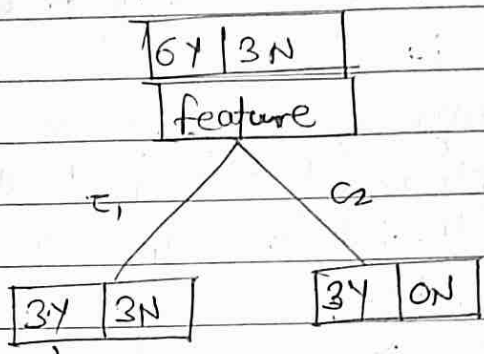
* Decision Tree

* Entropy Calculation :→

feature	o/p
c_1	Y
c_2	Y
c_1	Y
c_2	Y
c_1	Y
c_1	N
c_2	Y
c_1	N
c_1	N



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

N → Number of class. (2 → Y/N)

$$H(S) = -P_Y \log_2(P_Y) - P_N \log(P_N)$$

∴ Calculating for class c_1

$$H(S) = -\frac{3}{6}\log_2\left(\frac{3}{6}\right) - \frac{3}{6}\log_2\left(\frac{3}{6}\right)$$

$$= -\frac{1}{2}(-1)\log_2 2 - \frac{1}{2}(-1)\log_2 2$$

Heighest entropy that we can get → $= 1$

∴ This split is most impure split.

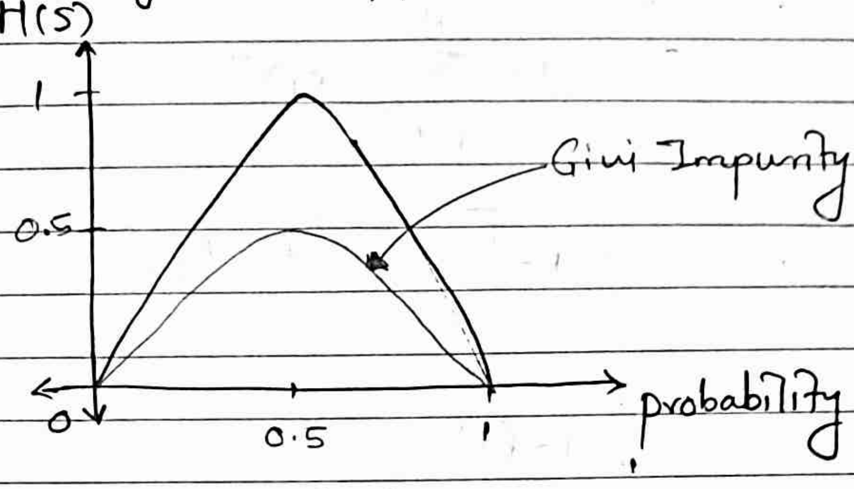
∴ Calculating for class c_2.

$$H(S) = -\frac{3}{3}\log_2\left(\frac{3}{3}\right) - \left(\frac{0}{3}\log_2\left(\frac{0}{3}\right)\right)$$

$$= -1\log_2(1) - 0$$

lowest entropy that we can get → $= 0$ → pure split.

* Graph probability vs entropy.



* <u>Gini Impurity Calculation</u> :-

- we are going we same above problem :-

3Y	3N

$$G.I = 1 - \left[P_Y^2 + P_N^2 \right]$$

$$G.I = 1 - \sum_{i=1}^{N} (P_i)^2 \text{ —— General}$$

N → Number of classes [Going to predict].

$$G.I = 1 - \left[\left(\frac{3}{6}\right)^2 + \left(\frac{3}{6}\right)^2 \right]$$

$$= 1 - \left[\frac{1}{4} + \frac{1}{4} \right]$$

$$= 1 - 0.5$$

Heighest G.I that we Can get

$$G.I = 0.5$$

most impure split [Can't draw any Conclusion]

3Y | 0N

$$G.I = 1 - \sum_{i=1}^{N} (P_i)^2$$

$$= 1 - \left[\left(\frac{3}{3}\right)^2 + \left(\frac{0}{3}\right)^2 \right]$$

$$= 1 - (1)$$

least G.I That we can get → $= 0$ → pure split we can conclude as yes here.

* Calculation of information gain :

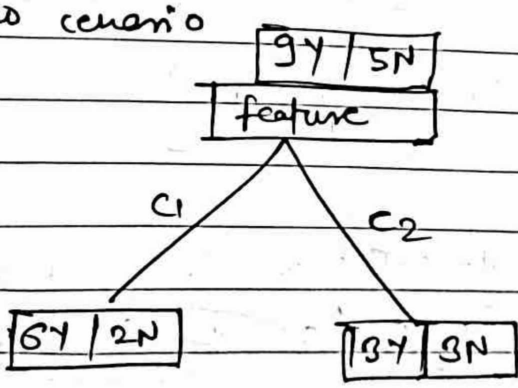
$$Gain = H(S) - \sum_{v \in val} \frac{|S_v|}{|S|} H(S_v)$$

Root node Entropy

child node entropy

$\Sigma \rightarrow$ of all child Node

with below cenario



- entropy for root feature

$$H(S) = -P_Y \log_2 (P_Y) - P_N \log_2 (P_N)$$

$$= -\frac{9}{14} \log_2\left(\frac{9}{14}\right) - \frac{5}{14} \log_2\left(\frac{5}{14}\right)$$

$$\boxed{H(S) = 0.9402}$$

- entropy for child Node

C1	C2
8Y | 2N	3Y | 3N

$$H(S) = -\frac{8}{2} \log_2\left(\frac{8}{2}\right) - \frac{2}{8} \log_2\left(\frac{2}{8}\right)$$

$= 0.81$

$= 1$

most impure split

split value

$$gain = H(S) \sum_{v \in val} \frac{|S_v|}{|S|} H(S_v)$$

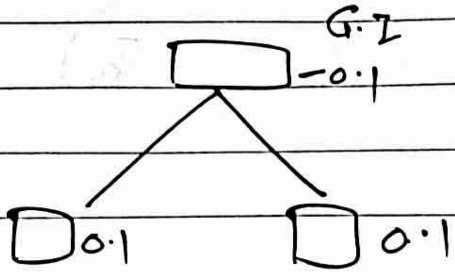
$$= 0.94 - \left[\frac{8}{14} \times 0.81 + \frac{6}{14} \times 1\right]$$

$$\boxed{gain = 0.049}$$

∴ The feature have more information gain that will be The root node selected by the Tree algorithms.

- The above process will be done on each level of node

- Information Gain Calculated in ID3 approach only
- In CART → sum of all Gini Impurity is calculated to select the root node.



sum of G.I = 0.1 + 0.1 + 0.1
= 0.3

The less the G.I sum is
This feature will be our root feature.

★ Pre-prunning :
Before building a decision tree I can Prune cut the branches.

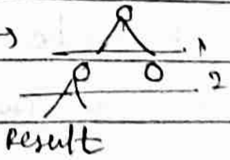
- if we going to build dT till complete depth?
① overfitting [Train accuracy will be good test accuracy will be bad.

② Computational will be expensive.

★ preprunning hyperparameter :-
1) Max depth
2) minimum Sample leaf
3) minimum Sample split.

1) Max depth :-
The depth of decision tree will decided by this parameter

Ex:- Max depth → 2 →



The minimum number of samples required to be at leaf node.

② Minimum - Samples leaf.

In the leaf node we can set minimum sample ~~amount~~ ~~of leaf node contain~~

- The minimum number of samples required to be at leaf node.
- if we have value at certain split ~~more~~ less than that remove it.

③ minimum Samples

- The minimum number of samples required to split an internal node.

① Max-depth

- The maximum depth of the trees.
- if none, then nodes are expanded until all leaves are pure or untill all leaves contain less than min_samples_split Samples.

Decision tree regressor :→

weight	Heart-disease
220	Y
180	Y
225	Y
150	N
155	N

Q) How to solve the above problem coz in weight we have continuous values

→ 1) Sort the values
2) Take the average of adjacent value
3) w.r.t every average value find out [Gini impurity / average gain]
as like we did for categorical values.

① Sorted data

	weight	HD
	155	N
167.5	180	Y
185	190	N
205	220	Y
222.5	225	Y

↓
average of two consecutive values.
↓
This is like a various classes
↓
~~for which classes we get less entropy and more gain~~
That class will consider as Root class [value].

③ [3Y | 2N]
weight
<167.5 → [0Y | 1N]
>167.5 → [3Y | 1N]

✓ calculate the gini impurity for above scenario

for left node
Gini impurity = 0
for Right node

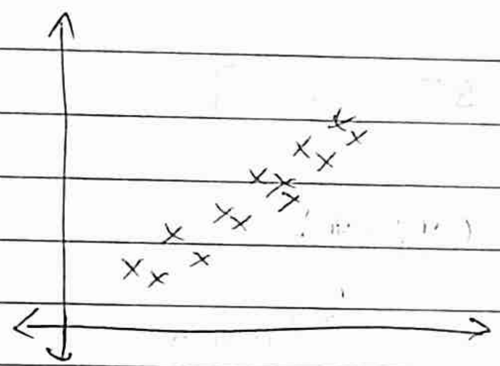
$$G.I = 1 - \left[\left(\frac{3}{4}\right)^2 + \left(\frac{1}{4}\right)^2\right]$$

$$\boxed{G.I = 0.37}$$

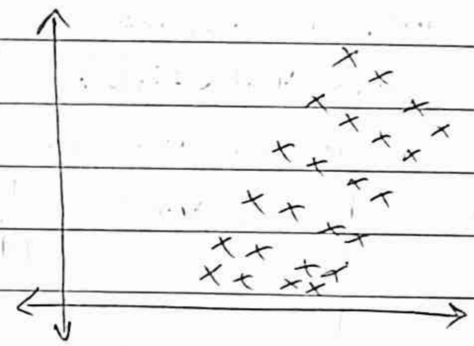
dependent feature

Height	weight
165	66
160	50
180	90
170	85
175	70

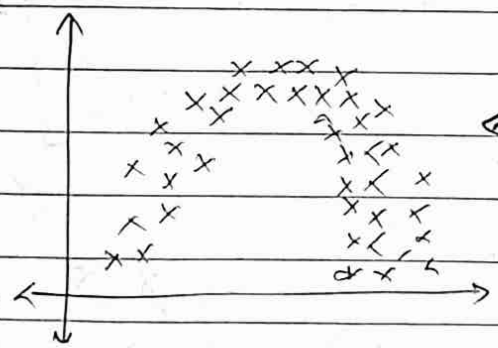
Gender	weight
M	66
F	50
M	90
F	85
M	70.



problem Suitable for linear Regression



problem Suitable for SVM.



NOT having linear Relation In features.

problem Suitable for DT

	Height	weight
	160	50
162.5	165	66
167.5	170	86
172.5	175	70
177.5	180	90

* Next page with this Example.

[50, 66, 85, 70, 90]

$$\begin{array}{ccc} & \boxed{\text{Height}} & \\ \swarrow_{<162.5} & & \searrow^{>162.5} \\ [50] & & [66, 85, 70, 90] \end{array}$$

- for Regression problem we found variance and reductance In variance. [MSE]

var(root) = [50, 65, 85, 70, 90]

$$var = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y})^2$$

$\hat{y}$ → mean

$\boxed{\mu = 72}$

$$var(root) = \frac{1}{5}\left[(72-50)^2 + (72-65)^2 + (72-85)^2 + (72-90)^2\right]$$

$\boxed{var(root) = 206}$

- var(left child node) = 0
- var(right child node) = [66, 85, 70, 90]

$$var = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y})^2$$

$\mu = 77.5$

$$= \frac{1}{4}\left[(77.5-66)^2 + (77.5-85)^2 + (77.5-70)^2 + (77.5-90)^2\right]$$

$= \cancel{706}$ 106.25

~~wrong~~ calculate the variance of child Node w.r.t mean of the root Node $\mu = 72.2$

Height var=206

var=0 (50)

(66, 85, 70, 90) var= 106.25

Redution in var = $\text{var}(\text{root}) - \sum_{i=1}^{N} w_i \times \text{var}(\text{child})$

$$= 206 \left[\frac{1}{5} \times 0 + \frac{4}{5} \times 106.25 \right]$$

$$= 206 - [0 + 85]$$

$$\boxed{= 121}$$

for threshold value → Height = 162.5 the Redution in variance = 121.

- like that we have to calculate for every Threshold value value

weight threshold value	Redution in VAR
162.5	121
167.5	116
172.5	105
177.5	130 → Root Node

The threshold which have ~~lowest~~ greatest Redution in VAR this Threshold we are selecting for spliting.

The final output for test data calculated by using the mean of the sample in the leaf node where my test data is falling.