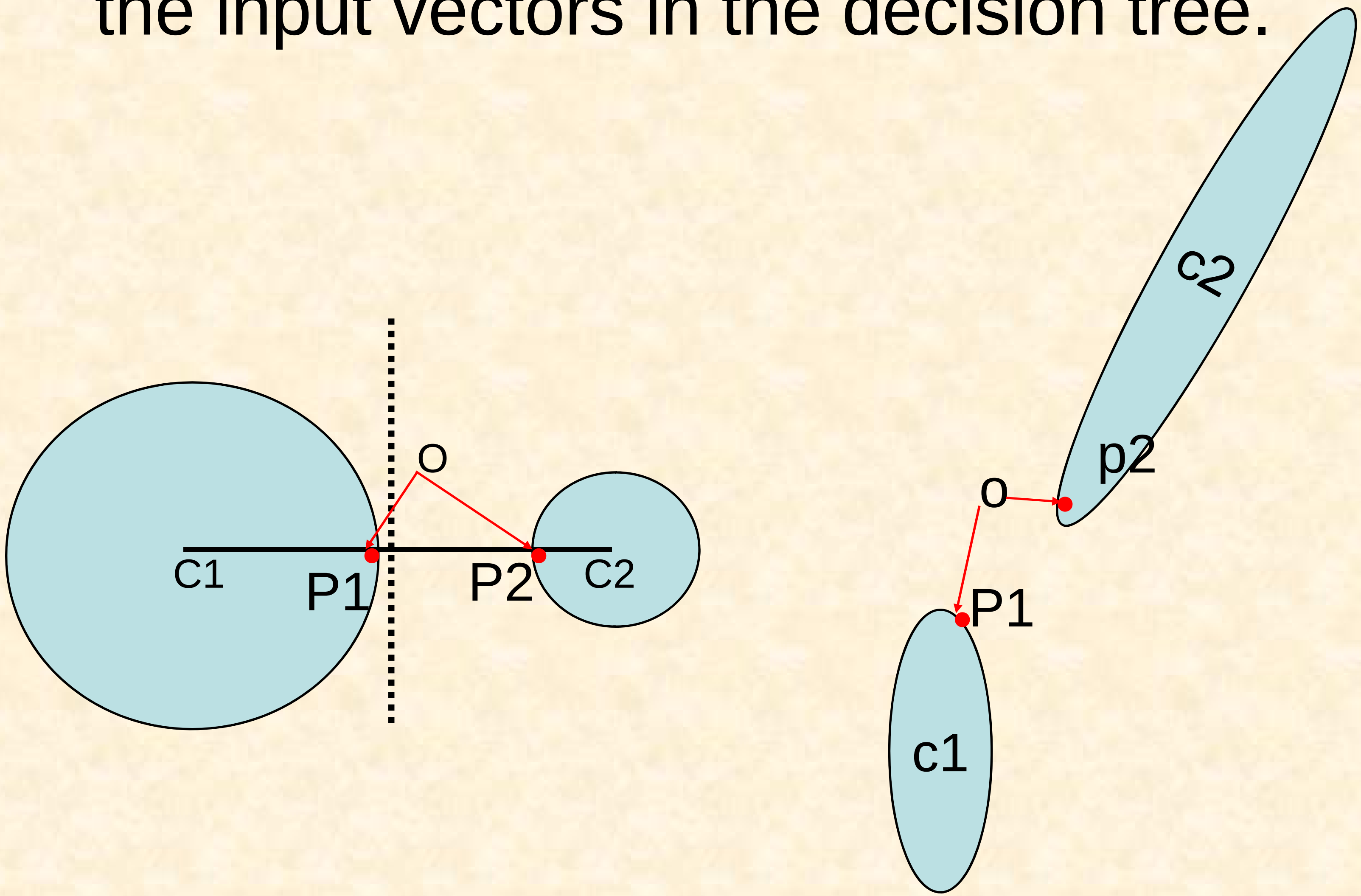
(2)Non-spherical shape of clusters



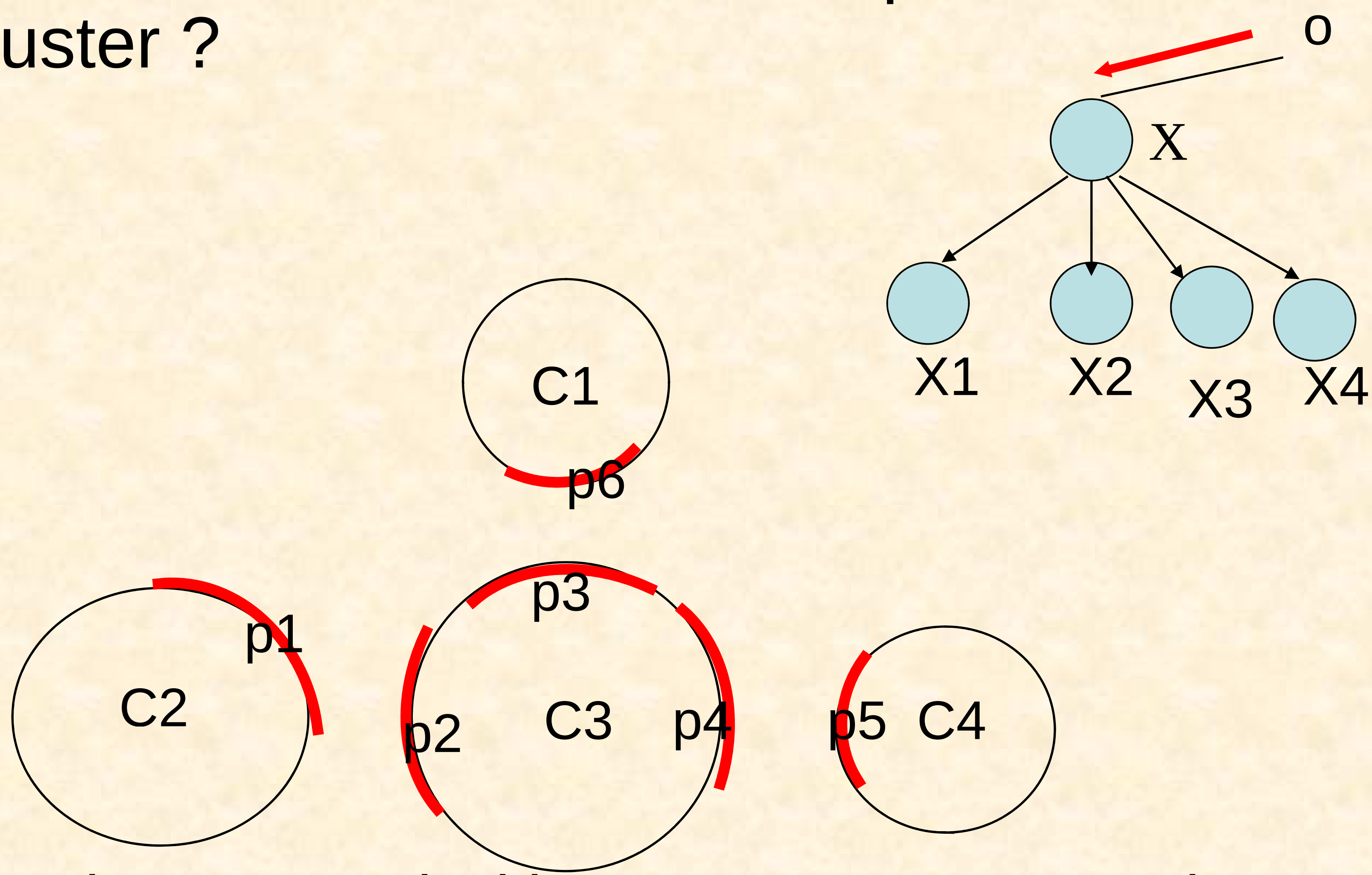
- Danger region among the clusters



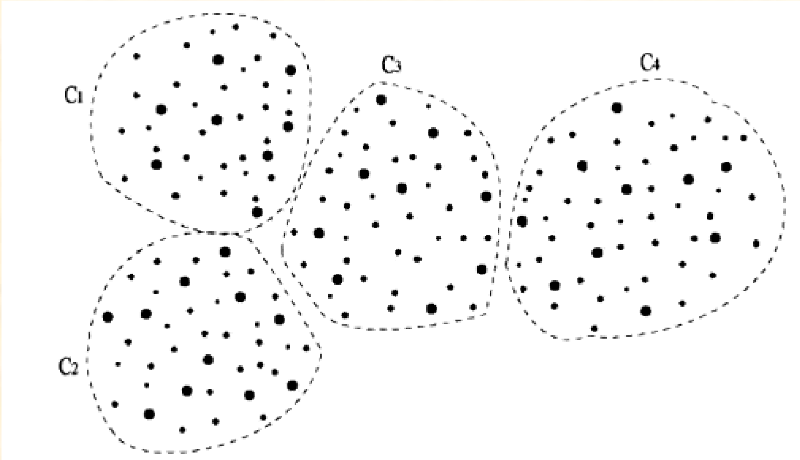
- <u>Classification point</u> is defined to classify the input vectors in the decision tree.



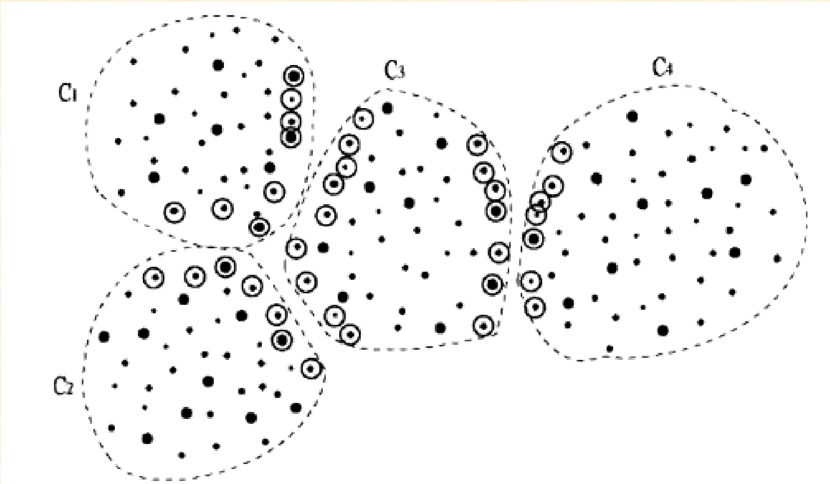
- How to find the classification points in a cluster ?



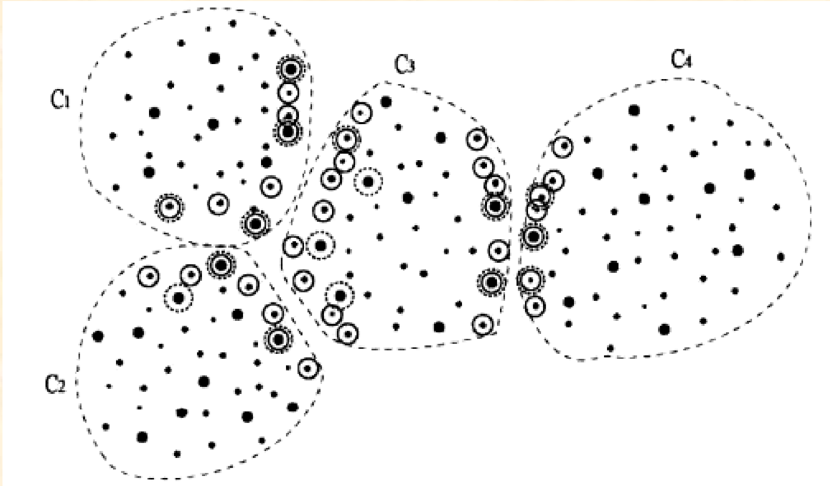
– O is compared with p1, p2, p3, p4, p5 and p6.



(1)



(2)



(3)

Conclusions

- Variable-branch decision tree can also be applied to recognition applications.
- Traditional NCUT tree can be improved by genetic algorithm.
- Adaptive variable-branch decision tree can be proposed in the further.

感謝