11/03/23 UNIT-3

Clustering and Regression

⇒ Clustering:

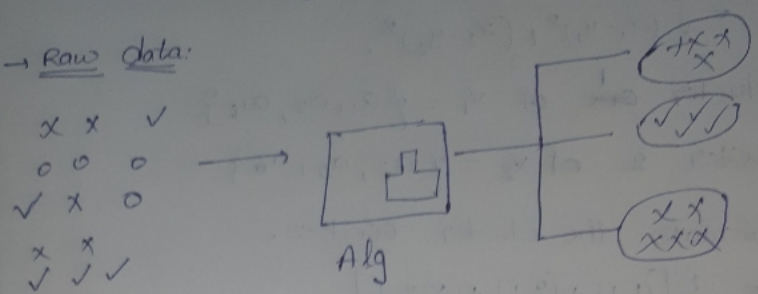
Clustering (or) cluster analysis is a machine learning techniques, which groups the unlabelled dataset. It can be defined as "A way of grouping the data points into different clusters, consisting of similar data points.

* The objects with the possible similarites remain in a group that has less (or) no similarites with another group.
* It does it by finding some similar patterns in the unlabelled dataset such as shape, size, color behaviour etc.... & divides them as per the presence & absence of the similar patterns.
* It is an unsupervised learning method, hence no supervision is provided to the algorithm & it deals with the unlabeled dataset.
* After applying this clustering technique, each cluster (or) group is provided with a cluster id. ml system can use this id to simplify the processing of large & complex datasets.
* The clustering technique is commonly used for statistical data analysis some most common users of this technique are,

1. Market segmentation.
2. Statistical data Analytics,
3. Social network Analysis.

4. Image Segmentation
5. Anomaly detection etc...

→ Raw data:



15/03/23

→ K means clustering Alg: to divide following data into 2 cluster set

x_1	1	2	2	3	4	5
x_2	1	1	3	2	3	5

Ans:-

1. Chossing randomly 2 cluster sets

$v_1 = (2,1)$

$v_2 = (2,3)$

2. finding the distance blw the cluster center & each data points

Distance Table:-

Data point	Distance from $v_1(2,1)$	Distance from $v_2(2,3)$	Assigned Centre
$a_1(1,1)$	1	$\sqrt{5} = 2.236$	v_1
$a_2(2,1)$	0	2	v_1
$a_3(2,3)$	2	0	v_2
$a_4(3,2)$	$\sqrt{2} = 1.414$	$\sqrt{2} = 1.414$	v_1
$a_5(4,3)$	$2\sqrt{2} = 2.828$	2	v_2
$a_6(5,5)$	5	$\sqrt{13} = 3.605$	v_2

→ Euclidian distance

(x_1, x_2) (y_1, y_2)

$= \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}$.

→ 3. cluster ~~one~~ 1 of $v_1 = \{a_1, a_2, a_4\}$

cluster 2 of $v_2 = \{a_3, a_5, a_6\}$

4. Recalculate the cluster centres.

$v_1 = \frac{1}{3}[(1,1) + (2,1) + (3,2)]$

$= \frac{1}{3}[(6,4)] = (2, 1.33)$.

$v_2 = \frac{1}{3}[(2,3) + (4,3) + (5,5)]$.

$= \frac{1}{3}[(11, 11)] = (3.67, 3.67)$.

5. Repeat from step 2, until we get same cluster centre (or) same cluster elements as in the previous iteration.

Data point	$v_1(2, 1.33)$	$v_2(3.67, 3.67)$	Assigned Centre
$a_1(1,1)$	1.05	3.78	v_1
$a_2(2,1)$	0.33	3.15	v_1
$a_3(2,3)$	1.67	1.8	v_1
$a_4(3,2)$	1.204	1.8	v_1
$a_5(4,3)$	2.605	0.75	v_2
$a_6(5,5)$	4.74	1.88	v_2

cluster 1 of $v_1 = \{a_1, a_2, a_3, a_4\}$

cluster 2 of $v_2 = \{a_5, a_6\}$

$$v_1 = \frac{1}{4}\left[(1,1) + (2,1) + (3,2) + (2,3)\right]$$

$$= \frac{1}{4}(8,7) = (2, 1.75).$$

$$v_2 = \frac{1}{2}\left[(4,3) + (5,5)\right]$$

$$= \frac{1}{2}(9,8) = (4.5, 4).$$

→ so, cluster Elements are not same as previous Elements.

Data point	v_1 (2, 1.75)	v_2 (4.5, 4)	Assigned Centre
a_1 (1,1)	1.25	4.61	v_1
a_2 (2,1)	0.75	3.9	v_1
a_3 (2,3)	1.25	2.69	v_1
a_4 (3,2)	1.03	2.5	v_1
a_5 (4,3)	2.36	1.12	v_2
a_6 (5,5)	4.42	1.12	v_2

→ cluster 1 of $v_1 = \{a_1, a_2, a_3, a_4\}$

cluster 2 of $v_2 = \{a_5, a_6\}$

$\frac{1}{4}$) x cluster elements are same as in the previous iteration. So, our clusters

cluster 1 $= \{(1,1), (2,1), (2,3), (3,2)$

cluster 2 $= \{(4,3), (5,5)\}$

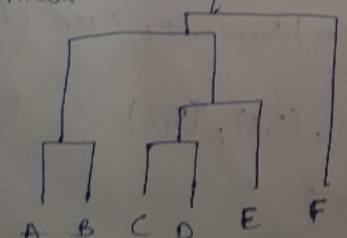
16/03/23

→ HAC :- Hierachical Agglomerative clustering

* HAC starts with one cluster, individual item in its own cluster & iteratively merge clusters until all the items belong to one cluster.
* Bottom up approach is followed to merge the clusters together.
* Demprogram (or) pictorially used to represent the HAC
* HAC represents 3 techniques:
  1. single, nearest (or) single linkage.
  2. Complete -- farthest distance (or) Complete linkage.
  3. Average distance (or) Average linkage.

1. S.N :- This is the diff b/w the closest members of the two clusters.
2. C.L :- This is the diff b/w the members that or farthest apart
3. A.L :- This Method involves looking at a distance b/w all pairs & Averages all of these distance. this is also called unwaited pair group mean average

* Demprogram :- A tree like structure which represent hierachical technique.



Problem:-

* Find the clusters using single link technique & draw the denprogram diagram.

	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	0.45	0.30

Sol:-

1. Euclidean distance

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

2.

	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	0.34	0.26	0.29	0	
P_6	0.23	0.25	0.11	0.22	0.39	0

Distance Table

→ Smallest element = 0.11

P_3 P_6

Recalculate the distance Matrix

(2)

	P_1	P_2	$P_3 P_6$	P_4	P_5
P_1	0				
P_2	0.23	0			
$P_3 P_6$	0.22	0.15	0		
P_4	0.37	0.20	0.15	0	
P_5	0.34	0.34	0.26	0.29	0

(0.14) → Small value.

$\min(\text{dist}(x, y))$

$\min(\text{dist}(P_3 P_6), P_1) = \min(\text{dist}((P_3, P_1), (P_6, P_1)))$

$= \min(\text{dist}(0.22, 0.23))$

$= 0.22$

(3) updated dist Matrix.

	P_1	$P_2 P_5$	$P_3 P_6$	P_4
P_1	0			
$P_2 P_5$	0.23	0		
$P_3 P_6$	0.22	(0.15)	0	
P_4	0.37	0.20	0.15	0

Small element = 0.15

⇒ New cluster $[(P_2 P_5)(P_3 P_6)]$

	P_1	P_2P_5 P_3 P_6	P_4
P_1	0		
P_2P_5 P_3 P_6	0.22	0	
P_4	0.37	0.15 ↓ small.	0

→ min (dist [C P_2P_5 P_3 P_6), P_1]

= min (dist [(P_2P_5), P_1] [(P_3 P_6), P_1])

= min (dist (0.23) (0.22))

= 0.22.

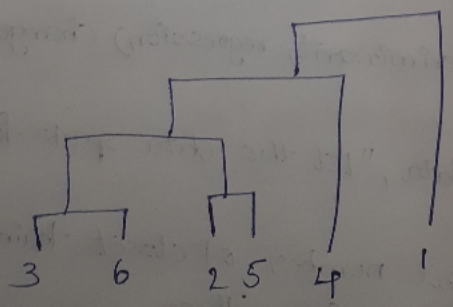
→ min (dist [C, P_2P_5 P_3 P_6), P_4]

= 0.15.

⇒ Updated

	P_1	P_2P_5 P_3P_6 P_4
P_1	0	0.22
P_2P_5 P_3 P_6 P_4	0.22	0

Diagram :-



17/03/23

→ Agglomerative clustering:

Distance b/w two groups G_i & G_j

1) single link

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

2) Complete-link

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

Dendrogram :- Decompose data objects into a several levels of nested partitioning (tree of clusters) - called a dendrogram.

=> Density Estimation :-

* Given the training set $X = \{x^t\}_t$ drawn iid from $p(x)$
* Divide data into bins of size h
* Histogram :-

$$\hat{p}(x) = \frac{\#\{x - h < x^t \leq x + h\}}{2Nh}$$

=> Non-parametric methods:

* parametric (single global model), semiparametric (small number of local models).
* Non-parametric : similar inputs have similar outputs.
* functions (pdf, discriminant, regression) change smoothly.
* keep the training data, "let the data speak for itself".
* Given x, find a small number of closest training instances & interpolate from these.

* Aka Lazy / memory-based / case-based / instance-based learning.

⇒ Kernal Estimator:

Kernal function, eg:- Gaussian

Kernal:-

$$K(u) = \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{u^2}{2}\right]$$

Kernal Estimator:- (parzen window)

$$\hat{p}(x) = \frac{1}{Nh} \sum_{t=1}^{N} K\left[\frac{x - x^t}{h}\right].$$

→ K= Nearest Neighbour Estimator.

$$\hat{p}(x) = \frac{k}{2Nd_k(x)}.$$

Generalisation of multivariate data:

1. Kernal density Estimator.
2. multivariate Gussian kernal
   - ↓
   1. Spheric
   2. Ellipsoid

→ Hamming distance:

$$H(a, x^t) = \sum_{j=1}^{d} 1(a_j \neq x_j^t).$$

→ Non-parametric classification:

1. Kernal estimator
2. K NN Estimator

→ Non-parametric Regression:

1. Running mean Smoother
2. Kernal smoother.

1. $$\hat{g}(x) = \frac{\sum_{i=1}^{N} w\left(\frac{x - x^t}{h}\right) r^t}{\sum_{t=1}^{N} w\left(\frac{x - x^t}{h}\right)}$$

2. $$\hat{g}(x) = \frac{\sum_{t=1}^{N} K\left(\frac{x - x^t}{h}\right) r^t}{\sum_{i=1}^{N} K\left(\frac{x - x^t}{h}\right)}$$

⇒ Linear Discrimination:

$$g_i(x_i / w_{i0}) = w^T x + w = \sum w x + w.$$

*likelihood Vs Discriminant Based Classification.

1. Assume model for $P(x|C_i)$, use Bayes's rule to calculate $P(C_i|x)$ $g_i(x) = \log P(C_i|x)$.

2. D.B.C = Assume model for $\phi(x|\phi_i)$ NO density Estimation. Estimating the boundary is enough. Noneed to accurately Estimate the densities inside the boundaries.

18/03/23

→ KNN Algorithm:-

* KNN is a Non-parametric classification Method.
* It is used for classification & regression.
* It is very simple easy to implement supervised Machine learning algorithm.

Advantages:-

* more effective of training data is large.

→ problem :-

S.No	maths	CS	Result
1	4	3	Fail
2	6	7	pass
3	7	8	pass
4	5	5	fail
5	8	8	pass
6	6	8	?

(*) select K neighbours
K=3.

X = (math - 6, CS - 8).

→ Euclideam Distance:-

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

x_2, y_2 - Observe values
x_1, y_1 - Actual values.

1. $= \sqrt{(6-4)^2 + (8-3)^2}$
$= \sqrt{4 + 25} = \sqrt{29} = 5.38$

2. $\sqrt{(6-6)^2 + (8-7)^2} = 1$

3. $\sqrt{(6-7)^2 + (8-8)^2} = 1$

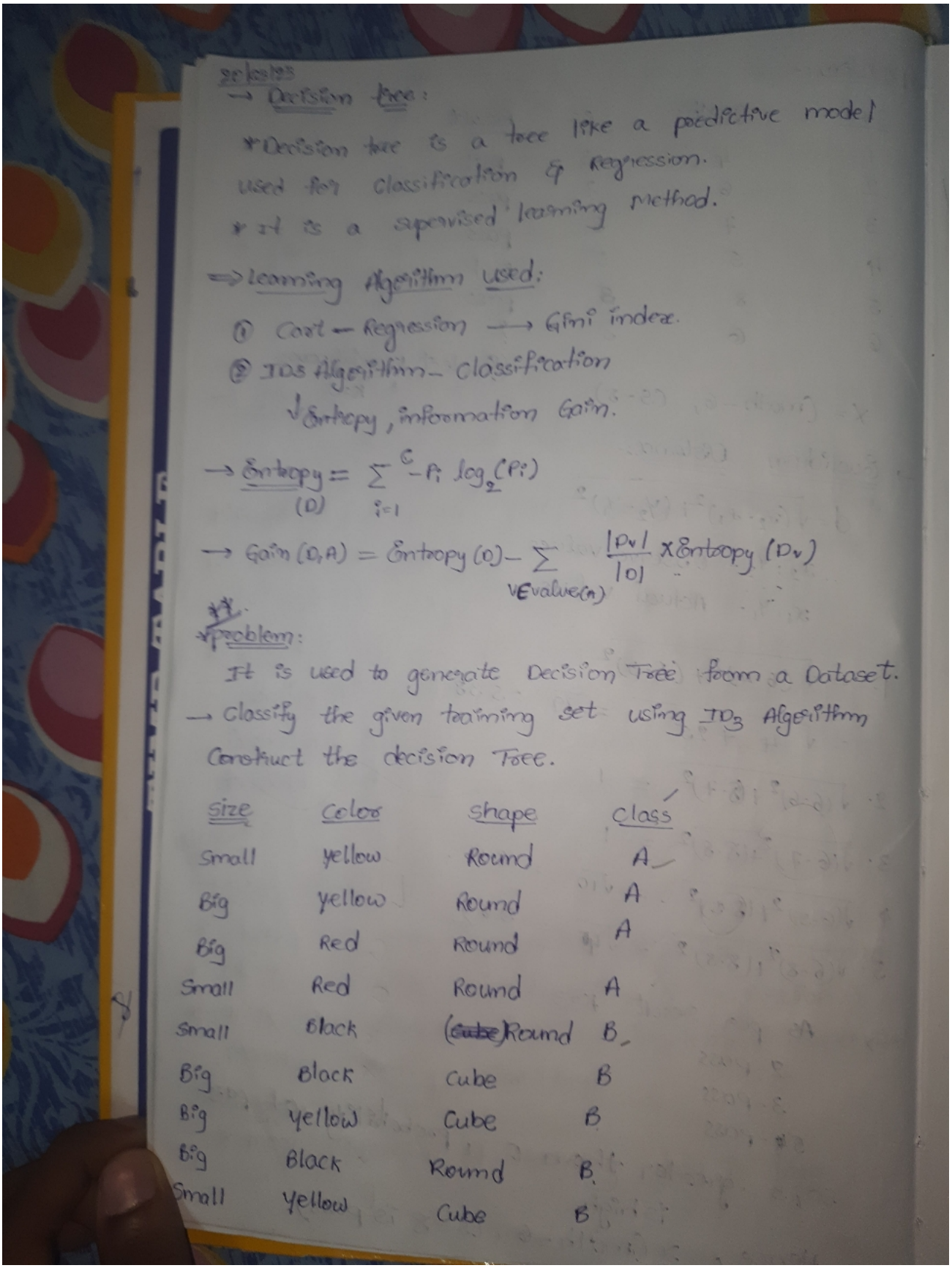
4. $\sqrt{(6-5)^2 + (8-5)^2} = \sqrt{10} = 3.16$.

5. $\sqrt{(6-8)^2 + (8-8)^2} = \sqrt{4} = 2$

→ As per result K=3
2 - pass
3 - pass
5 - pass

so, 3 greater than 0, probability of pass is high.

Hence, x (math = 6, CS = 8 is pass).



Decision Tree:

* Decision tree is a tree like a predictive model used for classification & regression.
* It is a supervised learning method.

⇒ Learning Algorithm used:

① Cart ← Regression → Gini index.

② ID3 Algorithm – Classification

↓ Entropy, information Gain.

→ $\underset{(D)}{\text{Entropy}} = \sum_{i=1}^{C} -P_i \log_2(P_i)$

→ $\text{Gain}(D,A) = \text{Entropy}(D) - \sum_{v \in \text{value}(A)} \frac{|D_v|}{|D|} \times \text{Entropy}(D_v)$

** Problem:

It is used to generate Decision Tree from a Dataset.

→ Classify the given training set using ID3 Algorithm Construct the decision Tree.

Size	Color	Shape	Class
Small	yellow	Round	A
Big	yellow	Round	A
Big	Red	Round	A
Small	Red	Round	A
Small	Black	(~~Cube~~) Round	B
Big	Black	Cube	B
Big	yellow	Cube	B
Big	Black	Round	B
Small	yellow	Cube	B

Sol:-

A=4 ✓

B=5 ✓

m=2

Calculate the Entropy(D):-

$$\text{Entropy}(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

$$= -\sum_{i=1}^{2} 4/9 \log_2(4/9)$$

$$= -4/9 \log_2(4/9) - 5/9 \log_2(5/9)$$

$$= (-0.444 * -1.171) - (0.555 * -0.850)$$

$$= 0.520 + 0.472$$

$$= 0.992$$

→ Identify the splitting attribute, Calculate the gain.

*Attribute - size:

	A	B	
small	2	2	-4
Big	2	3	-5

$$\text{Gain}(D, \text{size}) = 0.992 - \left[\frac{4}{9} \left(-\frac{2}{4} \log_2(2/4) - \frac{2}{4} \log_2(2/4) \right) + \frac{5}{9} \left(-\frac{3}{5} \log_2(3/5) - \frac{2}{5} \log_2(2/5) \right) \right].$$

$$= 0.992 - [0.444 + 0.539]$$

$$= 0.992 - 0.983$$

$$= 0.009$$

* Attribute - Color

	A	B	
yellow	2	2	-4
Red	2	0	-2
Black	3	0	-3

$$Gain(D, color) = 0.992 - \left[\frac{4}{9}\left(-\frac{2}{4}\log_2\left(\frac{2}{4}\right) - \frac{2}{4}\log_2\left(\frac{2}{4}\right)\right) + \frac{2}{9}\left(-\frac{2}{2}\log_2\left(\frac{2}{2}\right)\right) + \frac{3}{9}\left(-\frac{0}{3}\log_2\left(\frac{0}{3}\right) - \frac{3}{3}\log_2\left(\frac{3}{3}\right)\right)\right]$$

$$= 0.992 - [0.444 + 0 + 0]$$

$$= 0.55.$$

Size
Small
Big
Big
Small

* Attribute - shape

	A	B	
Round	4	2	-6
Cube	3	0	-3

$$Gain(D, shape) = 0.992 - \left[\frac{6}{9}\left(-\frac{4}{6}\log_2\left(\frac{4}{6}\right) - \frac{2}{6}\log_2\left(\frac{2}{6}\right)\right) + \frac{3}{9}\left(-\frac{3}{3}\log_2\left(\frac{3}{3}\right)\right)\right]$$

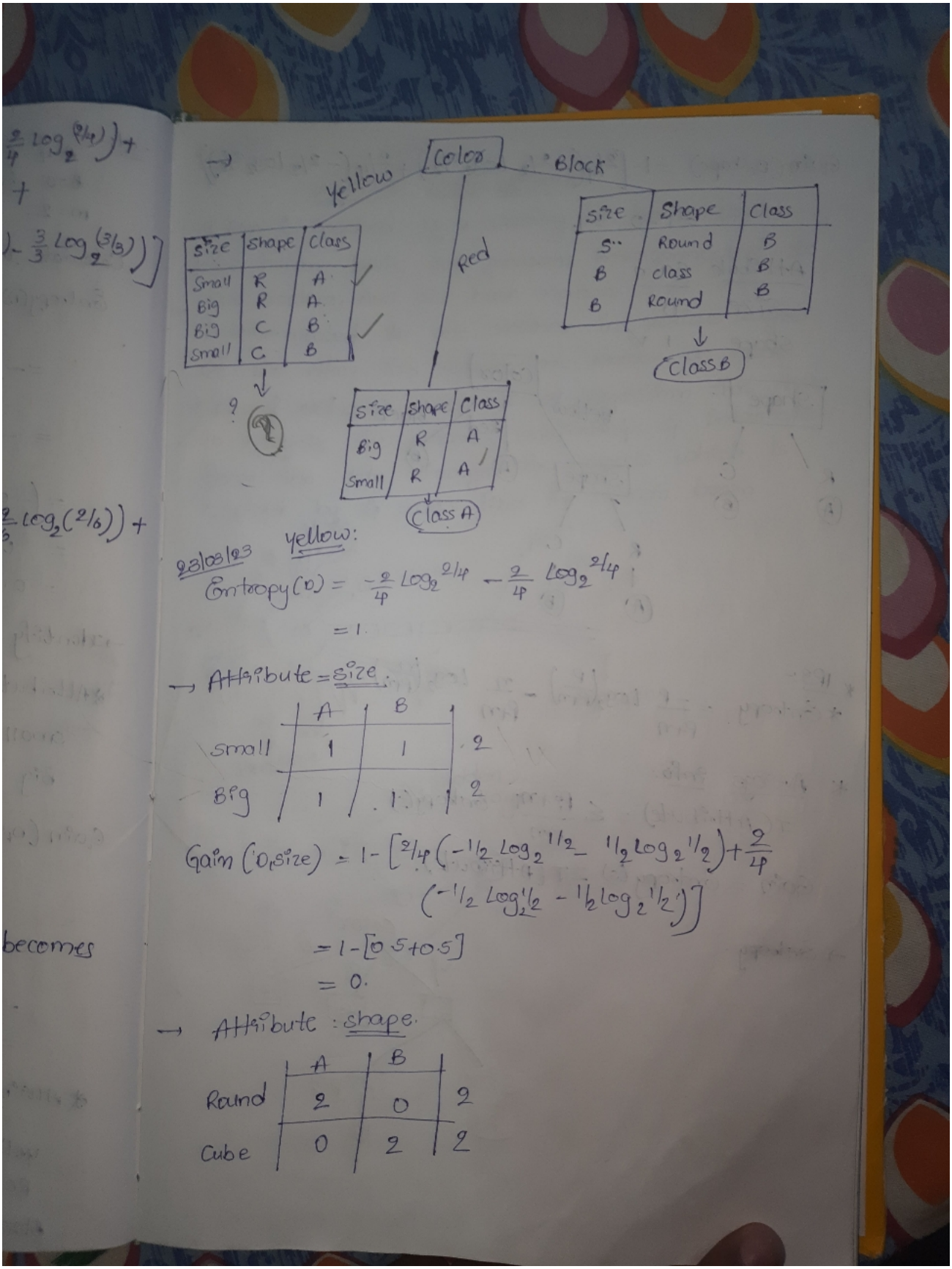
$$= 0.992 - 0.612$$

$$= 0.38.$$

→ Information Gain

Attribute	Gain
Size	0.009
Color	0.55
Shape	0.379

* Color has the largest gain, so its becomes the root node.



$\frac{2}{4}\log_2(2/4)) +$

$+$

$-\frac{3}{3}\log_2(3/3))]$

$\frac{2}{6}\log_2(2/6)) +$

becomes

→ Entropy(Yellow): $-\frac{2}{4}\log_2 2/4 - \frac{2}{4}\log_2 2/4$

$= 1.$

→ Attribute = size.

	A	B	
small	1	1	2
Big	1	1	2

Gain (O, size) $= 1 - [2/4(-1/2\log_2 1/2 - 1/2\log_2 1/2) + \frac{2}{4}(-1/2\log_2 1/2 - 1/2\log_2 1/2)]$

$= 1 - [0.5 + 0.5]$

$= 0.$

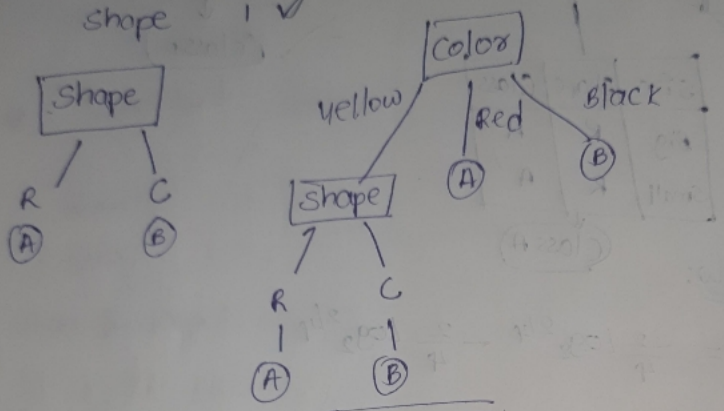
→ Attribute : shape.

	A	B	
Round	2	0	2
Cube	0	2	2

Gain(D, shape) $= 1 - \left[\frac{2}{4}\left(-\frac{2}{2}\log_2\frac{2}{2}\right) + \frac{2}{4}\left(-\frac{2}{2}\log_2\frac{2}{2}\right)\right]$

$= 1 - 0 = 1$

Attribute	Gain
Size	0
Shape	1 ✓



* IPS:-

* Entropy $= \frac{-p}{p+n}\log_2\left(\frac{p}{p+n}\right) - \frac{n}{p+n}\log_2\left(\frac{n}{p+n}\right)$.

* Average info:

$I(\text{Attribute}) = \sum \frac{p_i + n_i}{p+n}$ Entropy(S).

Gain = Entropy(S) $- I(\text{Attribute})$.

→ ~~Entropy~~.