⇒ Decision Tree

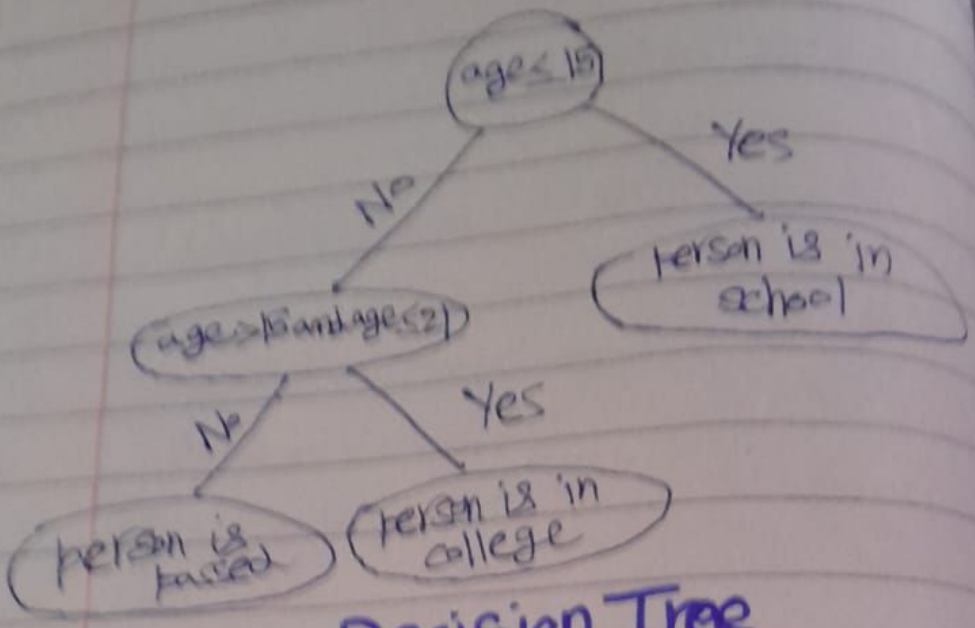
⇒ Decision Tree classifier:

Decision Tree Classifier → ID3
→ CART

Current Sklearn library uses CART Decision Tree classifier

- The basic difference b/w CART and ID3 is that in CART we can make only binary splits but in ID3 we can make more than 2 splits.
- Decision tree is similar to if else.

```
if(age ≤ 15):
    print("person is in school")
elif(age>15 and age≤21):
    print("person in college")
else:
    print("person is passed")
```



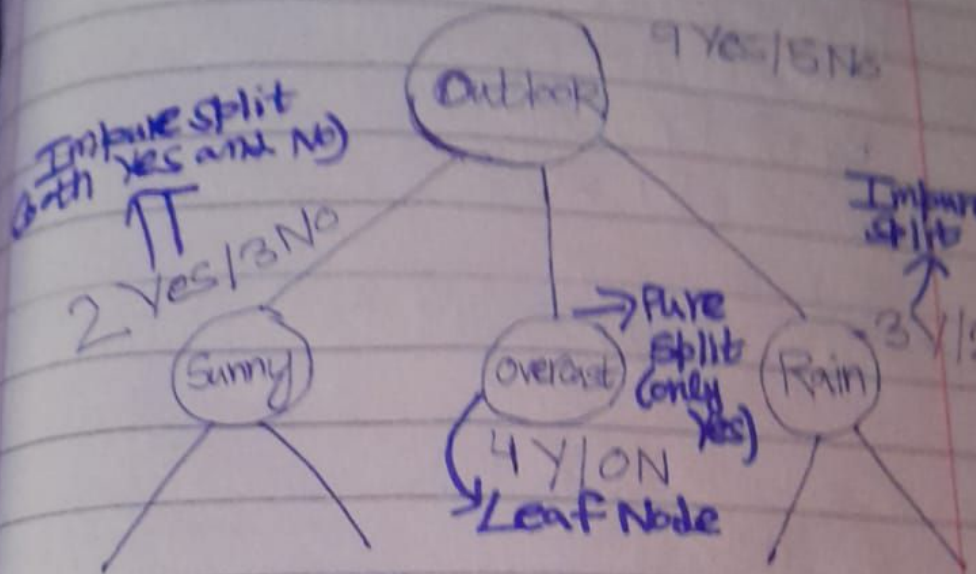
Decision Tree

Let a dataset

Independent | Dependent

Day	Outlook	Temperature	Humidity	Wind	Play
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

This is binary classification problem let's solve it with Decision Tree classifier



⇒ Now will split further w.r.t other features until we not get all leaf nodes.

① Purity → Pure or Impure split
Two techniques used to identify this are:
(i) Entropy
(ii) Gini Impurity

② What feature you need to select for splitting
Information gain

1) Entropy

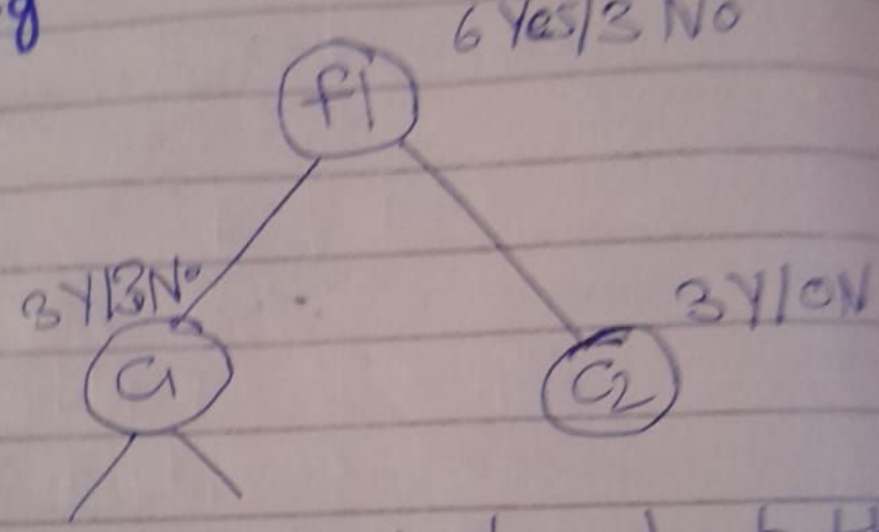
$H(G) = -P_+ \log_2 P_+ - P_- \log_2 P_-$

If we have binary classification (0 & 1)

$P_+ \Rightarrow$ Probability of being 1

$P_- \Rightarrow$ Probability of being 0

e.g



- Now we have to check the purity of splits using entropy

$H(G_1) = -P_+ \log_2 P_+ - P_- \log_2 P_-$

$P_+ \Rightarrow$ Probabilit of Yes $= \frac{3}{6} = \frac{1}{2}$

$P_- \Rightarrow$ Probability of No $= \frac{3}{6} = \frac{1}{2}$

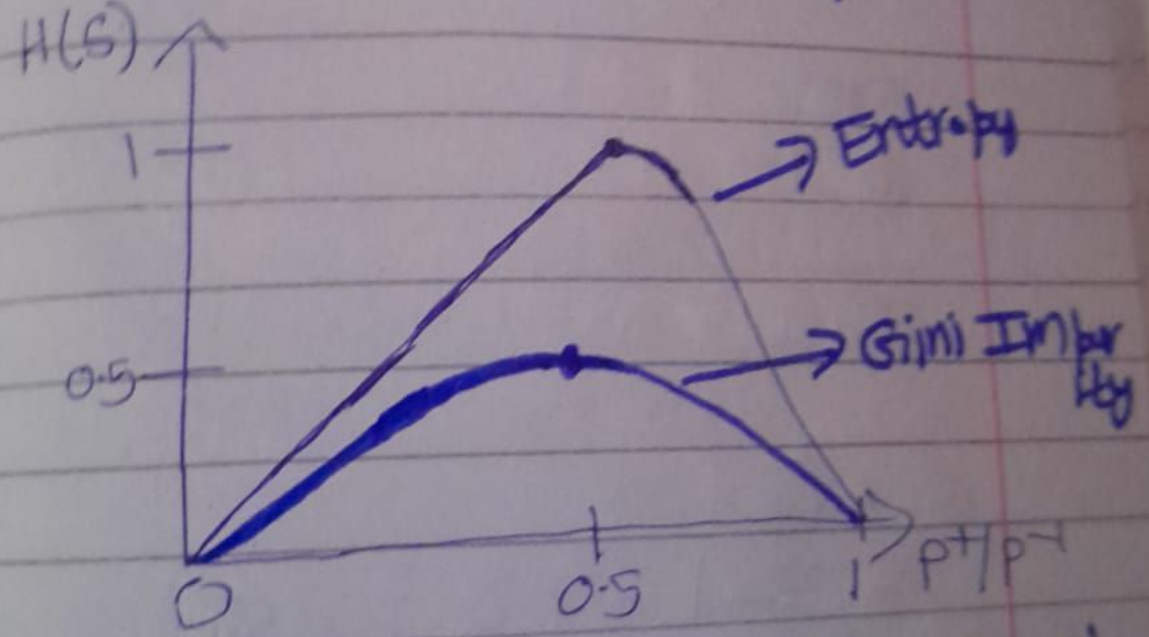
$H(C_1) = -\frac{3}{6}\log_2\frac{3}{6} - \frac{3}{6}\log_2\frac{3}{6}$

$H(C_1) = 1 \Rightarrow$ very Impure Split

$H(C_2) = -\frac{3}{3}\log_2\frac{3}{3} - 0\log_2 0$

$H(C_2) = -1\log_2 1 - 0$

$H(C_2) = 0 \Rightarrow$ Pure Split



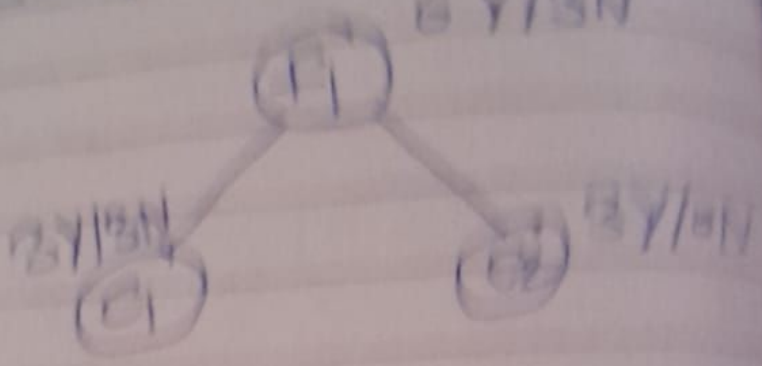
- Entropy ranges b/w 0 and 1 (including 0 & 1)
- Gini impurity ranges from 0 to 0.5

(2) Gini Impurity:

$$Gini = 1 - \sum_{i=1}^{n} (P_i)^2$$

$$Gini = 1 - ((P_1)^2 + (P_2)^2)$$

6 Y/3N



For C1:

$$Gini = 1 - \left(\left(\frac{3}{6}\right)^2 + \left(\frac{3}{6}\right)^2\right)$$

$Gini = 0.5$ ⇒ Complete Impure split

For C2:

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

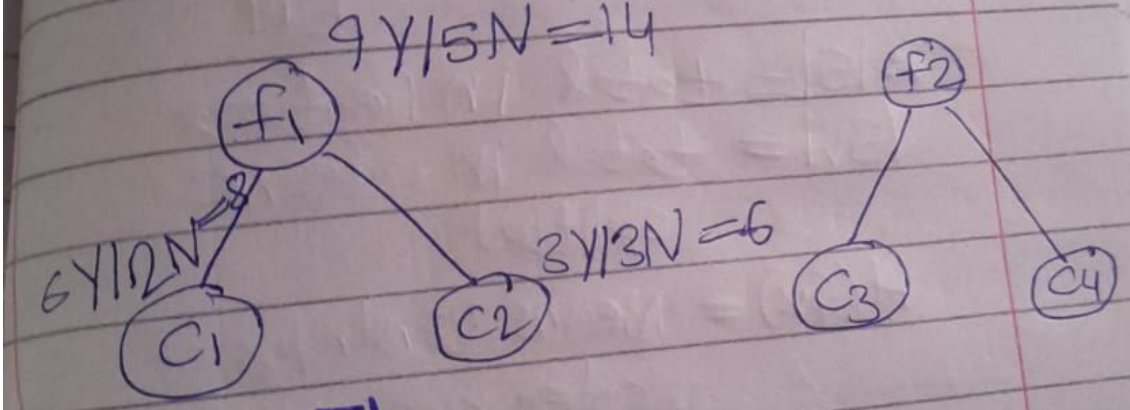
$Gini = 0$ ⇒ Pure split

We need to select which feature will be selected for splitting we do this using:

Information Gain:

$$\text{Gain}(S, f_1) = H(S) - \sum_{v \in \text{val}} \frac{|S_v|}{|S|} H(S_v)$$

e.g let's compare 2 features from which we start splitting



- For F1:

$$\text{Gain}(S, f_1) = H(S) - \sum_{v \in \text{val}} \frac{|S_v|}{|S|} H(S_v)$$

$$H(S) = -P_+ \log_2 P_+ - P_- \log_2 P_-$$

$$H(S) = -\frac{9}{14} \log_2 \frac{9}{14} - \frac{5}{14} \log_2 \frac{5}{14}$$

$$\boxed{H(S) = 0.94}$$

Now

$$H(C_1) = -\frac{6}{8}\log_2\frac{6}{8} - \frac{2}{8}\log_2\frac{2}{8}$$

$$\boxed{H(C_1) = 0.81}$$

$$H(C_2) = \frac{3}{6}\log_2\frac{3}{6} - \frac{3}{6}\log_2\frac{3}{6}$$

$$\boxed{H(C_2) = 1}$$

$$\text{Gain}(S, f_1) = 0.94 - \left[\frac{8}{14} \times 0.81 + \frac{6}{14} \times 1\right]$$

$\therefore |S|$ = total in features

$|S_v|$ = total in categories splitted

$H(S_v)$ = means Entropy of categories splitted

$$\boxed{\text{Gain}(S, f_1) = 0.049}$$

Now similarly if I calculate gain(S, F_2) and it is greater than gain(S, f_1) so we have to choose feature f_2 for splitting.

We can check in more depth by just adding S_v

When Should we use Entropy and when should we use Gini Impurity

- Entropy:

$$H(S) = -p_+ \log_2 p_+ - p_- \log_2 p_-$$

Where S is the sample space, p_+ is the probability of positive class and p_- is the probability of negative class.

⇒ Whenever your dataset is small we will use Entropy (because taking log takes time)

Gini Impurity:

$$Gini\ Impurity = 1 - \sum_{i=1}^{n} (p)^2$$

⇒ When our dataset is small we use Gini Impurity. Gini Impurity is used in most of cases

⇒ Sklearn uses Gini Impurity by default.

Decision Tree split for Numerical Features

let we have a dataset

f1	O/P
2.3	Yes
3.6	Yes
4	No
5.2	No
6.7	Yes
8.9	No
10.5	Yes

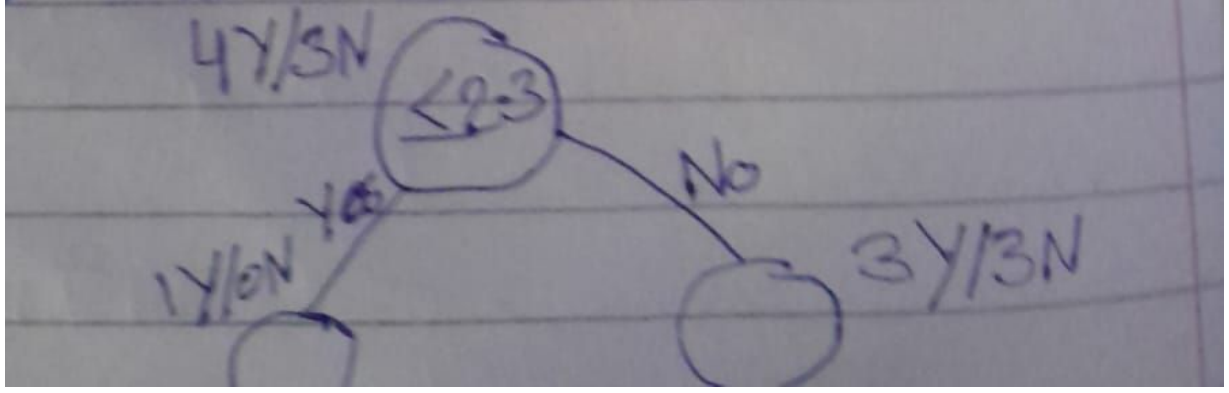
Step 01:

Sort the feature values in ascending order as we did

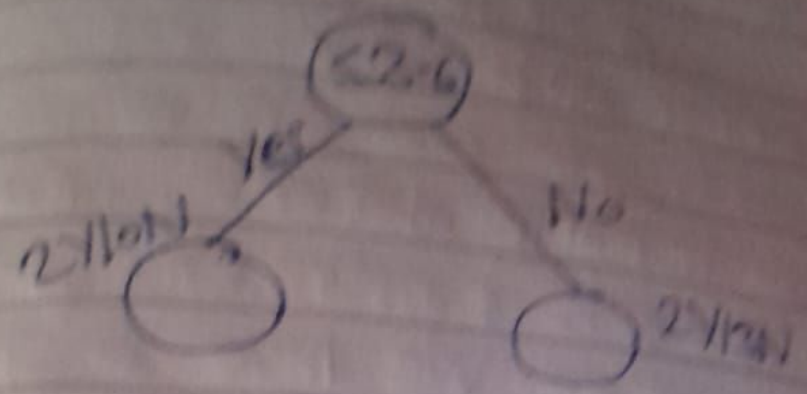
Step 02:

Select the threshold and make trees

Threshold = 2.3



(2) Threshold = 8.6



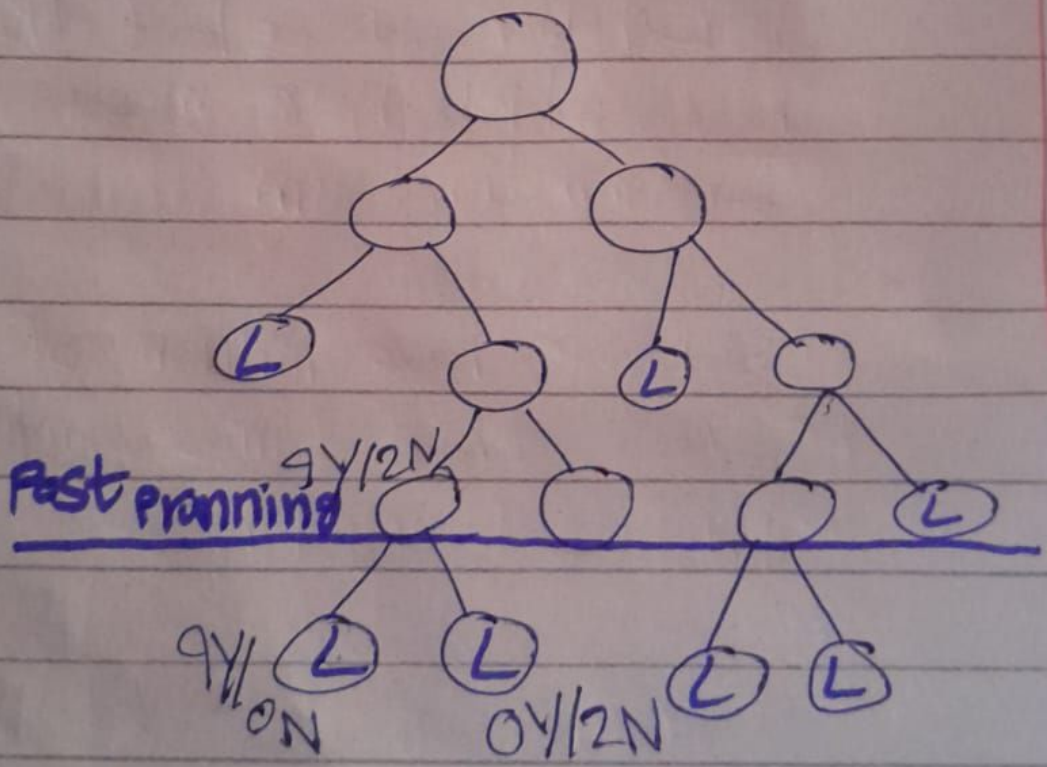
Now we will keep on splitting on the basis of threshold and select the root which have most Information Gain

- **Disadvantage:**

If we have million of records then time complexity will be very large.

⇛ Post Pruning and Pre Pruning in Decision Trees:

Let we have a training dataset and we are constructing decision tree. We have to split until we get all leaf nodes



⇛ When we do splitting till end we face problem of overfitting.

Pruning:

In order to reduce overfitting we have two techrique

1. Post Pruning
2. Pre Pruning

1. Post Pruning:

In post pruning we first construct the whole decision tree and then we prune it w.r.t depth

- Post pruning should be applied for smaller datasets
- You can see that we constracted whole decision tree and then we do post pruning by cutting and considering 9Y/2N as leaf node. because it has maximum probability of Yes and we don't want perfect purity

② Pre-Pranning:

In pre-pranning we play/ tune with hyperparameters (max depth, max features) etc. or do Hyperparameter Tunning while constracting decision tree.

- We use it for large data sets.

⇒ **Decision Tree Regression:**

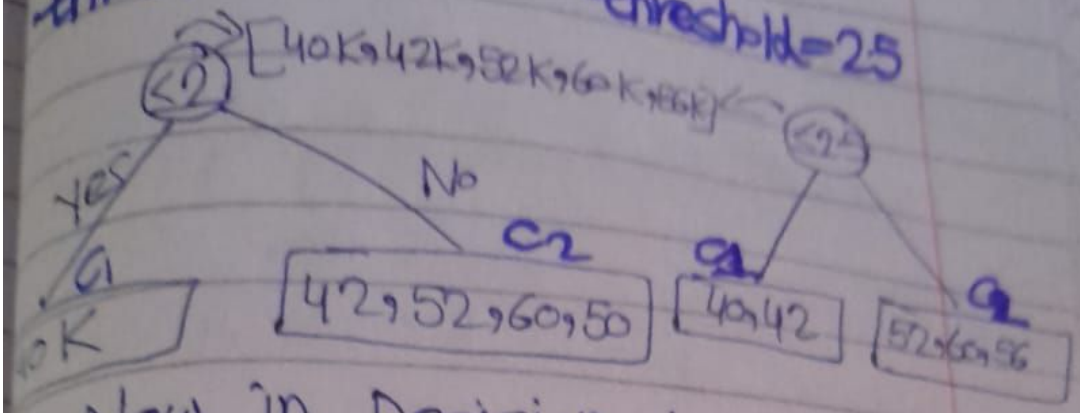
Let we have a dataset

Independent		Dependent/output
Experience	Gap	Salary
2	Yes	40K
2.5	Yes	42K
3	No	52K
4	No	60K
4.5	Yes	56K

Let perform two splits on the basis of feature experience with different threshold values

threshold=2

threshold=25



- Now in Decision tree classifier we decide which split is more appropriate using Information Gain.
- But here in Decision tree Regressor we decide which split is more appropriate using **Variance Reduction**
- For **Variance Reduction** we calculate **variance** (**Mean Squared Error**)

$$\text{variance} = \frac{1}{n}\sum_{i=1}^{n}(y-\bar{y})^2$$

↳ average

$\bar{y}$ = Average of Salary $= \frac{40+42+52+60+56}{5}$

$\bar{y} = 50K$

For threshold = 2

$$\text{variance(root)} = \frac{1}{5}\left[(40-50)^2 + (42-50)^2 + (52-50)^2 + (60-50)^2 + (56-50)^2\right]$$

$$\text{variance(root)} = \frac{1}{5}\left[100 + 64 + 4 + 100 + 36\right]$$

$$\boxed{\text{variance(root)} = 60.8} \Rightarrow \text{This is also for root of threshold} = 2.5$$

$$\text{variance}(c_1) = \frac{1}{1}(40-50)^2$$

$$\boxed{\text{variance}(c_1) = 100}$$

$$\text{variance}(c_2) = \frac{1}{n}\sum_{i=1}^{n}(y-\bar{y})^2$$

$$\text{variance}(c_2) = \frac{1}{4}\left[(42-50)^2 + (52-50)^2 + (60-50)^2 + (56-50)^2\right]$$

$$\boxed{\text{variance}(c_2) = 51}$$

Variance Reduction = $\text{var(root)} - \sum w_i \text{var}(c_i)$

$$\therefore w_i = \frac{\text{number of elements in child}}{\text{number of elements in root}}$$

$= 60.8 - \left[\frac{1}{5} \times 0 + \frac{4}{5} \times 0\right]$

Variance Reduction = 0

» For left split with threshold = 2

For threshold = 2.5

Variance (root) = 60.8

Variance (c_1) = 82

Variance (c_2) = 46.66

Variance reduction = var(root) − Σ(wi) var(child)

$= 60.8 - \left(\frac{2}{5} \times 82 + \frac{3}{5} \times 46.66\right)$

Variance reduction = 0.004

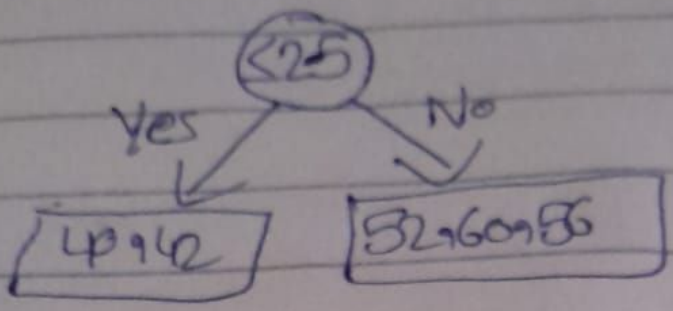
» For right split with threshold = 2.5

PS

Variance reduction (left split) < variance reduction (right split)

So we will select right split with threshold = 2.5

Now we have selected



Now let's consider these are our leaf nodes if we get test data

< 2.5 then the result will be average of values at left split i.e $\frac{40+42}{2} = 41$